Statistical machine learning and convex optimization

Francis Bach

INRIA - Ecole Normale Supérieure, Paris, France

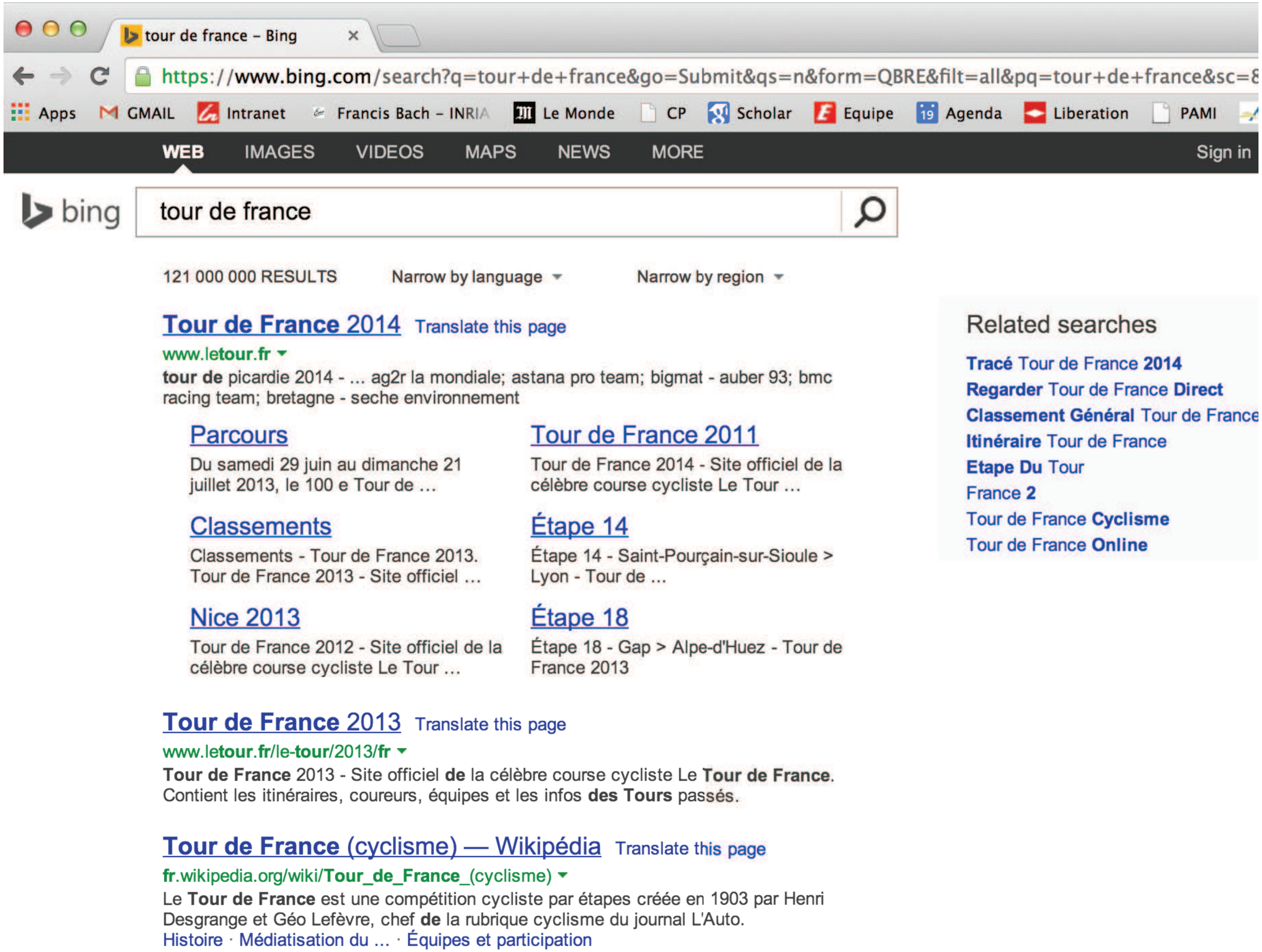
Spring school - Ecole des Mines 2017

Slides available: `www.di.ens.fr/~fbach/mines_2017_slides_bach.pdf`

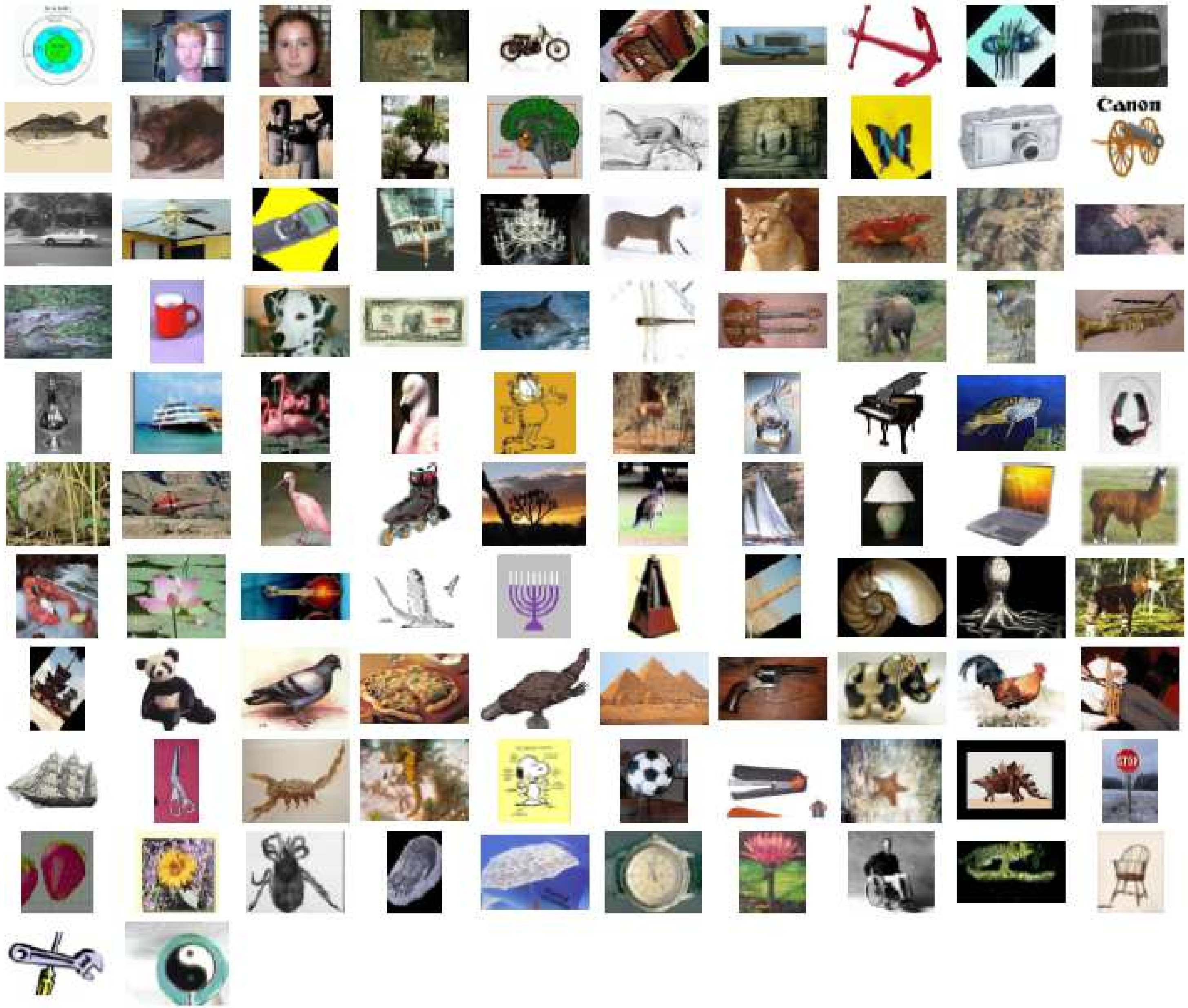
"Big data" revolution?
A new scientific context

- **Data everywhere**: size does not (always) matter
- **Science and industry**
- **Size and variety**
- **Learning from examples**
  - n observations in dimension d

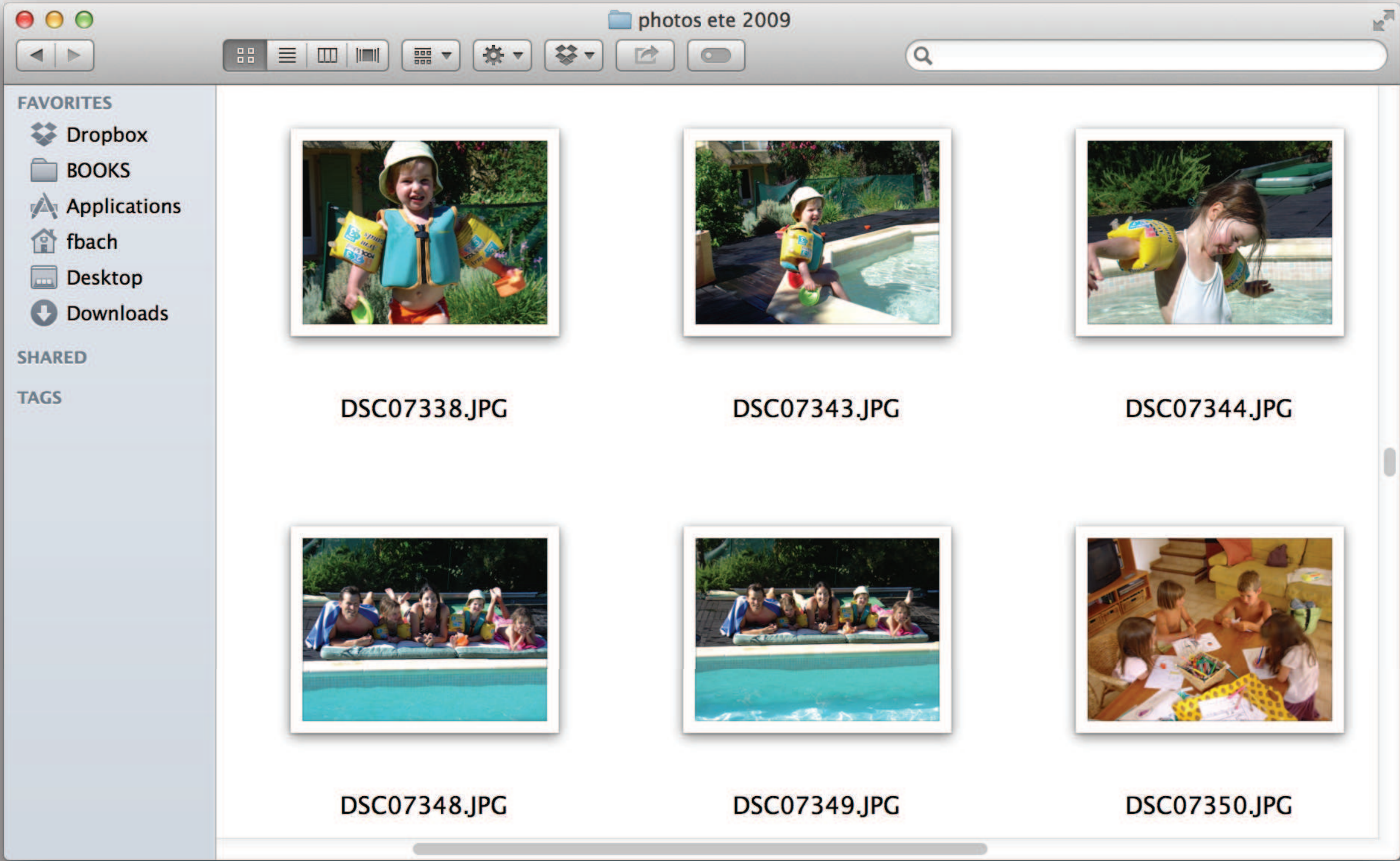
Search engines - Advertising



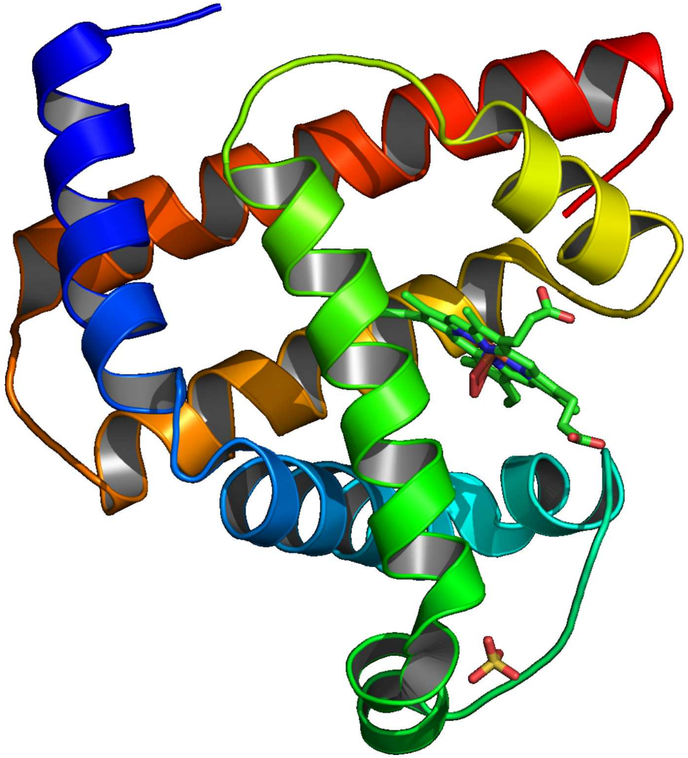
Visual object recognition



Personal photos



Bioinformatics



- **Protein**: Crucial elements of cell life

- **Massive data**: 2 millions for humans

- **Complex data**

Context
Machine learning for "big data"

- **Large-scale machine learning**: **large d, large n**
  - d : dimension of each observation (input)
  - n : number of observations

- **Examples**: computer vision, bioinformatics, advertising

Context

Machine learning for "big data"

- **Large-scale machine learning**: **large d, large n**
  - d : dimension of each observation (input)
  - n : number of observations

- **Examples**: computer vision, bioinformatics, advertising

- **Ideal running-time complexity**: $O(dn)$

Context
Machine learning for "big data"

- **Large-scale machine learning**: **large d, large n**
  - d : dimension of each observation (input)
  - n : number of observations
- **Examples**: computer vision, bioinformatics, advertising
- **Ideal running-time complexity**: $O(dn)$
- **Going back to simple methods**
  - Stochastic gradient methods (Robbins and Monro, 1951)
  - Mixing statistics and optimization

Outline - I

1. **Introduction**

   - Large-scale machine learning and optimization
   - Classes of functions (convex, smooth, etc.)
   - Traditional statistical analysis through Rademacher complexity

2. **Classical methods for convex optimization**

   - Smooth optimization (gradient descent, Newton method)
   - Non-smooth optimization (subgradient descent)

3. **Classical stochastic approximation** (not covered)

   - Robbins-Monro algorithm (1951)

Outline - II

4. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds

5. **Smooth stochastic approximation algorithms**
   - Non-asymptotic analysis for smooth functions
   - Least-squares regression without decaying step-sizes

6. **Finite data sets**
   - Gradient methods with exponential convergence rates

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 $\Phi(x) \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

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

convex data fitting term + regularizer

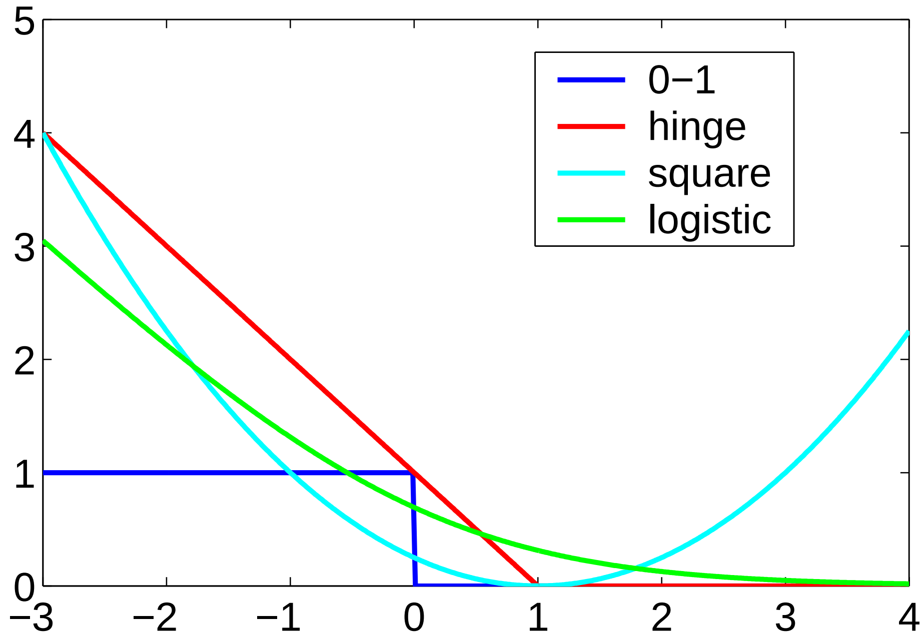
Usual losses

- **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:



Main motivating examples

- **Support vector machine** (hinge loss): non-smooth

$$\ell(Y, \theta^\top \Phi(X)) = \max\{1 - Y\theta^\top \Phi(X), 0\}$$

- **Logistic regression**: smooth

$$\ell(Y, \theta^\top \Phi(X)) = \log(1 + \exp(-Y\theta^\top \Phi(X)))$$

- **Least-squares regression**

$$\ell(Y, \theta^\top \Phi(X)) = \frac{1}{2}(Y - \theta^\top \Phi(X))^2$$

- **Structured output regression**
  - See Tsochantaridis et al. (2005); Lacoste-Julien et al. (2013)

Usual regularizers

- **Main goal**: avoid overfitting

- **(squared) Euclidean norm**: $\|\theta\|_2^2 = \sum_{j=1}^{d} |\theta_j|^2$
  - 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)

Usual regularizers

- **Main goal**: avoid overfitting

- **(squared) Euclidean norm**: $\|\theta\|_2^2 = \sum_{j=1}^{d} |\theta_j|^2$
  - 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 $\|\theta\|_1 = \sum_{j=1}^{d} |\theta_j|$
  - Perform model selection as well as regularization
  - Non-smooth optimization and structured sparsity
  - See, e.g., Bach, Jenatton, Mairal, and Obozinski (2012b,a)

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 $\Phi(x) \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

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

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 $\Phi(x) \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

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

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 $\Phi(x) \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

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

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**

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 $\Phi(x) \in \mathbb{R}^d$
- **(regularized) empirical risk minimization**: find $\hat{\theta}$ solution of

$$\min_{\theta \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^{n} \ell\big(y_i, \theta^\top \Phi(x_i)\big) \text{ such that } \Omega(\theta) \leqslant D$$

convex data fitting term + constraint

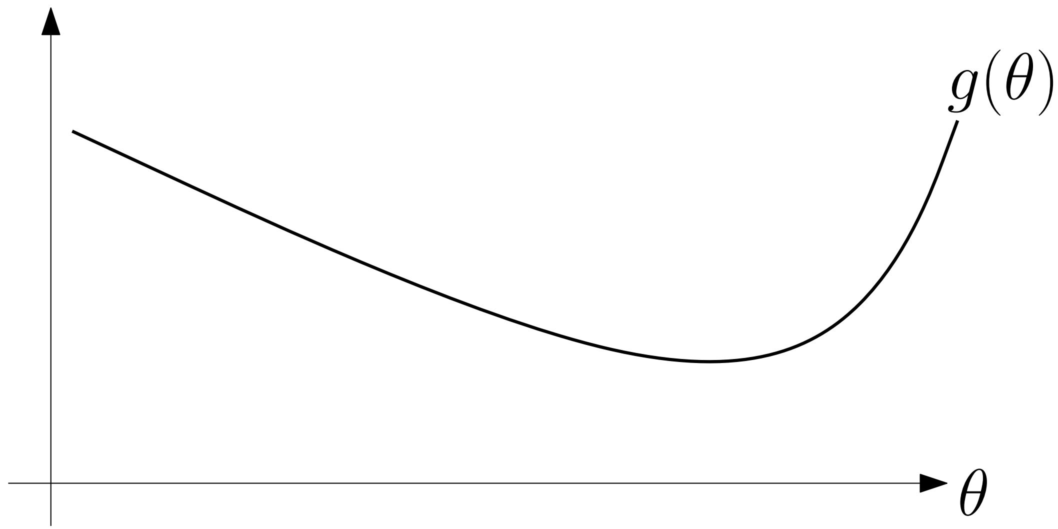
- Empirical risk: $\hat{f}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, \theta^\top \Phi(x_i))$ training cost
- Expected risk: $f(\theta) = \mathbb{E}_{(x,y)} \ell(y, \theta^\top \Phi(x))$ testing cost
- **Two fundamental questions**: (1) computing $\hat{\theta}$ and (2) analyzing $\hat{\theta}$
 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

General assumptions

- **Data**: n observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$, **i.i.d.**
- Bounded features $\Phi(x) \in \mathbb{R}^d$: $\|\Phi(x)\|_2 \leqslant R$
- 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
- Loss for a single observation: $f_i(\theta) = \ell(y_i, \theta^\top \Phi(x_i))$
  $\Rightarrow \forall i, \ f(\theta) = \mathbb{E} f_i(\theta)$
- **Properties of** $f_i, f, \hat{f}$
  - Convex on $\mathbb{R}^d$
  - Additional regularity assumptions: Lipschitz-continuity, smoothness and strong convexity

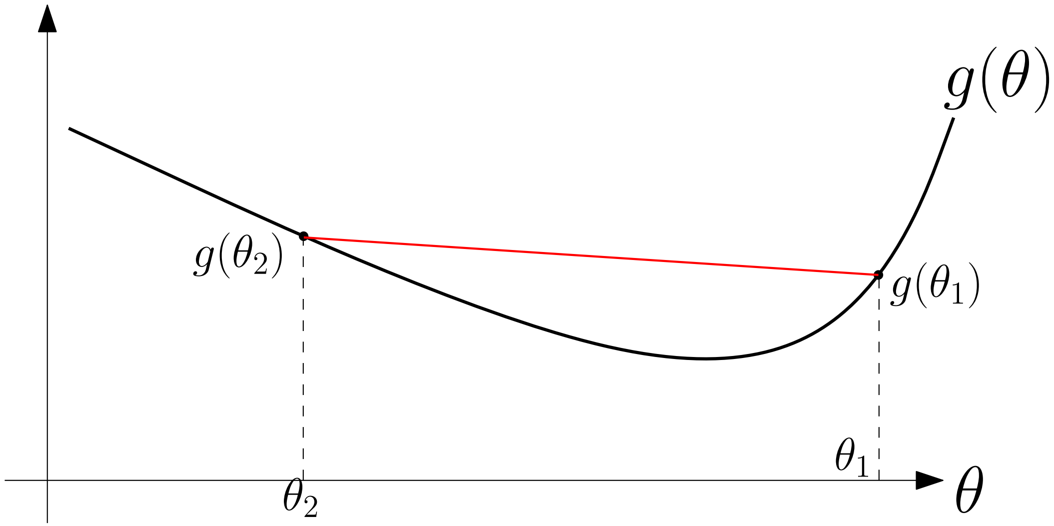
Convexity

- **Global definitions**



Convexity

- **Global definitions (full domain)**

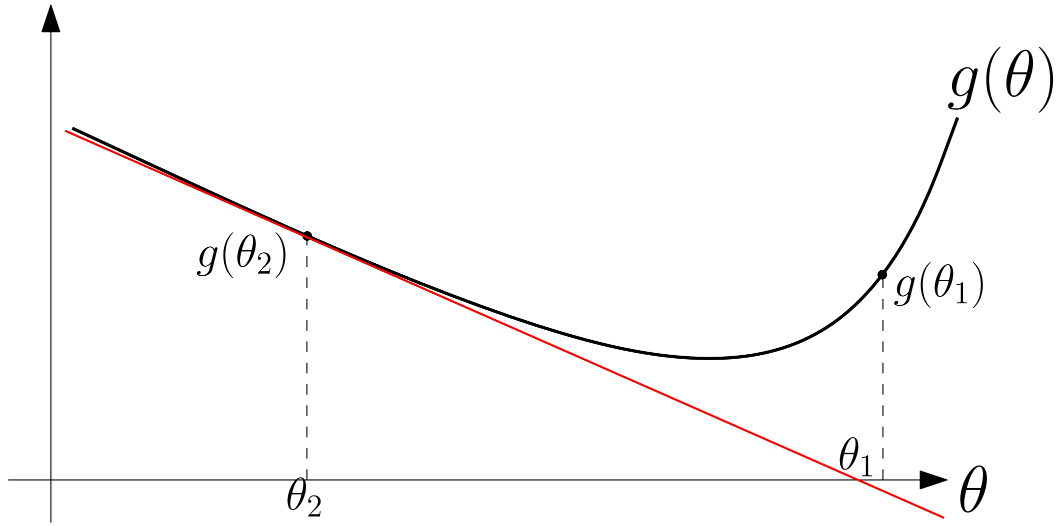


  – Not assuming differentiability:

$$\forall \theta_1, \theta_2, \alpha \in [0,1], \quad g(\alpha\theta_1 + (1-\alpha)\theta_2) \leqslant \alpha g(\theta_1) + (1-\alpha) g(\theta_2)$$

Convexity

- **Global definitions (full domain)**



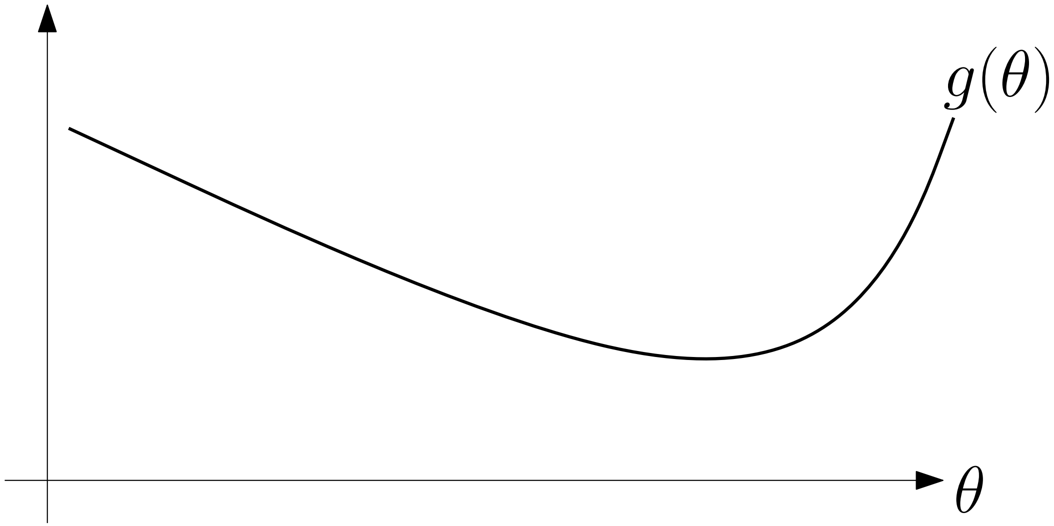
  – Assuming differentiability:

$$\forall \theta_1, \theta_2, \quad g(\theta_1) \geqslant g(\theta_2) + g'(\theta_2)^\top (\theta_1 - \theta_2)$$

- **Extensions to all functions with subgradients / subdifferential**

Convexity

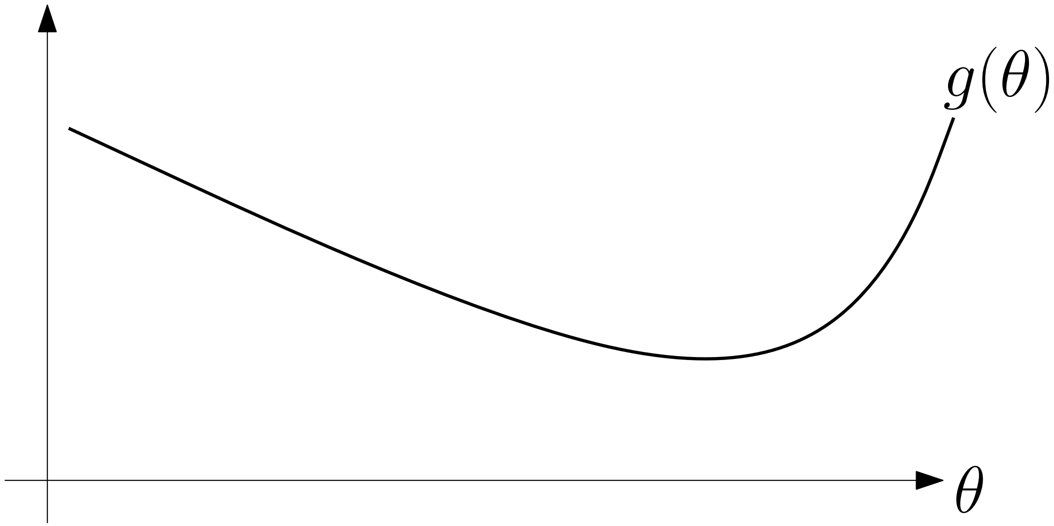
- **Global definitions (full domain)**



- **Local definitions**
  - Twice differentiable functions
  - $\forall \theta,\ g''(\theta) \succcurlyeq 0$ (positive semi-definite Hessians)

Convexity

- **Global definitions (full domain)**



- **Local definitions**
  - Twice differentiable functions
  - $\forall\theta,\ g''(\theta) \succcurlyeq 0$ (positive semi-definite Hessians)

- **Why convexity?**

Why convexity?

- **Local minimum = global minimum**
  - Optimality condition (non-smooth): $0 \in \partial g(\theta)$
  - Optimality condition (smooth): $g'(\theta) = 0$

- **Convex duality**
  - See Boyd and Vandenberghe (2003)

- **Recognizing convex problems**
  - See Boyd and Vandenberghe (2003)

Lipschitz continuity

- **Bounded gradients of g ($\Leftrightarrow$ Lipschitz-continuity)**: the function g if convex, differentiable and has (sub)gradients uniformly bounded by B on the ball of center 0 and radius D:

$$\forall \theta \in \mathbb{R}^d, \|\theta\|_2 \leqslant D \Rightarrow \|g'(\theta)\|_2 \leqslant B$$

$$\Leftrightarrow$$

$$\forall \theta, \theta' \in \mathbb{R}^d, \|\theta\|_2, \|\theta'\|_2 \leqslant D \Rightarrow |g(\theta) - g(\theta')| \leqslant B\|\theta - \theta'\|_2$$

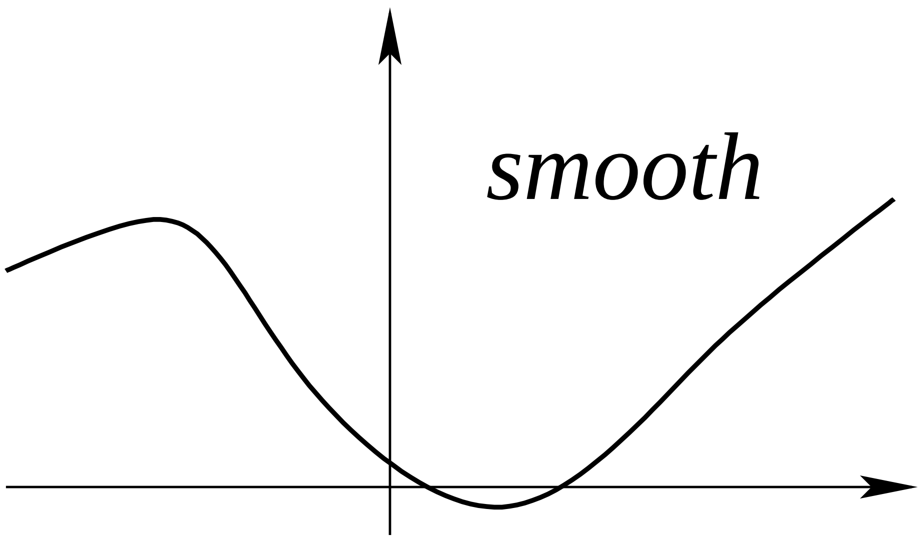
- **Machine learning**
  - with $g(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - G-Lipschitz loss and R-bounded data: $B = GR$

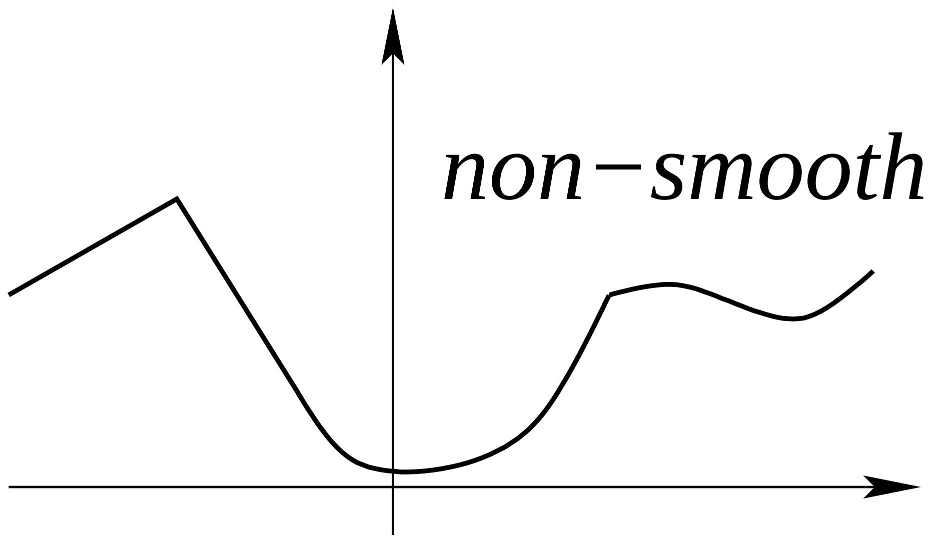
Smoothness and strong convexity

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

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d,\ \|g'(\theta_1) - g'(\theta_2)\|_2 \leqslant L\|\theta_1 - \theta_2\|_2$$

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





Smoothness and strong convexity

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

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, \ \|g'(\theta_1) - g'(\theta_2)\|_2 \leqslant L\|\theta_1 - \theta_2\|_2$$

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

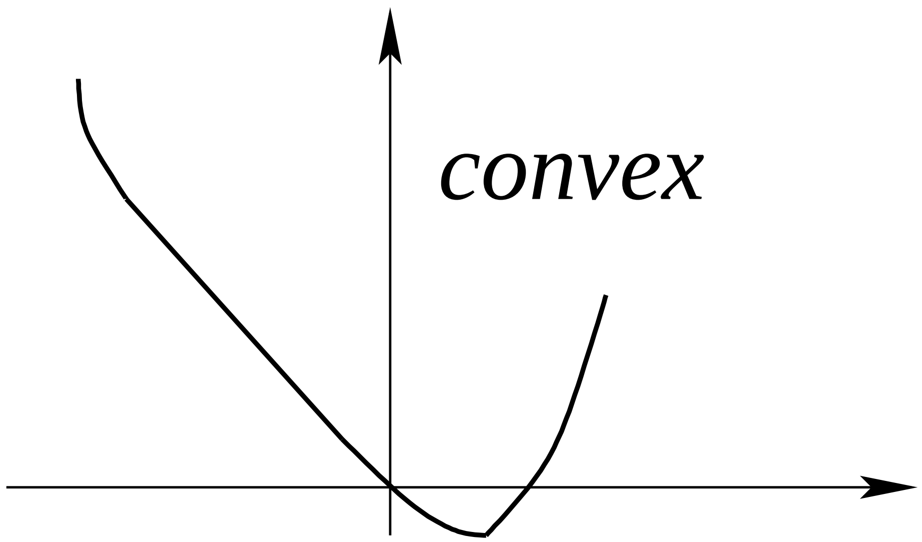
- **Machine learning**
  - with $g(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - Hessian $\approx$ covariance matrix $\frac{1}{n}\sum_{i=1}^n \Phi(x_i)\Phi(x_i)^\top$
  - $L_{\rm loss}$-smooth loss and R-bounded data: $L = L_{\rm loss} R^2$

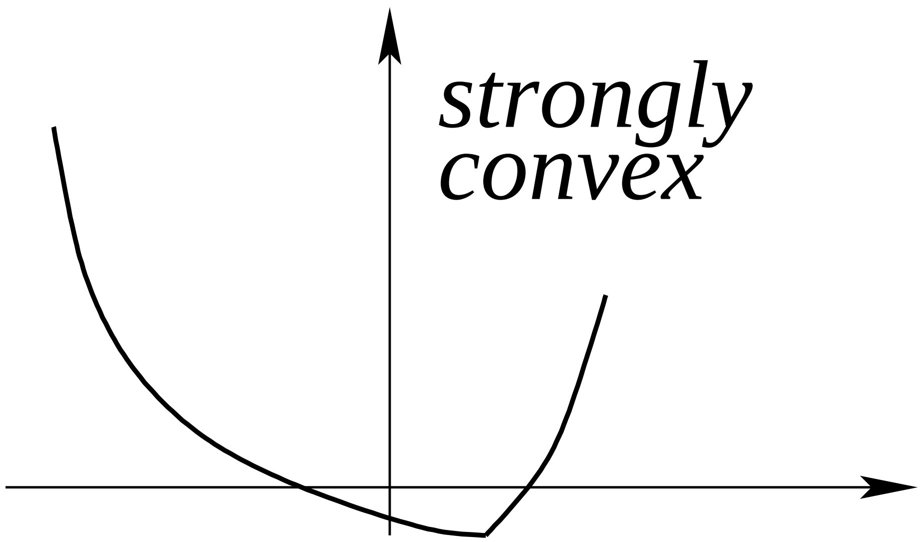
Smoothness and strong convexity

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

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, \;\; g(\theta_1) \geqslant g(\theta_2) + g'(\theta_2)^\top(\theta_1 - \theta_2) + \frac{\mu}{2}\|\theta_1 - \theta_2\|_2^2$$

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



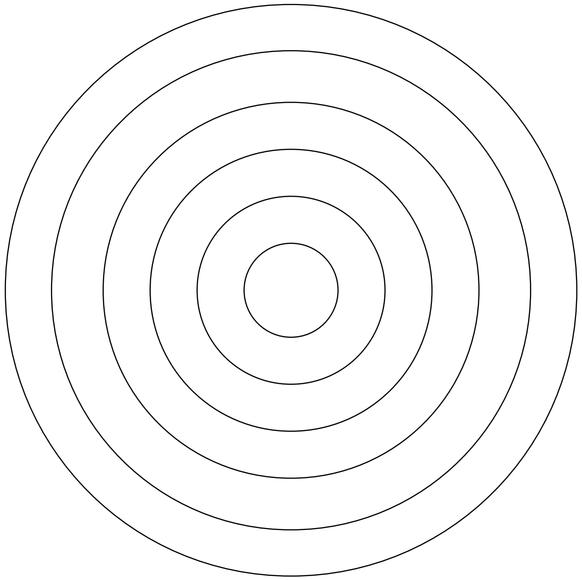


Smoothness and strong convexity

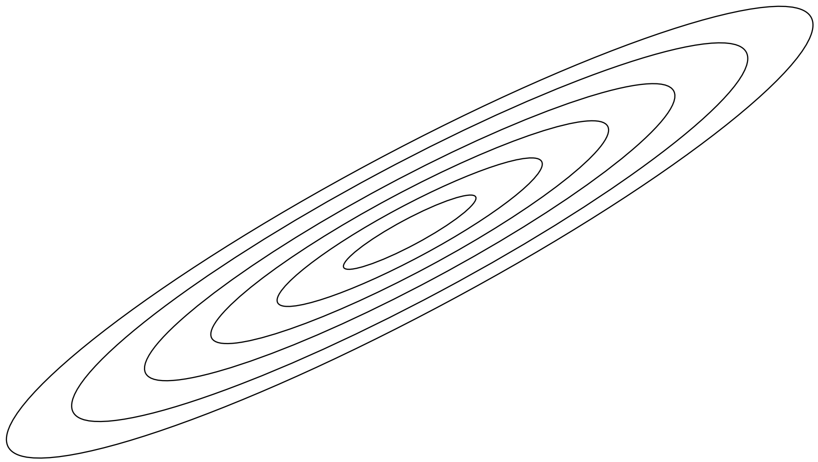
- A function $g : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex if and only if

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, \; g(\theta_1) \geqslant g(\theta_2) + g'(\theta_2)^\top (\theta_1 - \theta_2) + \frac{\mu}{2} \|\theta_1 - \theta_2\|_2^2$$

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



(large μ/L)



(small μ/L)

Smoothness and strong convexity

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

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, \; g(\theta_1) \geqslant g(\theta_2) + g'(\theta_2)^\top (\theta_1 - \theta_2) + \frac{\mu}{2} \|\theta_1 - \theta_2\|_2^2$$

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

- **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)

Smoothness and strong convexity

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

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, \; g(\theta_1) \geqslant g(\theta_2) + g'(\theta_2)^\top (\theta_1 - \theta_2) + \frac{\mu}{2} \|\theta_1 - \theta_2\|_2^2$$

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

- **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)

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

Summary of smoothness/convexity assumptions

- **Bounded gradients of g (Lipschitz-continuity)**: the function g if convex, differentiable and has (sub)gradients uniformly bounded by B on the ball of center 0 and radius D:

$$\forall \theta \in \mathbb{R}^d, \|\theta\|_2 \leqslant D \Rightarrow \|g'(\theta)\|_2 \leqslant B$$

- **Smoothness of g**: the function g is convex, differentiable with L-Lipschitz-continuous gradient g' (e.g., bounded Hessians):

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, \quad \|g'(\theta_1) - g'(\theta_2)\|_2 \leqslant L\|\theta_1 - \theta_2\|_2$$

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

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, \; g(\theta_1) \geqslant g(\theta_2) + g'(\theta_2)^\top(\theta_1 - \theta_2) + \frac{\mu}{2}\|\theta_1 - \theta_2\|_2^2$$

Analysis of empirical risk minimization

- **Approximation and estimation errors**: $\Theta = \{\theta \in \mathbb{R}^d, \Omega(\theta) \leqslant D\}$

$$f(\hat{\theta}) - \min_{\theta \in \mathbb{R}^d} f(\theta) \ = \ \left[f(\hat{\theta}) - \min_{\theta \in \Theta} f(\theta) \right] + \left[\min_{\theta \in \Theta} f(\theta) - \min_{\theta \in \mathbb{R}^d} f(\theta) \right]$$

Estimation error — Approximation error

  – NB: may replace $\min_{\theta \in \mathbb{R}^d} f(\theta)$ by best (non-linear) predictions

Analysis of empirical risk minimization

- **Approximation and estimation errors**: $\Theta = \{\theta \in \mathbb{R}^d, \Omega(\theta) \leqslant D\}$

$$f(\hat{\theta}) - \min_{\theta \in \mathbb{R}^d} f(\theta) = \Big[f(\hat{\theta}) - \min_{\theta \in \Theta} f(\theta) \Big] + \Big[\min_{\theta \in \Theta} f(\theta) - \min_{\theta \in \mathbb{R}^d} f(\theta) \Big]$$

Estimation error   Approximation error

1. **Uniform deviation bounds**, with $\boxed{\hat{\theta} \in \arg\min_{\theta \in \Theta} \hat{f}(\theta)}$

$$f(\hat{\theta}) - \min_{\theta \in \Theta} f(\theta) \leqslant 2 \cdot \sup_{\theta \in \Theta} |f(\theta) - \hat{f}(\theta)|$$

   – Typically slow rate $O\big(1/\sqrt{n}\big)$

2. **More refined concentration results** with faster rates $O(1/n)$

Slow rate for supervised learning

- **Assumptions** (f is the expected risk, $\hat{f}$ the empirical risk)
  - $\Omega(\theta) = \|\theta\|_2$ (Euclidean norm)
  - "Linear" predictors: $\theta(x) = \theta^\top \Phi(x)$, with $\|\Phi(x)\|_2 \leqslant R$ a.s.
  - G-Lipschitz loss: f and $\hat{f}$ are GR-Lipschitz on $\Theta = \{\|\theta\|_2 \leqslant D\}$
  - No assumptions regarding convexity

Slow rate for supervised learning

- **Assumptions** (f is the expected risk, $\hat{f}$ the empirical risk)
  - $\Omega(\theta) = \|\theta\|_2$ (Euclidean norm)
  - "Linear" predictors: $\theta(x) = \theta^\top \Phi(x)$, with $\|\Phi(x)\|_2 \leqslant R$ a.s.
  - G-Lipschitz loss: f and $\hat{f}$ are GR-Lipschitz on $\Theta = \{\|\theta\|_2 \leqslant D\}$
  - No assumptions regarding convexity

- With probability greater than $1 - \delta$

$$\sup_{\theta \in \Theta} |\hat{f}(\theta) - f(\theta)| \leqslant \frac{\ell_0 + GRD}{\sqrt{n}} \left[2 + \sqrt{2 \log \frac{2}{\delta}} \right]$$

- Expectated estimation error: $\mathbb{E}\big[\sup_{\theta \in \Theta} |\hat{f}(\theta) - f(\theta)|\big] \leqslant \dfrac{4\ell_0 + 4GRD}{\sqrt{n}}$

- Using Rademacher averages (see, e.g., Boucheron et al., 2005)

- **Lipschitz functions $\Rightarrow$ slow rate**

Motivation from mean estimation

- Estimator $\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} z_i = \arg\min_{\theta \in \mathbb{R}} \frac{1}{2n}\sum_{i=1}^{n}(\theta - z_i)^2 = \hat{f}(\theta)$

- From before:
  - $f(\theta) = \frac{1}{2}\mathbb{E}(\theta - z)^2 = \frac{1}{2}(\theta - \mathbb{E}z)^2 + \frac{1}{2}\operatorname{var}(z) = \hat{f}(\theta) + O(1/\sqrt{n})$
  - $f(\hat{\theta}) = \frac{1}{2}(\hat{\theta} - \mathbb{E}z)^2 + \frac{1}{2}\operatorname{var}(z) = f(\mathbb{E}z) + O(1/\sqrt{n})$

Motivation from mean estimation

- Estimator $\hat{\theta} = \frac{1}{n}\sum_{i=1}^{n} z_i = \arg\min_{\theta\in\mathbb{R}} \frac{1}{2n}\sum_{i=1}^{n}(\theta - z_i)^2 = \hat{f}(\theta)$
- From before:
  - $f(\theta) = \frac{1}{2}\mathbb{E}(\theta - z)^2 = \frac{1}{2}(\theta - \mathbb{E}z)^2 + \frac{1}{2}\operatorname{var}(z) = \hat{f}(\theta) + O(1/\sqrt{n})$
  - $f(\hat{\theta}) = \frac{1}{2}(\hat{\theta} - \mathbb{E}z)^2 + \frac{1}{2}\operatorname{var}(z) = f(\mathbb{E}z) + O(1/\sqrt{n})$
- More refined/direct bound:

$$\begin{aligned} f(\hat{\theta}) - f(\mathbb{E}z) &= \frac{1}{2}(\hat{\theta} - \mathbb{E}z)^2 \\ \mathbb{E}\big[f(\hat{\theta}) - f(\mathbb{E}z)\big] &= \frac{1}{2}\mathbb{E}\left(\frac{1}{n}\sum_{i=1}^{n} z_i - \mathbb{E}z\right)^2 = \frac{1}{2n}\operatorname{var}(z) \end{aligned}$$

- Bound only at $\hat{\theta}$ + strong convexity (instead of uniform bound)

Fast rate for supervised learning

- **Assumptions** (f is the expected risk, $\hat{f}$ the empirical risk)
  - Same as before (bounded features, Lipschitz loss)
  - Regularized risks: $f^\mu(\theta) = f(\theta) + \frac{\mu}{2}\|\theta\|_2^2$ and $\hat{f}^\mu(\theta) = \hat{f}(\theta) + \frac{\mu}{2}\|\theta\|_2^2$
  - Convexity
- For any $a > 0$, with probability greater than $1 - \delta$, for all $\theta \in \mathbb{R}^d$,

$$f^\mu(\hat{\theta}) - \min_{\eta \in \mathbb{R}^d} f^\mu(\eta) \leqslant \frac{8(1 + \frac{1}{a})G^2R^2(32 + \log\frac{1}{\delta})}{\mu n}$$

- Results from Sridharan, Srebro, and Shalev-Shwartz (2008)
  - see also Boucheron and Massart (2011) and references therein
- **Strongly convex functions $\Rightarrow$ fast rate**
  - Warning: μ should decrease with n to reduce approximation error

Outline - I

1. **Introduction**

   - Large-scale machine learning and optimization
   - Classes of functions (convex, smooth, etc.)
   - Traditional statistical analysis through Rademacher complexity

2. **Classical methods for convex optimization**

   - Smooth optimization (gradient descent, Newton method)
   - Non-smooth optimization (subgradient descent)

3. **Classical stochastic approximation** (not covered)

   - Robbins-Monro algorithm (1951)

Outline - II

4. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds

5. **Smooth stochastic approximation algorithms**
   - Non-asymptotic analysis for smooth functions
   - Least-squares regression without decaying step-sizes

6. **Finite data sets**
   - Gradient methods with exponential convergence rates

Complexity results in convex optimization

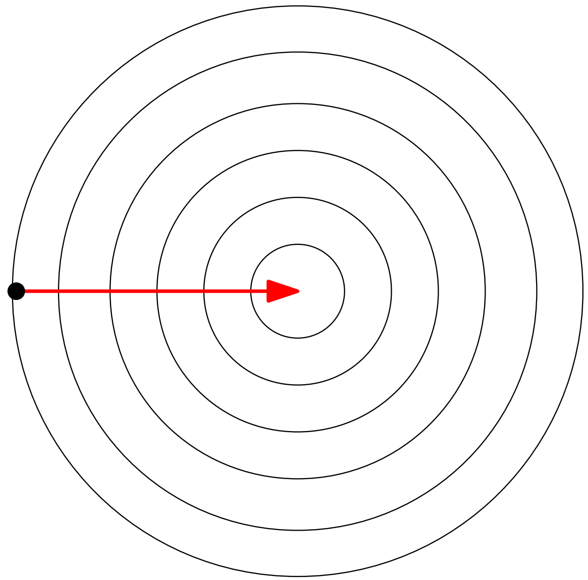
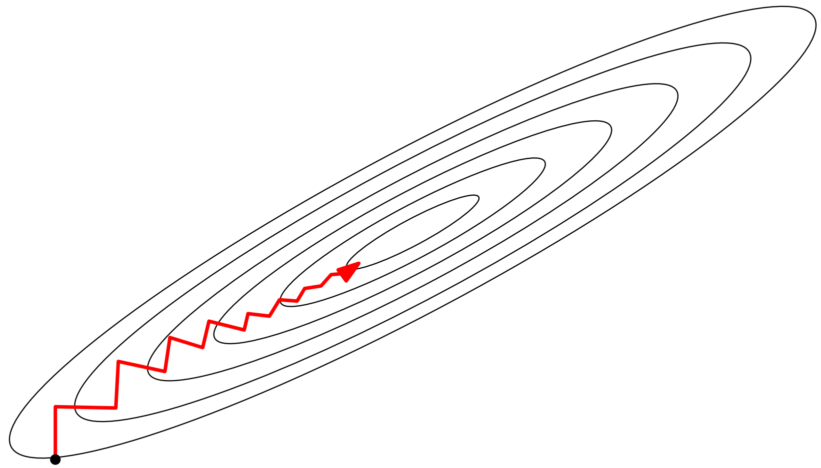
- **Assumption**: g convex on $\mathbb{R}^d$
- **Classical generic algorithms**
  - Gradient descent and accelerated gradient descent
  - Newton method
  - Subgradient method and ellipsoid algorithm
- **Key additional properties of g**
  - Lipschitz continuity, smoothness or strong convexity
- **Key insight from Bottou and Bousquet (2008)**
  - In machine learning, no need to optimize below estimation error
- **Key references**: Nesterov (2004), Bubeck (2015)

(smooth) gradient descent

- **Assumptions**
  - g convex with L-Lipschitz-continuous gradient (e.g., L-smooth)

- **Algorithm**:

$$\theta_t = \theta_{t-1} - \frac{1}{L} g'(\theta_{t-1})$$

(smooth) gradient descent - strong convexity

- **Assumptions**
  - g convex with L-Lipschitz-continuous gradient (e.g., L-smooth)
  - g μ-strongly convex
- **Algorithm**:
$$\theta_t = \theta_{t-1} - \frac{1}{L} g'(\theta_{t-1})$$
- **Bound**:
$$g(\theta_t) - g(\theta_*) \leqslant (1 - \mu/L)^t \big[g(\theta_0) - g(\theta_*) \big]$$
- Three-line proof
- **Line search, steepest descent or constant step-size**

(smooth) gradient descent - slow rate

- **Assumptions**
  - g convex with L-Lipschitz-continuous gradient (e.g., L-smooth)
  - Minimum attained at θ_*

- **Algorithm**:

$$\theta_t = \theta_{t-1} - \frac{1}{L} g'(\theta_{t-1})$$

- **Bound**:

$$g(\theta_t) - g(\theta_*) \leqslant \frac{2L\|\theta_0 - \theta_*\|^2}{t+4}$$

- Four-line proof

- **Adaptivity of gradient descent to problem difficulty**

- Not best possible convergence rates after $O(d)$ iterations

Gradient descent - Proof for quadratic functions

- Quadratic convex function: $g(\theta) = \frac{1}{2}\theta^\top H\theta - c^\top\theta$
  - μ and L are smallest largest eigenvalues of H
  - Global optimum $\theta_* = H^{-1}c$ (or $H^\dagger c$)

- Gradient descent:

$$\begin{aligned}
\theta_t &= \theta_{t-1} - \frac{1}{L}(H\theta - c) = \theta_{t-1} - \frac{1}{L}(H\theta - H\theta_*) \\
\theta_t - \theta_* &= (I - \frac{1}{L}H)(\theta_{t-1} - \theta_*) = (I - \frac{1}{L}H)^t(\theta_0 - \theta_*)
\end{aligned}$$

- **Strong convexity** $\mu > 0$: eigenvalues of $(I - \frac{1}{L}H)^t$ in $[0, (1 - \frac{\mu}{L})^t]$
  - Convergence of iterates: $\|\theta_t - \theta_*\|^2 \leqslant (1 - \mu/L)^{2t}\|\theta_0 - \theta_*\|^2$
  - Function values: $g(\theta_t) - g(\theta_*) \leqslant (1 - \mu/L)^{2t}\big[g(\theta_0) - g(\theta_*)\big]$

Gradient descent - Proof for quadratic functions

- Quadratic convex function: $g(\theta) = \frac{1}{2}\theta^\top H\theta - c^\top\theta$
  - μ and L are smallest largest eigenvalues of H
  - Global optimum $\theta_* = H^{-1}c$ (or $H^\dagger c$)

- Gradient descent:

$$\begin{aligned}\theta_t &= \theta_{t-1} - \frac{1}{L}(H\theta - c) = \theta_{t-1} - \frac{1}{L}(H\theta - H\theta_*)\\ \theta_t - \theta_* &= (I - \frac{1}{L}H)(\theta_{t-1} - \theta_*) = (I - \frac{1}{L}H)^t(\theta_0 - \theta_*)\end{aligned}$$

- **Convexity** $\mu = 0$: eigenvalues of $(I - \frac{1}{L}H)^t$ in $[0, 1]$
  - No convergence of iterates: $\|\theta_t - \theta_*\|^2 \leqslant \|\theta_0 - \theta_*\|^2$
  - Function values: $g(\theta_t) - g(\theta_*) \leqslant \max_{v\in[0,L]} v(1 - v/L)^{2t}\|\theta_0 - \theta_*\|^2$
    $$g(\theta_t) - g(\theta_*) \leqslant \frac{L}{t}\|\theta_0 - \theta_*\|^2$$

Accelerated gradient methods (Nesterov, 1983)

- **Assumptions**
  - g convex with L-Lipschitz-cont. gradient , min. attained at θ_*
- **Algorithm**:

$$\begin{aligned}\theta_t &= \eta_{t-1} - \frac{1}{L} g'(\eta_{t-1}) \\ \eta_t &= \theta_t + \frac{t-1}{t+2}(\theta_t - \theta_{t-1})\end{aligned}$$

- **Bound**:

$$g(\theta_t) - g(\theta_*) \leqslant \frac{2L\|\theta_0 - \theta_*\|^2}{(t+1)^2}$$

- Ten-line proof (see, e.g., Schmidt, Le Roux, and Bach, 2011)
- Not improvable
- Extension to strongly-convex functions

Accelerated gradient methods - strong convexity

- **Assumptions**
  - g convex with L-Lipschitz-cont. gradient , min. attained at θ_*
  - g μ-strongly convex
- **Algorithm**:

$$\begin{aligned} \theta_t &= \eta_{t-1} - \frac{1}{L} g'(\eta_{t-1}) \\ \eta_t &= \theta_t + \frac{1-\sqrt{\mu/L}}{1+\sqrt{\mu/L}}(\theta_t - \theta_{t-1}) \end{aligned}$$

- **Bound**: $g(\theta_t) - f(\theta_*) \leqslant L\|\theta_0 - \theta_*\|^2 (1 - \sqrt{\mu/L})^t$
  - Ten-line proof (see, e.g., Schmidt, Le Roux, and Bach, 2011)
  - Not improvable
  - Relationship with conjugate gradient for quadratic functions

Optimization for sparsity-inducing norms
(see Bach, Jenatton, Mairal, and Obozinski, 2012b)

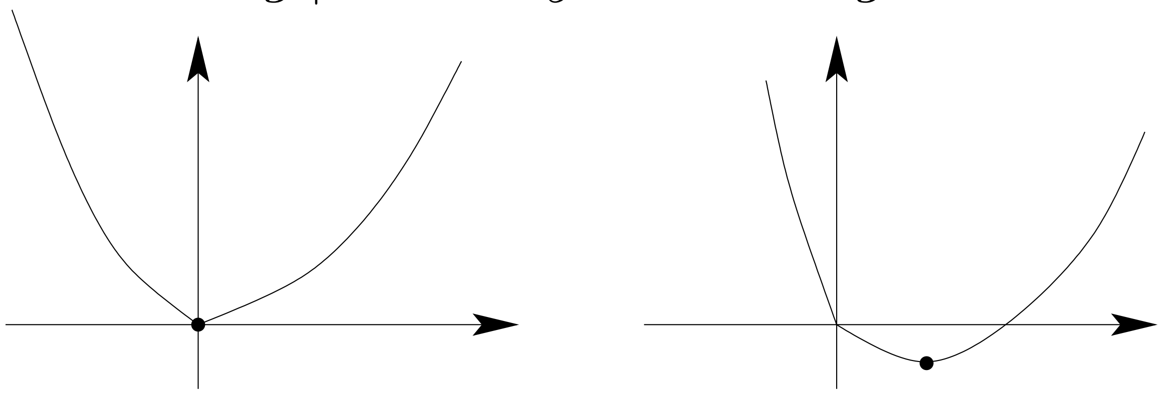
- Gradient descent as a **proximal method** (differentiable functions)
  - $\theta_{t+1} = \arg\min_{\theta \in \mathbb{R}^d} f(\theta_t) + (\theta - \theta_t)^\top \nabla f(\theta_t) + \frac{L}{2} \|\theta - \theta_t\|_2^2$
  - $\theta_{t+1} = \theta_t - \frac{1}{L} \nabla f(\theta_t)$

Optimization for sparsity-inducing norms (see Bach, Jenatton, Mairal, and Obozinski, 2012b)

- Gradient descent as a **proximal method** (differentiable functions)
  - $\theta_{t+1} = \arg\min_{\theta \in \mathbb{R}^d} f(\theta_t) + (\theta - \theta_t)^\top \nabla f(\theta_t) + \frac{L}{2}\|\theta - \theta_t\|_2^2$
  - $\theta_{t+1} = \theta_t - \frac{1}{L}\nabla f(\theta_t)$
- Problems of the form: $\boxed{\min_{\theta \in \mathbb{R}^d} f(\theta) + \mu\Omega(\theta)}$
  - $\theta_{t+1} = \arg\min_{\theta \in \mathbb{R}^d} f(\theta_t) + (\theta - \theta_t)^\top \nabla f(\theta_t) + \mu\Omega(\theta) + \frac{L}{2}\|\theta - \theta_t\|_2^2$
  - $\Omega(\theta) = \|\theta\|_1 \Rightarrow$ **Thresholded gradient descent**
- Similar convergence rates than smooth optimization
  - Acceleration methods (Nesterov, 2007; Beck and Teboulle, 2009)

Soft-thresholding for the ℓ_1-norm

- **Example 1**: quadratic problem in 1D, i.e. $\boxed{\min_{x\in\mathbb{R}} \frac{1}{2}x^2 - xy + \lambda|x|}$
- Piecewise quadratic function with a kink at zero
  - Derivative at $0+$: $g_+ = \lambda - y$ and $0-$: $g_- = -\lambda - y$

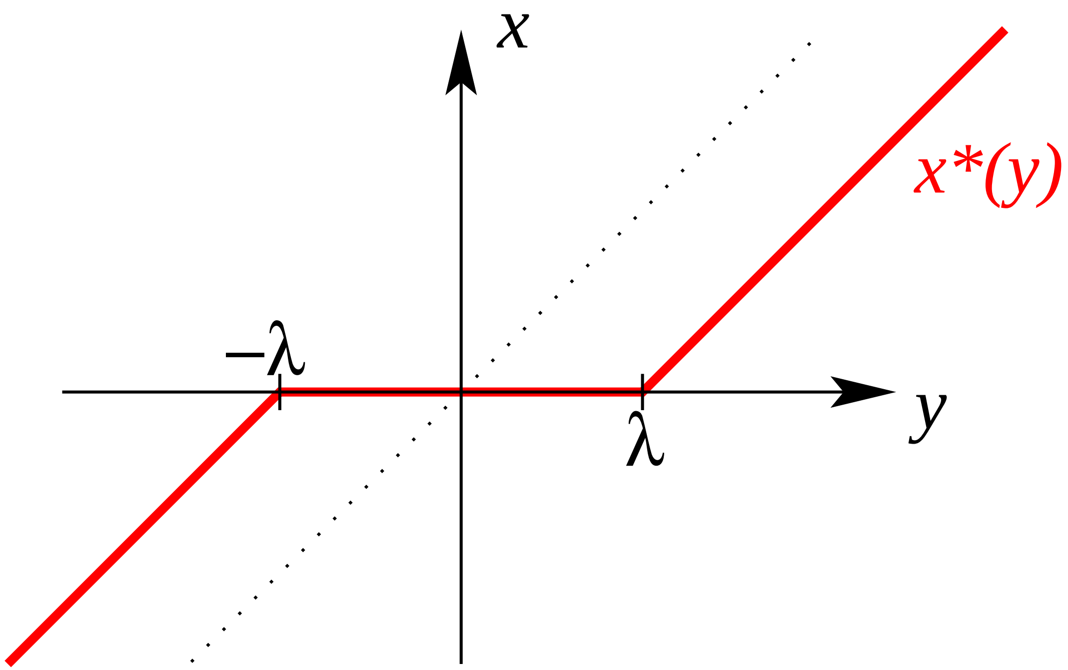


  - $x = 0$ is the solution iff $g_+ \geqslant 0$ and $g_- \leqslant 0$ (i.e., $|y| \leqslant \lambda$)
  - $x \geqslant 0$ is the solution iff $g_+ \leqslant 0$ (i.e., $y \geqslant \lambda$) $\Rightarrow x^* = y - \lambda$
  - $x \leqslant 0$ is the solution iff $g_- \leqslant 0$ (i.e., $y \leqslant -\lambda$) $\Rightarrow x^* = y + \lambda$

- Solution $\boxed{x^* = \text{sign}(y)(|y| - \lambda)_+}$ = soft thresholding

Soft-thresholding for the ℓ_1-norm

- **Example 1**: quadratic problem in 1D, i.e. $\boxed{\min_{x \in \mathbb{R}} \frac{1}{2}x^2 - xy + \lambda|x|}$
- Piecewise quadratic function with a kink at zero
- Solution $\boxed{x^* = \operatorname{sign}(y)(|y| - \lambda)_+}$ = soft thresholding



Newton method

- Given θ_{t-1}, minimize second-order Taylor expansion

$$\tilde{g}(\theta) = g(\theta_{t-1}) + g'(\theta_{t-1})^\top(\theta - \theta_{t-1}) + \frac{1}{2}(\theta - \theta_{t-1})^\top g''(\theta_{t-1})^\top(\theta - \theta_{t-1})$$

- **Expensive Iteration**: $\theta_t = \theta_{t-1} - g''(\theta_{t-1})^{-1} g'(\theta_{t-1})$
  - Running-time complexity: $O(d^3)$ in general

- **Quadratic convergence**: If $\|\theta_{t-1} - \theta_*\|$ small enough, for some constant C, we have

$$(C\|\theta_t - \theta_*\|) = (C\|\theta_{t-1} - \theta_*\|)^2$$

  - See Boyd and Vandenberghe (2003)

Summary: minimizing smooth convex functions

- **Assumption**: g convex
- **Gradient descent**: $\theta_t = \theta_{t-1} - \gamma_t \, g'(\theta_{t-1})$
  - $O(1/t)$ convergence rate for smooth convex functions
  - $O(e^{-t\mu/L})$ convergence rate for strongly smooth convex functions
  - Optimal rates $O(1/t^2)$ and $O(e^{-t\sqrt{\mu/L}})$
- **Newton method**: $\theta_t = \theta_{t-1} - f''(\theta_{t-1})^{-1} f'(\theta_{t-1})$
  - $O\big(e^{-\rho 2^t}\big)$ convergence rate

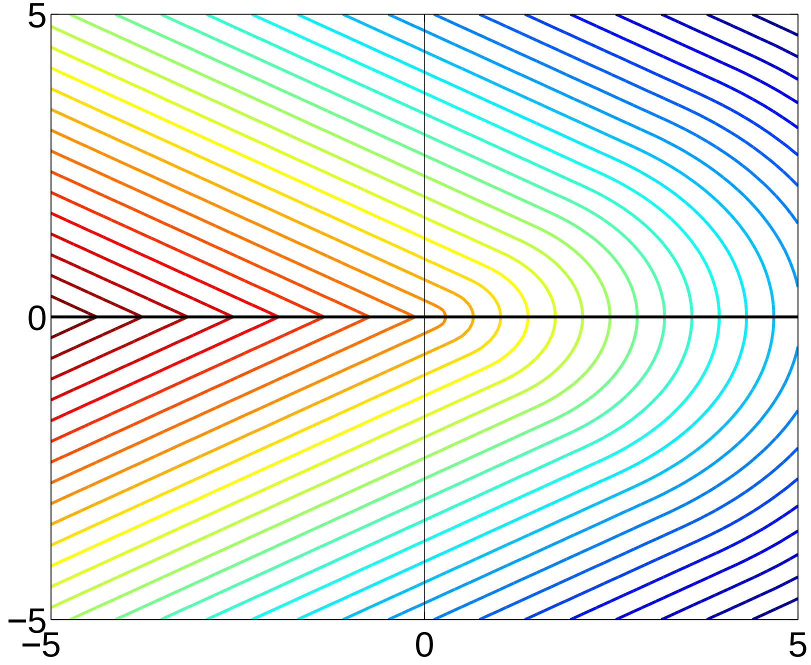
Summary: minimizing smooth convex functions

- **Assumption**: g convex
- **Gradient descent**: $\theta_t = \theta_{t-1} - \gamma_t \, g'(\theta_{t-1})$
  - $O(1/t)$ convergence rate for smooth convex functions
  - $O(e^{-t\mu/L})$ convergence rate for strongly smooth convex functions
  - Optimal rates $O(1/t^2)$ and $O(e^{-t\sqrt{\mu/L}})$
- **Newton method**: $\theta_t = \theta_{t-1} - f''(\theta_{t-1})^{-1} f'(\theta_{t-1})$
  - $O\big(e^{-\rho 2^t}\big)$ convergence rate
- **From smooth to non-smooth**
  - Subgradient method and ellipsoid (not covered)

Counter-example (Bertsekas, 1999)
Steepest descent for nonsmooth objectives

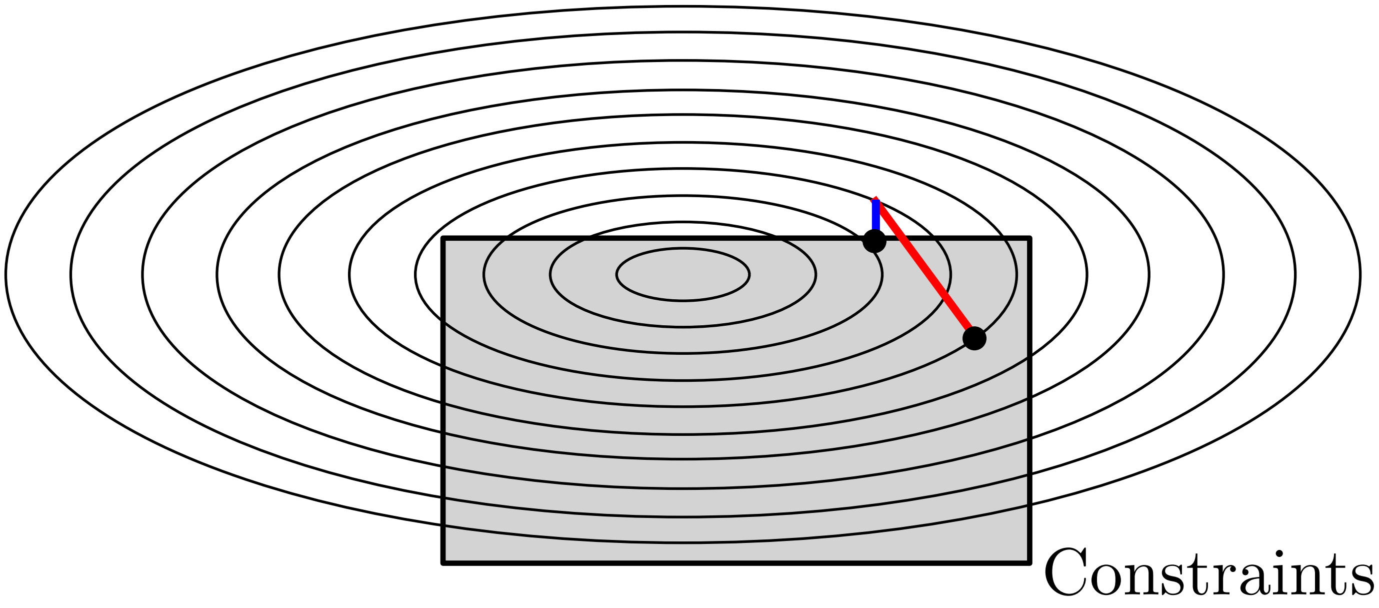
- $g(\theta_1, \theta_2) = \begin{cases} -5(9\theta_1^2 + 16\theta_2^2)^{1/2} & \text{if } \theta_1 > |\theta_2| \\ -(9\theta_1 + 16|\theta_2|)^{1/2} & \text{if } \theta_1 \leqslant |\theta_2| \end{cases}$

- Steepest descent starting from any θ such that $\theta_1 > |\theta_2| > (9/16)^2|\theta_1|$



Subgradient method/"descent" (Shor et al., 1985)

- **Assumptions**
  - g convex and B-Lipschitz-continuous on $\{\|\theta\|_2 \leqslant D\}$
- **Algorithm**: $\theta_t = \Pi_D\Big(\theta_{t-1} - \frac{2D}{B\sqrt{t}} g'(\theta_{t-1})\Big)$
  - Π_D : orthogonal projection onto $\{\|\theta\|_2 \leqslant D\}$



Subgradient method/“descent” (Shor et al., 1985)

- **Assumptions**
  - g convex and B-Lipschitz-continuous on $\{\|\theta\|_2 \leqslant D\}$
- **Algorithm**: $\theta_t = \Pi_D\Big(\theta_{t-1} - \frac{2D}{B\sqrt{t}} g'(\theta_{t-1})\Big)$
  - Π_D : orthogonal projection onto $\{\|\theta\|_2 \leqslant D\}$
- **Bound**:

$$g\bigg(\frac{1}{t}\sum_{k=0}^{t-1}\theta_k\bigg) - g(\theta_*) \leqslant \frac{2DB}{\sqrt{t}}$$

- Three-line proof
- Best possible convergence rate after $O(d)$ iterations (Bubeck, 2015)

Subgradient method/“descent” - proof - I

- Iteration: $\theta_t = \Pi_D(\theta_{t-1} - \gamma_t g'(\theta_{t-1}))$ with $\gamma_t = \frac{2D}{B\sqrt{t}}$
- Assumption: $\|g'(\theta)\|_2 \leqslant B$ and $\|\theta\|_2 \leqslant D$

$$\begin{aligned}
\|\theta_t - \theta_*\|_2^2 &\leqslant \|\theta_{t-1} - \theta_* - \gamma_t g'(\theta_{t-1})\|_2^2 \text{ by contractivity of projections} \\
&\leqslant \|\theta_{t-1} - \theta_*\|_2^2 + B^2\gamma_t^2 - 2\gamma_t(\theta_{t-1} - \theta_*)^\top g'(\theta_{t-1}) \text{ because } \|g'(\theta_{t-1})\|_2 \leqslant B \\
&\leqslant \|\theta_{t-1} - \theta_*\|_2^2 + B^2\gamma_t^2 - 2\gamma_t\big[g(\theta_{t-1}) - g(\theta_*)\big] \text{ (property of subgradients)}
\end{aligned}$$

- leading to

$$g(\theta_{t-1}) - g(\theta_*) \leqslant \frac{B^2\gamma_t}{2} + \frac{1}{2\gamma_t}\big[\|\theta_{t-1} - \theta_*\|_2^2 - \|\theta_t - \theta_*\|_2^2\big]$$

Subgradient method/“descent” - proof - II

- Starting from $$g(\theta_{t-1}) - g(\theta_*) \leqslant \frac{B^2\gamma_t}{2} + \frac{1}{2\gamma_t}\big[\|\theta_{t-1}-\theta_*\|_2^2 - \|\theta_t-\theta_*\|_2^2\big]$$

- Constant step-size $\gamma_t = \gamma$

$$\begin{aligned}\sum_{u=1}^t \big[g(\theta_{u-1}) - g(\theta_*)\big] &\leqslant \sum_{u=1}^t \frac{B^2\gamma}{2} + \sum_{u=1}^t \frac{1}{2\gamma}\big[\|\theta_{u-1}-\theta_*\|_2^2 - \|\theta_u-\theta_*\|_2^2\big] \\ &\leqslant t\frac{B^2\gamma}{2} + \frac{1}{2\gamma}\|\theta_0-\theta_*\|_2^2 \leqslant t\frac{B^2\gamma}{2} + \frac{2}{\gamma}D^2\end{aligned}$$

- Optimized step-size $\gamma_t = \frac{2D}{B\sqrt{t}}$ depends on “horizon”
  - Leads to bound of $2DB\sqrt{t}$

- Using convexity: $$g\bigg(\frac{1}{t}\sum_{k=0}^{t-1}\theta_k\bigg) - g(\theta_*) \leqslant \frac{2DB}{\sqrt{t}}$$

Subgradient method/"descent" - proof - III

- Starting from $\quad g(\theta_{t-1}) - g(\theta_*) \leqslant \frac{B^2\gamma_t}{2} + \frac{1}{2\gamma_t}\big[\|\theta_{t-1}-\theta_*\|_2^2 - \|\theta_t-\theta_*\|_2^2\big]$

- Decreasing step-size

$$
\begin{aligned}
\sum_{u=1}^{t}\big[g(\theta_{u-1}) - g(\theta_*)\big] &\leqslant \sum_{u=1}^{t}\frac{B^2\gamma_u}{2} + \sum_{u=1}^{t}\frac{1}{2\gamma_u}\big[\|\theta_{u-1}-\theta_*\|_2^2 - \|\theta_u-\theta_*\|_2^2\big] \\
&= \sum_{u=1}^{t}\frac{B^2\gamma_u}{2} + \sum_{u=1}^{t-1}\|\theta_u-\theta_*\|_2^2\Big(\frac{1}{2\gamma_{u+1}} - \frac{1}{2\gamma_u}\Big) + \frac{\|\theta_0-\theta_*\|_2^2}{2\gamma_1} - \frac{\|\theta_t-\theta_*\|_2^2}{2\gamma_t} \\
&\leqslant \sum_{u=1}^{t}\frac{B^2\gamma_u}{2} + \sum_{u=1}^{t-1}4D^2\Big(\frac{1}{2\gamma_{u+1}} - \frac{1}{2\gamma_u}\Big) + \frac{4D^2}{2\gamma_1} \\
&= \sum_{u=1}^{t}\frac{B^2\gamma_u}{2} + \frac{4D^2}{2\gamma_t} \leqslant 2DB\sqrt{t} \text{ with } \gamma_t = \frac{2D}{B\sqrt{t}}
\end{aligned}
$$

- Using convexity: $\quad g\Big(\frac{1}{t}\sum_{k=0}^{t-1}\theta_k\Big) - g(\theta_*) \leqslant \frac{2DB}{\sqrt{t}}$

Subgradient descent for machine learning

- **Assumptions** (f is the expected risk, $\hat{f}$ the empirical risk)
  - "Linear" predictors: $\theta(x) = \theta^\top \Phi(x)$, with $\|\Phi(x)\|_2 \leqslant R$ a.s.
  - $\hat{f}(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \Phi(x_i)^\top \theta)$
  - G-Lipschitz loss: f and $\hat{f}$ are GR-Lipschitz on $\Theta = \{\|\theta\|_2 \leqslant D\}$

- **Statistics**: with probability greater than $1-\delta$

$$\sup_{\theta \in \Theta} |\hat{f}(\theta) - f(\theta)| \leqslant \frac{GRD}{\sqrt{n}} \left[2 + \sqrt{2 \log \frac{2}{\delta}} \right]$$

- **Optimization**: after t iterations of subgradient method

$$\hat{f}(\hat{\theta}) - \min_{\eta \in \Theta} \hat{f}(\eta) \leqslant \frac{GRD}{\sqrt{t}}$$

- $t = n$ iterations, with total running-time complexity of $O(n^2 d)$

Subgradient descent - strong convexity

- **Assumptions**
  - g convex and B-Lipschitz-continuous on $\{\|\theta\|_2 \leqslant D\}$
  - g μ-strongly convex

- **Algorithm**: $\theta_t = \Pi_D\left(\theta_{t-1} - \frac{2}{\mu(t+1)} g'(\theta_{t-1})\right)$

- **Bound**:

$$g\left(\frac{2}{t(t+1)} \sum_{k=1}^{t} k\theta_{k-1}\right) - g(\theta_*) \leqslant \frac{2B^2}{\mu(t+1)}$$

- Three-line proof

- Best possible convergence rate after $O(d)$ iterations (Bubeck, 2015)

Subgradient method - strong convexity - proof - I

- Iteration: $\theta_t = \Pi_D(\theta_{t-1} - \gamma_t g'(\theta_{t-1}))$ with $\gamma_t = \frac{2}{\mu(t+1)}$

- Assumption: $\|g'(\theta)\|_2 \leqslant B$ and $\|\theta\|_2 \leqslant D$ and μ-strong convexity of f

$$\begin{aligned}
\|\theta_t - \theta_*\|_2^2 &\leqslant \|\theta_{t-1} - \theta_* - \gamma_t g'(\theta_{t-1})\|_2^2 \text{ by contractivity of projections} \\
&\leqslant \|\theta_{t-1} - \theta_*\|_2^2 + B^2\gamma_t^2 - 2\gamma_t(\theta_{t-1} - \theta_*)^\top g'(\theta_{t-1}) \text{ because } \|g'(\theta_{t-1})\|_2 \leqslant B \\
&\leqslant \|\theta_{t-1} - \theta_*\|_2^2 + B^2\gamma_t^2 - 2\gamma_t\big[g(\theta_{t-1}) - g(\theta_*) + \frac{\mu}{2}\|\theta_{t-1} - \theta_*\|_2^2\big] \\
&\qquad \text{(property of subgradients and strong convexity)}
\end{aligned}$$

- leading to

$$\begin{aligned}
g(\theta_{t-1}) - g(\theta_*) &\leqslant \frac{B^2\gamma_t}{2} + \frac{1}{2}\Big[\frac{1}{\gamma_t} - \mu\Big]\|\theta_{t-1} - \theta_*\|_2^2 - \frac{1}{2\gamma_t}\|\theta_t - \theta_*\|_2^2 \\
&\leqslant \frac{B^2}{\mu(t+1)} + \frac{\mu}{2}\Big[\frac{t-1}{2}\Big]\|\theta_{t-1} - \theta_*\|_2^2 - \frac{\mu(t+1)}{4}\|\theta_t - \theta_*\|_2^2
\end{aligned}$$

Subgradient method - strong convexity - proof - II

- From $g(\theta_{t-1}) - g(\theta_*) \leqslant \dfrac{B^2}{\mu(t+1)} + \dfrac{\mu}{2}\big[\dfrac{t-1}{2}\big]\|\theta_{t-1} - \theta_*\|_2^2 - \dfrac{\mu(t+1)}{4}\|\theta_t - \theta_*\|_2^2$

$$\sum_{u=1}^{t} u\big[g(\theta_{u-1}) - g(\theta_*)\big] \leqslant \sum_{t=1}^{u} \frac{B^2 u}{\mu(u+1)} + \frac{1}{4}\sum_{u=1}^{t}\big[u(u-1)\|\theta_{u-1} - \theta_*\|_2^2 - u(u+1)\|\theta_u - \theta_*\|_2^2\big]$$
$$\leqslant \quad \frac{B^2 t}{\mu} + \frac{1}{4}\big[0 - t(t+1)\|\theta_t - \theta_*\|_2^2\big] \leqslant \frac{B^2 t}{\mu}$$

- Using convexity: $g\bigg(\dfrac{2}{t(t+1)}\displaystyle\sum_{u=1}^{t} u\theta_{u-1}\bigg) - g(\theta_*) \leqslant \dfrac{2B^2}{t+1}$

- NB: with step-size $\gamma_n = 1/(n\mu)$, extra logarithmic factor

Summary: minimizing convex functions

- **Assumption**: g convex

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

Summary: minimizing convex functions

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

Summary of rates of convergence

- Problem parameters
  - D diameter of the domain
  - B Lipschitz-constant
  - L smoothness constant
  - μ strong convexity constant

	convex	strongly convex
nonsmooth	deterministic: $BD/\sqrt{t}$	deterministic: $B^2/(t\mu)$
smooth	deterministic: LD^2/t^2	deterministic: $\exp(-t\sqrt{\mu/L})$
quadratic	deterministic: LD^2/t^2	deterministic: $\exp(-t\sqrt{\mu/L})$

Outline - I

1. **Introduction**
   - Large-scale machine learning and optimization
   - Classes of functions (convex, smooth, etc.)
   - Traditional statistical analysis through Rademacher complexity
2. **Classical methods for convex optimization**
   - Smooth optimization (gradient descent, Newton method)
   - Non-smooth optimization (subgradient descent)
3. **Classical stochastic approximation** (not covered)
   - Robbins-Monro algorithm (1951)

Outline - II

4. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds

5. **Smooth stochastic approximation algorithms**
   - Non-asymptotic analysis for smooth functions
   - Least-squares regression without decaying step-sizes

6. **Finite data sets**
   - Gradient methods with exponential convergence rates

Stochastic approximation

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

Stochastic approximation

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

- **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: $f'(\theta) = \mathbb{E} f_n'(\theta) = \mathbb{E}\left\{\ell'(y_n, \theta^\top \Phi(x_n))\, \Phi(x_n)\right\}$
  - Non-asymptotic results

- **Number of iterations = number of observations**

Stochastic approximation

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

- **Stochastic approximation**
  - (much) broader applicability beyond convex optimization

$$\theta_n = \theta_{n-1} - \gamma_n h_n(\theta_{n-1}) \text{ with } \mathbb{E}\big[h_n(\theta_{n-1})|\theta_{n-1}\big] = h(\theta_{n-1})$$

  - Beyond convex problems, i.i.d assumption, finite dimension, etc.
  - Typically asymptotic results
  - See, e.g., Kushner and Yin (2003); Benveniste et al. (2012)

Relationship to online learning

- **Stochastic approximation**
  - Minimize $f(\theta) = \mathbb{E}_z \ell(\theta, z)$ = **generalization error** of θ
  - Using the gradients of single i.i.d. observations

Relationship to online learning

- **Stochastic approximation**
  - Minimize $f(\theta) = \mathbb{E}_z \ell(\theta, z)$ = **generalization error** of θ
  - Using the gradients of single i.i.d. observations

- **Batch learning**
  - Finite set of observations: $z_1, \dots, z_n$
  - Empirical risk: $\hat{f}(\theta) = \frac{1}{n} \sum_{k=1}^{n} \ell(\theta, z_i)$
  - Estimator $\hat{\theta}$ = Minimizer of $\hat{f}(\theta)$ over a certain class Θ
  - Generalization bound using uniform concentration results

Relationship to online learning

- **Stochastic approximation**
  - Minimize $f(\theta) = \mathbb{E}_z \ell(\theta, z)$ = **generalization error** of θ
  - Using the gradients of single i.i.d. observations

- **Batch learning**
  - Finite set of observations: $z_1, \dots, z_n$
  - Empirical risk: $\hat{f}(\theta) = \frac{1}{n} \sum_{k=1}^{n} \ell(\theta, z_i)$
  - Estimator $\hat{\theta}$ = Minimizer of $\hat{f}(\theta)$ over a certain class Θ
  - Generalization bound using uniform concentration results

- **Online learning**
  - Update $\hat{\theta}_n$ after each new (potentially adversarial) observation z_n
  - Cumulative loss: $\frac{1}{n} \sum_{k=1}^{n} \ell(\hat{\theta}_{k-1}, z_k)$
  - Online to batch through averaging (Cesa-Bianchi et al., 2004)

Convex stochastic approximation

- **Key properties of f and/or f_n**
  - Smoothness: f B-Lipschitz continuous, f' L-Lipschitz continuous
  - Strong convexity: f μ-strongly convex

Convex stochastic approximation

- **Key properties of f and/or f_n**
  - Smoothness: f B-Lipschitz continuous, f' L-Lipschitz continuous
  - 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: $\boxed{\gamma_n = C n^{-\alpha}}$

Convex stochastic approximation

- **Key properties of f and/or f_n**
  - Smoothness: f B-Lipschitz continuous, f' L-Lipschitz continuous
  - 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 classical supervised learning problems
  - Robustness to (potentially unknown) constants (L,B,μ)
  - Adaptivity to difficulty of the problem (e.g., strong convexity)

Stochastic subgradient "descent"/method

- **Assumptions**
  - f_n convex and B-Lipschitz-continuous on $\{\|\theta\|_2 \leqslant D\}$
  - (f_n) i.i.d. functions such that $\mathbb{E} f_n = f$
  - θ_* global optimum of f on $\mathcal{C} = \{\|\theta\|_2 \leqslant D\}$

- **Algorithm**: $\theta_n = \Pi_D\bigg(\theta_{n-1} - \dfrac{2D}{B\sqrt{n}} f_n'(\theta_{n-1})\bigg)$

Stochastic subgradient "descent"/method

- **Assumptions**
  - f_n convex and B-Lipschitz-continuous on $\{\|\theta\|_2 \leqslant D\}$
  - (f_n) i.i.d. functions such that $\mathbb{E} f_n = f$
  - θ_* global optimum of f on $\mathcal{C} = \{\|\theta\|_2 \leqslant D\}$
- **Algorithm**: $\theta_n = \Pi_D\bigg(\theta_{n-1} - \dfrac{2D}{B\sqrt{n}} f_n'(\theta_{n-1})\bigg)$
- **Bound**:

$$\mathbb{E} f\bigg(\frac{1}{n}\sum_{k=0}^{n-1}\theta_k\bigg) - f(\theta_*) \leqslant \frac{2DB}{\sqrt{n}}$$

- "Same" three-line proof as in the deterministic case
- Minimax rate (Nemirovsky and Yudin, 1983; Agarwal et al., 2012)
- Running-time complexity: $O(dn)$ after n iterations

Stochastic subgradient method - proof - I

- Iteration: $\theta_n = \Pi_D(\theta_{n-1} - \gamma_n f_n'(\theta_{n-1}))$ with $\gamma_n = \frac{2D}{B\sqrt{n}}$

- $\mathcal{F}_n$: information up to time n

- $\|f_n'(\theta)\|_2 \leqslant B$ and $\|\theta\|_2 \leqslant D$, unbiased gradients/functions $\mathbb{E}(f_n|\mathcal{F}_{n-1}) = f$

$$
\begin{aligned}
\|\theta_n - \theta_*\|_2^2 &\leqslant \|\theta_{n-1} - \theta_* - \gamma_n f_n'(\theta_{n-1})\|_2^2 \text{ by contractivity of projections} \\
&\leqslant \|\theta_{n-1} - \theta_*\|_2^2 + B^2\gamma_n^2 - 2\gamma_n(\theta_{n-1} - \theta_*)^\top f_n'(\theta_{n-1}) \text{ because } \|f_n'(\theta_{n-1})\|_2 \leqslant B
\end{aligned}
$$

$$
\begin{aligned}
\mathbb{E}\big[\|\theta_n - \theta_*\|_2^2 | \mathcal{F}_{n-1}\big] &\leqslant \|\theta_{n-1} - \theta_*\|_2^2 + B^2\gamma_n^2 - 2\gamma_n(\theta_{n-1} - \theta_*)^\top f'(\theta_{n-1}) \\
&\leqslant \|\theta_{n-1} - \theta_*\|_2^2 + B^2\gamma_n^2 - 2\gamma_n\big[f(\theta_{n-1}) - f(\theta_*)\big] \text{ (subgradient property)} \\
\mathbb{E}\|\theta_n - \theta_*\|_2^2 &\leqslant \mathbb{E}\|\theta_{n-1} - \theta_*\|_2^2 + B^2\gamma_n^2 - 2\gamma_n\big[\mathbb{E}f(\theta_{n-1}) - f(\theta_*)\big]
\end{aligned}
$$

- leading to $\mathbb{E}f(\theta_{n-1}) - f(\theta_*) \leqslant \dfrac{B^2\gamma_n}{2} + \dfrac{1}{2\gamma_n}\big[\mathbb{E}\|\theta_{n-1} - \theta_*\|_2^2 - \mathbb{E}\|\theta_n - \theta_*\|_2^2\big]$

Stochastic subgradient method - proof - II

- Starting from $\mathbb{E}f(\theta_{n-1}) - f(\theta_*) \leqslant \frac{B^2\gamma_n}{2} + \frac{1}{2\gamma_n}\big[\mathbb{E}\|\theta_{n-1}-\theta_*\|_2^2 - \mathbb{E}\|\theta_n-\theta_*\|_2^2\big]$

$$\begin{aligned}\sum_{u=1}^{n}\big[\mathbb{E}f(\theta_{u-1}) - f(\theta_*)\big] &\leqslant \sum_{u=1}^{n}\frac{B^2\gamma_u}{2} + \sum_{u=1}^{n}\frac{1}{2\gamma_u}\big[\mathbb{E}\|\theta_{u-1}-\theta_*\|_2^2 - \mathbb{E}\|\theta_u-\theta_*\|_2^2\big] \\ &\leqslant \sum_{u=1}^{n}\frac{B^2\gamma_u}{2} + \frac{4D^2}{2\gamma_n} \leqslant 2DB\sqrt{n} \text{ with } \gamma_n = \frac{2D}{B\sqrt{n}}\end{aligned}$$

- Using convexity: $\mathbb{E}f\left(\frac{1}{n}\sum_{k=0}^{n-1}\theta_k\right) - f(\theta_*) \leqslant \frac{2DB}{\sqrt{n}}$

Stochastic subgradient descent - strong convexity - I

- **Assumptions**
  - f_n convex and B-Lipschitz-continuous
  - (f_n) i.i.d. functions such that $\mathbb{E}f_n = f$
  - f μ-strongly convex on $\{\|\theta\|_2 \leqslant D\}$
  - θ_* global optimum of f over $\{\|\theta\|_2 \leqslant D\}$

- **Algorithm**: $\theta_n = \Pi_D\bigg(\theta_{n-1} - \dfrac{2}{\mu(n+1)} f_n'(\theta_{n-1})\bigg)$

- **Bound**:

$$\mathbb{E}f\bigg(\frac{2}{n(n+1)}\sum_{k=1}^n k\theta_{k-1}\bigg) - f(\theta_*) \leqslant \frac{2B^2}{\mu(n+1)}$$

- "Same" proof than deterministic case (Lacoste-Julien et al., 2012)

- Minimax rate (Nemirovsky and Yudin, 1983; Agarwal et al., 2012)

Stochastic subgradient - strong convexity - proof - I

- Iteration: $\theta_n = \Pi_D(\theta_{n-1} - \gamma_n f_n'(\theta_{t-1}))$ with $\gamma_n = \frac{2}{\mu(n+1)}$

- Assumption: $\|f_n'(\theta)\|_2 \leqslant B$ and $\|\theta\|_2 \leqslant D$ and μ-strong convexity of f

$$\begin{aligned}
\|\theta_n - \theta_*\|_2^2 &\leqslant \|\theta_{n-1} - \theta_* - \gamma_n f_n'(\theta_{t-1})\|_2^2 \text{ by contractivity of projections} \\
&\leqslant \|\theta_{n-1} - \theta_*\|_2^2 + B^2\gamma_n^2 - 2\gamma_n(\theta_{n-1} - \theta_*)^\top f_n'(\theta_{t-1}) \text{ because } \|f_n'(\theta_{t-1})\|_2 \leqslant B \\
\mathbb{E}(\cdot|\mathcal{F}_{n-1}) &\leqslant \|\theta_{n-1} - \theta_*\|_2^2 + B^2\gamma_n^2 - 2\gamma_n\big[f(\theta_{n-1}) - f(\theta_*) + \frac{\mu}{2}\|\theta_{n-1} - \theta_*\|_2^2\big]
\end{aligned}$$

(property of subgradients and strong convexity)

- leading to

$$\begin{aligned}
\mathbb{E}f(\theta_{n-1}) - f(\theta_*) &\leqslant \frac{B^2\gamma_n}{2} + \frac{1}{2}\Big[\frac{1}{\gamma_n} - \mu\Big]\|\theta_{n-1} - \theta_*\|_2^2 - \frac{1}{2\gamma_n}\|\theta_n - \theta_*\|_2^2 \\
&\leqslant \frac{B^2}{\mu(n+1)} + \frac{\mu}{2}\Big[\frac{n-1}{2}\Big]\|\theta_{n-1} - \theta_*\|_2^2 - \frac{\mu(n+1)}{4}\|\theta_n - \theta_*\|_2^2
\end{aligned}$$

Stochastic subgradient - strong convexity - proof - II

- From $\mathbb{E} f(\theta_{n-1}) - f(\theta_*) \leqslant \frac{B^2}{\mu(n+1)} + \frac{\mu}{2}\left[\frac{n-1}{2}\right]\mathbb{E}\|\theta_{n-1}-\theta_*\|_2^2 - \frac{\mu(n+1)}{4}\mathbb{E}\|\theta_n - \theta_*\|_2^2$

$$\sum_{u=1}^{n} u\big[\mathbb{E} f(\theta_{u-1}) - f(\theta_*)\big] \leqslant \sum_{u=1}^{n} \frac{B^2 u}{\mu(u+1)} + \frac{1}{4}\sum_{u=1}^{n}\big[u(u-1)\mathbb{E}\|\theta_{u-1}-\theta_*\|_2^2 - u(u+1)\mathbb{E}\|\theta_u - \theta_*$$

$$\leqslant \quad \frac{B^2 n}{\mu} + \frac{1}{4}\big[0 - n(n+1)\mathbb{E}\|\theta_n - \theta_*\|_2^2\big] \leqslant \frac{B^2 n}{\mu}$$

- Using convexity: $\mathbb{E} f\left(\frac{2}{n(n+1)}\sum_{u=1}^{n} u\theta_{u-1}\right) - g(\theta_*) \leqslant \frac{2B^2}{n+1}$

- NB: with step-size $\gamma_n = 1/(n\mu)$, extra logarithmic factor (see later)

Stochastic subgradient descent - strong convexity - II

- **Assumptions**
  - f_n convex and B-Lipschitz-continuous
  - (f_n) i.i.d. functions such that $\mathbb{E} f_n = f$
  - θ_* global optimum of $g = f + \frac{\mu}{2}\| \cdot \|_2^2$
  - No compactness assumption - no projections
- **Algorithm**:

$$\theta_n = \theta_{n-1} - \frac{2}{\mu(n+1)} g_n'(\theta_{n-1}) = \theta_{n-1} - \frac{2}{\mu(n+1)} \big[f_n'(\theta_{n-1}) + \mu \theta_{n-1} \big]$$

- **Bound**: $\mathbb{E} g \left(\frac{2}{n(n+1)} \sum_{k=1}^{n} k \theta_{k-1} \right) - g(\theta_*) \leqslant \frac{2B^2}{\mu(n+1)}$
- Minimax convergence rate

Beyond convergence in expectation

- **Typical result**: $\mathbb{E} f\left(\frac{1}{n}\sum_{k=0}^{n-1}\theta_k\right) - f(\theta_*) \leqslant \frac{2DB}{\sqrt{n}}$
  - Obtained with simple conditioning arguments

- **High-probability bounds**
  - Markov inequality: $\mathbb{P}\left(f\left(\frac{1}{n}\sum_{k=0}^{n-1}\theta_k\right) - f(\theta_*) \geqslant \varepsilon\right) \leqslant \frac{2DB}{\sqrt{n}\varepsilon}$

Beyond convergence in expectation

- **Typical result**: $\mathbb{E} f\left(\frac{1}{n}\sum_{k=0}^{n-1}\theta_k\right) - f(\theta_*) \leqslant \frac{2DB}{\sqrt{n}}$
  - Obtained with simple conditioning arguments

- **High-probability bounds**
  - Markov inequality: $\mathbb{P}\Big(f\big(\frac{1}{n}\sum_{k=0}^{n-1}\theta_k\big) - f(\theta_*) \geqslant \varepsilon\Big) \leqslant \frac{2DB}{\sqrt{n}\varepsilon}$
  - Concentration inequality (Nemirovski et al., 2009; Nesterov and Vial, 2008)

$$\mathbb{P}\left(f\left(\frac{1}{n}\sum_{k=0}^{n-1}\theta_k\right) - f(\theta_*) \geqslant \frac{2DB}{\sqrt{n}}(2+4t)\right) \leqslant 2\exp(-t^2)$$

- See also Bach (2013) for logistic regression

Beyond stochastic gradient method

- **Adding a proximal step**
  - Goal: $\min_{\theta \in \mathbb{R}^d} f(\theta) + \Omega(\theta) = \mathbb{E} f_n(\theta) + \Omega(\theta)$
  - Replace recursion $\theta_n = \theta_{n-1} - \gamma_n f_n'(\theta_n)$ by

$$\theta_n = \min_{\theta \in \mathbb{R}^d} \big\| \theta - \theta_{n-1} + \gamma_n f_n'(\theta_n) \big\|_2^2 + C\Omega(\theta)$$

  - Xiao (2010); Hu et al. (2009)
  - May be accelerated (Ghadimi and Lan, 2013)

- **Related frameworks**
  - Regularized dual averaging (Nesterov, 2009; Xiao, 2010)
  - Mirror descent (Nemirovski et al., 2009; Lan et al., 2012)

Minimax rates (Agarwal et al., 2012)

- **Model of computation (i.e., algorithms): first-order oracle**
  - Queries a function f by obtaining $f(\theta_k)$ and $f'(\theta_k)$ with zero-mean bounded variance noise, for $k = 0, \dots, n-1$ and outputs θ_n

- **Class of functions**
  - convex B-Lipschitz-continuous (w.r.t. ℓ_2-norm) on a compact convex set $\mathcal{C}$ containing an ℓ_∞-ball

- **Performance measure**
  - for a given algorithm and function $\varepsilon_n(\text{algo}, f) = f(\theta_n) - \inf_{\theta \in \mathcal{C}} f(\theta)$
  - for a given algorithm: $\displaystyle \sup_{\text{functions } f} \varepsilon_n(\text{algo}, f)$

- **Minimax performance**: $\displaystyle \inf_{\text{algo}} \sup_{\text{functions } f} \varepsilon_n(\text{algo}, f)$

Minimax rates (Agarwal et al., 2012)

- **Convex functions**: domain $\mathcal{C}$ that contains an ℓ_∞-ball of radius D

$$\inf_{\text{algo}} \sup_{\text{functions } f} \varepsilon(\text{algo}, f) \geqslant \text{ cst } \times \min\left\{BD\sqrt{\frac{d}{n}}, BD\right\}$$

  - Consequences for ℓ_2-ball of radius D: $BD/\sqrt{n}$
  - Upper-bound through stochastic subgradient

- μ-**strongly-convex functions**:

$$\inf_{\text{algo}} \sup_{\text{functions } f} \varepsilon_n(\text{algo}, f) \geqslant \text{ cst } \times \min\left\{\frac{B^2}{\mu n}, \frac{B^2}{\mu d}, BD\sqrt{\frac{d}{n}}, BD\right\}$$

Summary of rates of convergence

- Problem parameters
  - D diameter of the domain
  - B Lipschitz-constant
  - L smoothness constant
  - μ strong convexity constant

	convex	strongly convex
nonsmooth	deterministic: $BD/\sqrt{t}$ stochastic: $BD/\sqrt{n}$	deterministic: $B^2/(t\mu)$ stochastic: $B^2/(n\mu)$
smooth	deterministic: LD^2/t^2	deterministic: $\exp(-t\sqrt{\mu/L})$
quadratic	deterministic: LD^2/t^2	deterministic: $\exp(-t\sqrt{\mu/L})$

Outline - I

1. **Introduction**
   - Large-scale machine learning and optimization
   - Classes of functions (convex, smooth, etc.)
   - Traditional statistical analysis through Rademacher complexity

2. **Classical methods for convex optimization**
   - Smooth optimization (gradient descent, Newton method)
   - Non-smooth optimization (subgradient descent)

3. **Classical stochastic approximation** (not covered)
   - Robbins-Monro algorithm (1951)

Outline - II

4. **Non-smooth stochastic approximation**
    - Stochastic (sub)gradient and averaging
    - Non-asymptotic results and lower bounds

5. **Smooth stochastic approximation algorithms**
    - Non-asymptotic analysis for smooth functions
    - Least-squares regression without decaying step-sizes

6. **Finite data sets**
    - Gradient methods with exponential convergence rates

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

- **Non-asymptotic analysis for smooth problems?**

Smoothness/convexity assumptions

- Iteration: $\boxed{\theta_n = \theta_{n-1} - \gamma_n f_n'(\theta_{n-1})}$
  - Polyak-Ruppert averaging: $\bar{\theta}_n = \frac{1}{n}\sum_{k=0}^{n-1}\theta_k$
- **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':
  - Smooth loss and bounded data
- **Strong convexity of f**: The function f is strongly convex with respect to the norm $\|\cdot\|$, with convexity constant $\mu > 0$:
  - Invertible population covariance matrix
  - or regularization by $\frac{\mu}{2}\|\theta\|^2$

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)

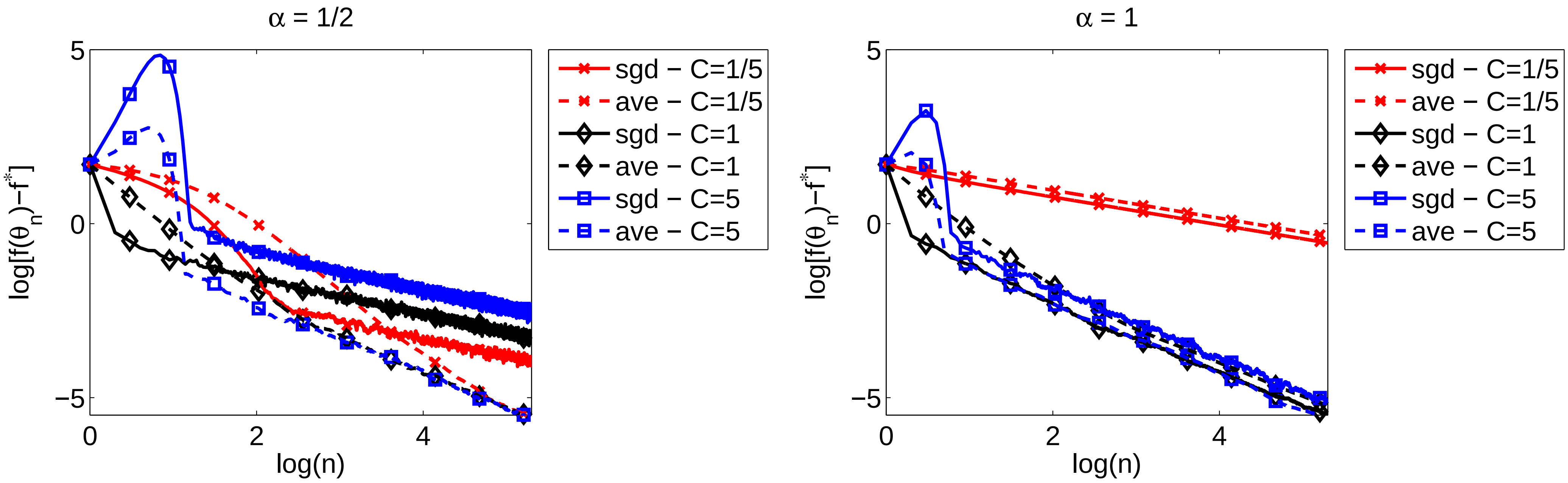
- Stochastic gradient descent with learning rate $\gamma_n = Cn^{-\alpha}$

- **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

- **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: $\frac{\operatorname{tr} H(\theta_*)^{-1}}{n} + \mu^{-1}O(n^{-2\alpha}+n^{-2+\alpha}) + O\Big(\frac{\|\theta_0 - \theta_*\|^2}{\mu^2 n^2}\Big)$

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

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



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

Summary of new results (Bach and Moulines, 2011)

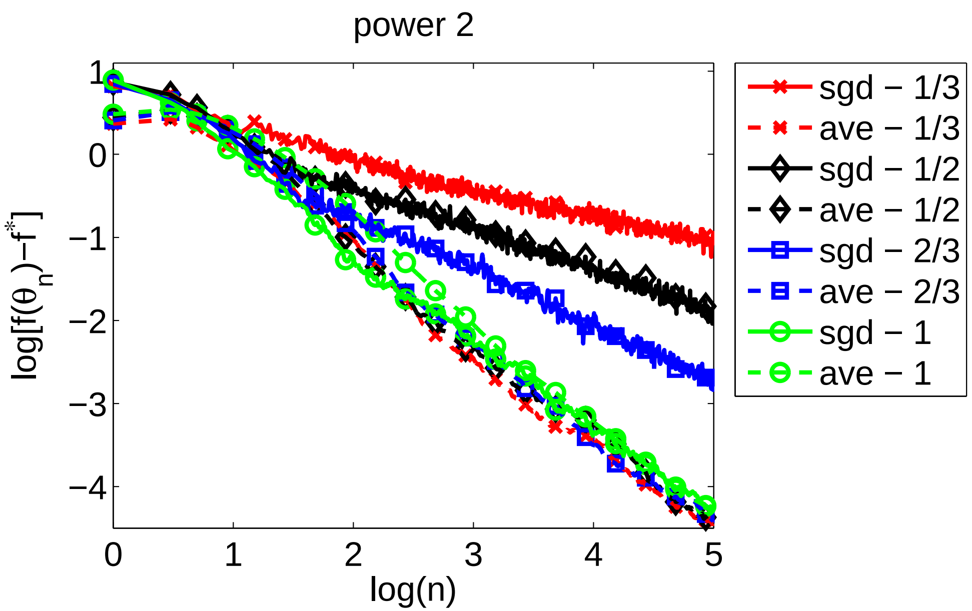
- Stochastic gradient descent with learning rate $\gamma_n = Cn^{-\alpha}$
- **Strongly convex smooth objective functions**
  - Old: $O(n^{-1})$ rate achieved without averaging for $\alpha = 1$
  - New: $O(n^{-1})$ rate achieved with averaging for $\alpha \in [1/2, 1]$
  - Non-asymptotic analysis with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

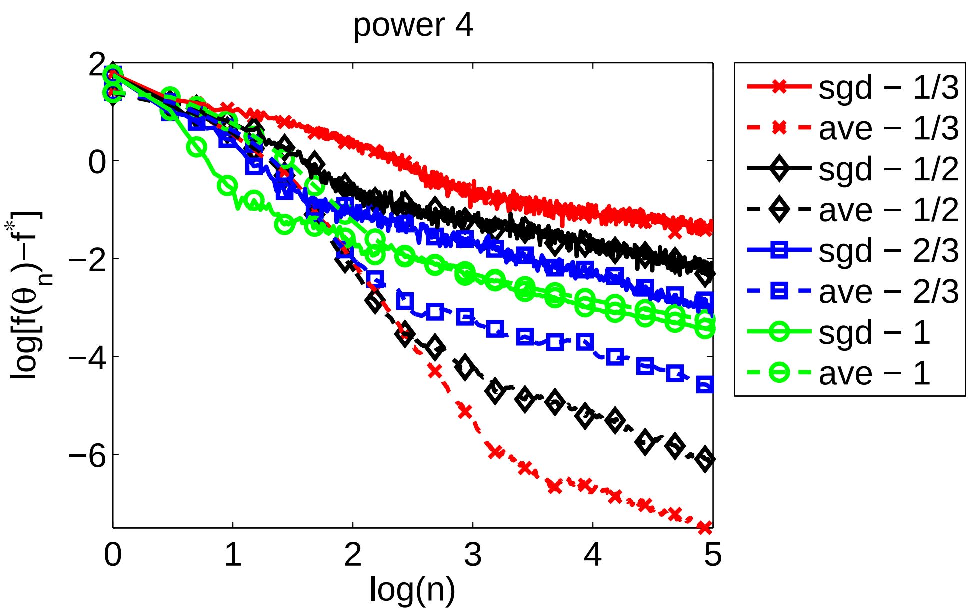
Summary of new results (Bach and Moulines, 2011)

- 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.





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 adaptive algorithm for smooth problems with convergence rate $O(\min\{1/\mu n, 1/\sqrt{n}\})$ in all situations?**

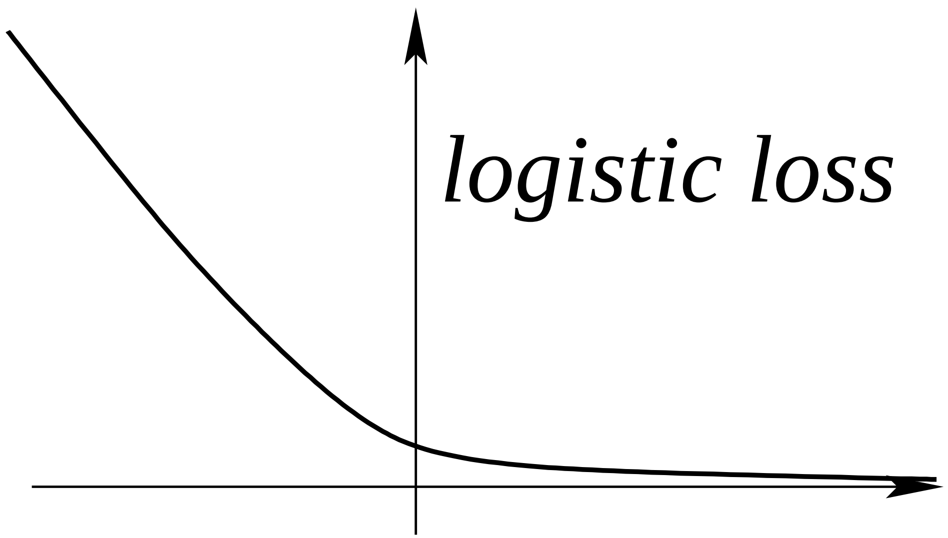
Adaptive algorithm for logistic regression

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

Adaptive algorithm for logistic regression

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

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



Adaptive algorithm for logistic regression

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

- **Cannot be strongly convex** $\Rightarrow$ local strong convexity
  - unless restricted to $|\theta^\top \Phi(x_n)| \leqslant M$ (with constants e^M - *proof*)
  - $\mu =$ 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 self-concordance (Nesterov and Nemirovski, 1994)

Self-concordance

- Usual definition for convex $\varphi : \mathbb{R} \to \mathbb{R}$: $|\varphi'''(t)| \leqslant 2\varphi''(t)^{3/2}$
  - Affine invariant
  - Extendable to all convex functions on $\mathbb{R}^d$ by looking at rays
  - Used for the sharp proof of quadratic convergence of Newton method (Nesterov and Nemirovski, 1994)

- Generalized notion: $|\varphi'''(t)| \leqslant \varphi''(t)$
  - Applicable to logistic regression (with extensions)
  - $\varphi(t) = \log(1 + e^{-t})$, $\varphi'(t) = (1 + e^{t})^{-1}$, etc...

- Important properties
  - Allows global Taylor expansions
  - Relates expansions of derivatives of different orders

Adaptive algorithm for logistic regression

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

- **Cannot be strongly convex** $\Rightarrow$ local strong convexity
  - unless restricted to $|\theta^\top \Phi(x_n)| \leqslant M$ (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** $1/(2R^2\sqrt{n})$
  - with $R =$ radius of data (Bach, 2013):

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

  - **A single adaptive algorithm for smooth problems with convergence rate $O(1/n)$ in all situations?**

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 $\theta \in \mathbb{R}^d$
  - 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 $\theta \in \mathbb{R}^d$
  - 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: $\boxed{\mathbb{E} f(\bar{\theta}_{n-1}) - f(\theta_*) \leqslant \frac{4\sigma^2 d}{n} + \frac{4R^2\|\theta_0 - \theta_*\|^2}{n}}$
- **Matches statistical lower bound** (Tsybakov, 2003)
  - Non-asymptotic robust version of Györfi and Walk (1996)

Least-squares - Proof technique - I

- LMS recursion:

$$\theta_n - \theta_* = \big[I - \gamma \Phi(x_n) \otimes \Phi(x_n)\big](\theta_{n-1} - \theta_*) + \gamma\, \varepsilon_n \Phi(x_n)$$

- Simplified LMS recursion: with $H = \mathbb{E}\big[\Phi(x_n) \otimes \Phi(x_n)\big]$

$$\theta_n - \theta_* = \big[I - \gamma H\big](\theta_{n-1} - \theta_*) + \gamma\, \varepsilon_n \Phi(x_n)$$

  - Direct proof technique of Polyak and Juditsky (1992), e.g.,

$$\theta_n - \theta_* = \big[I - \gamma H\big]^n(\theta_0 - \theta_*) + \gamma \sum_{k=1}^{n} \big[I - \gamma H\big]^{n-k} \varepsilon_k \Phi(x_k)$$

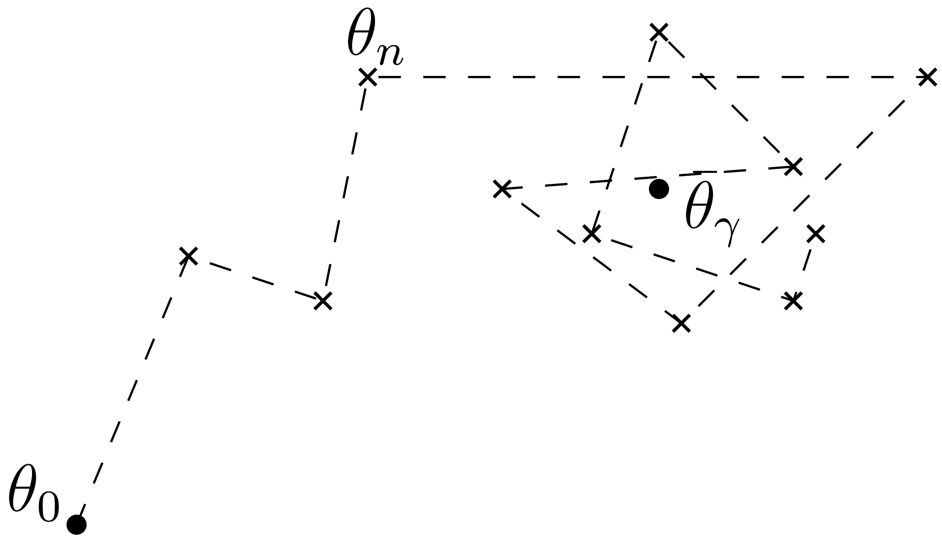
- Infinite expansion of Aguech, Moulines, and Priouret (2000) in powers of γ

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 $\bar{\theta}_\gamma \stackrel{\text{def}}{=} \int \theta \pi_\gamma(\mathrm{d}\theta)$



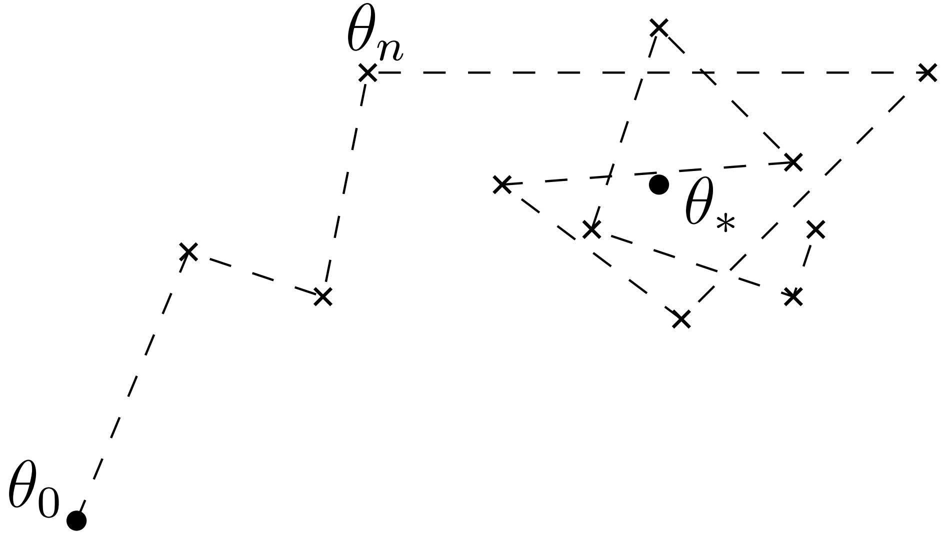
Markov chain interpretation of constant step sizes

- 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,** $\bar{\theta}_\gamma = \theta_*$



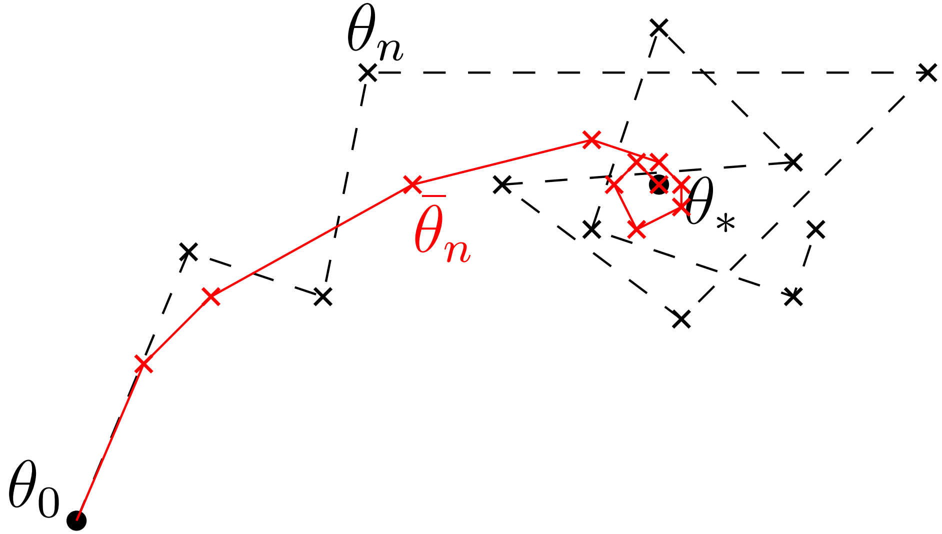
Markov chain interpretation of constant step sizes

- 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, $\bar{\theta}_\gamma = \theta_*$**



Markov chain interpretation of constant step sizes

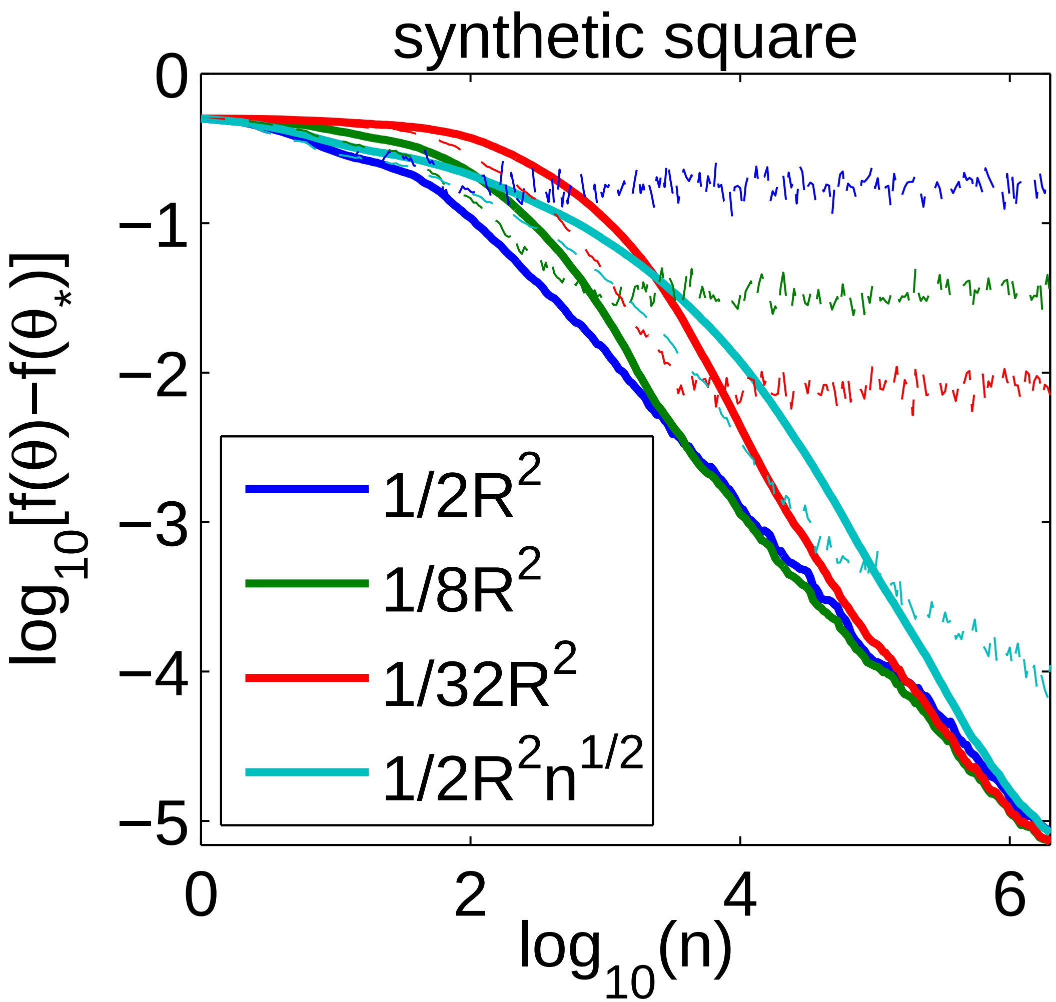
- LMS recursion for $f_n(\theta) = \frac{1}{2}\big(y_n - \langle \Phi(x_n), \theta \rangle\big)^2$

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

- The sequence $(\theta_n)_n$ is a homogeneous Markov chain
  - convergence to a stationary distribution π_γ
  - with expectation $\bar{\theta}_\gamma \overset{\text{def}}{=} \int \theta \pi_\gamma(d\theta)$
- **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 $O(1/n)$

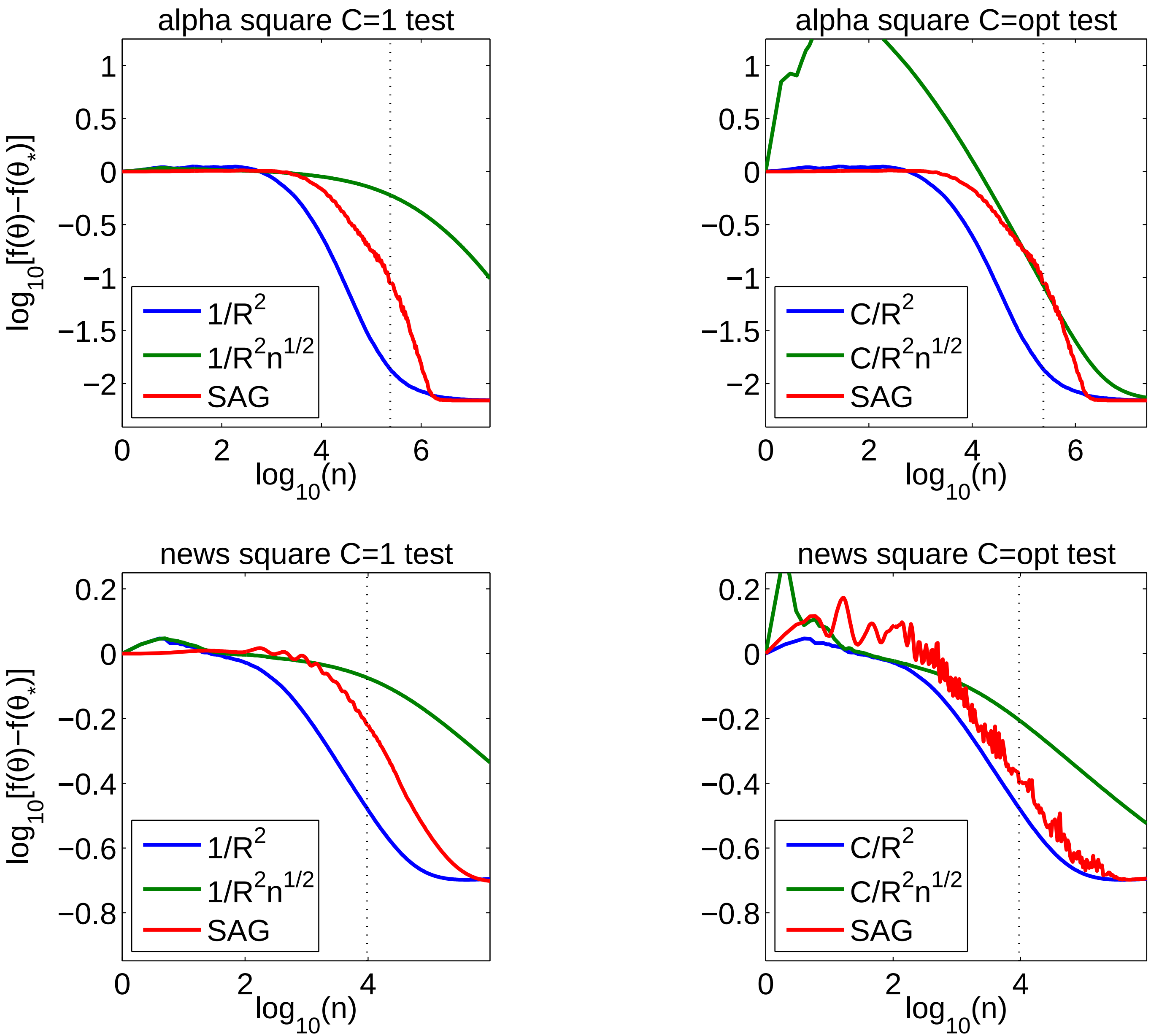
Simulations - synthetic examples

- Gaussian distributions - $d = 20$



Simulations - benchmarks

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



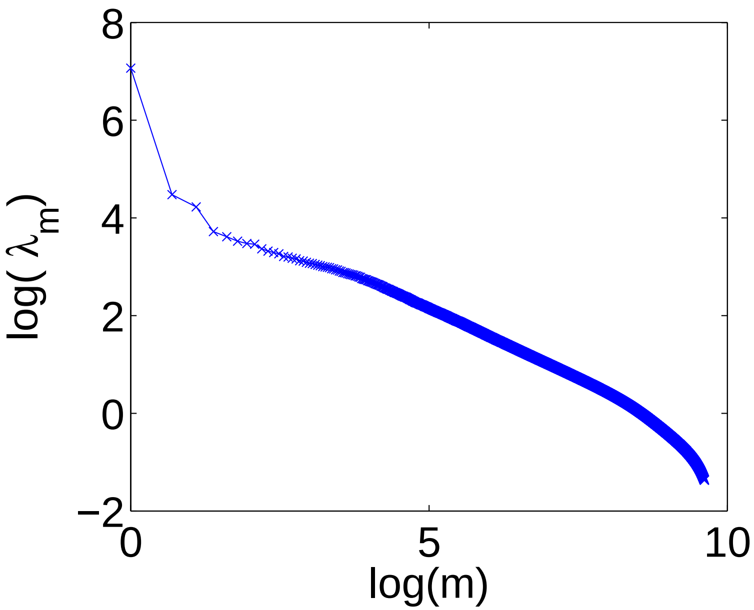
Optimal bounds for least-squares?

- **Least-squares**: cannot beat $\sigma^2 d/n$ (Tsybakov, 2003). Really?
  - What if $d \gg n$?

- **Refined assumptions with adaptivity** (Dieuleveut and Bach, 2014)
  - Beyond strong convexity or lack thereof

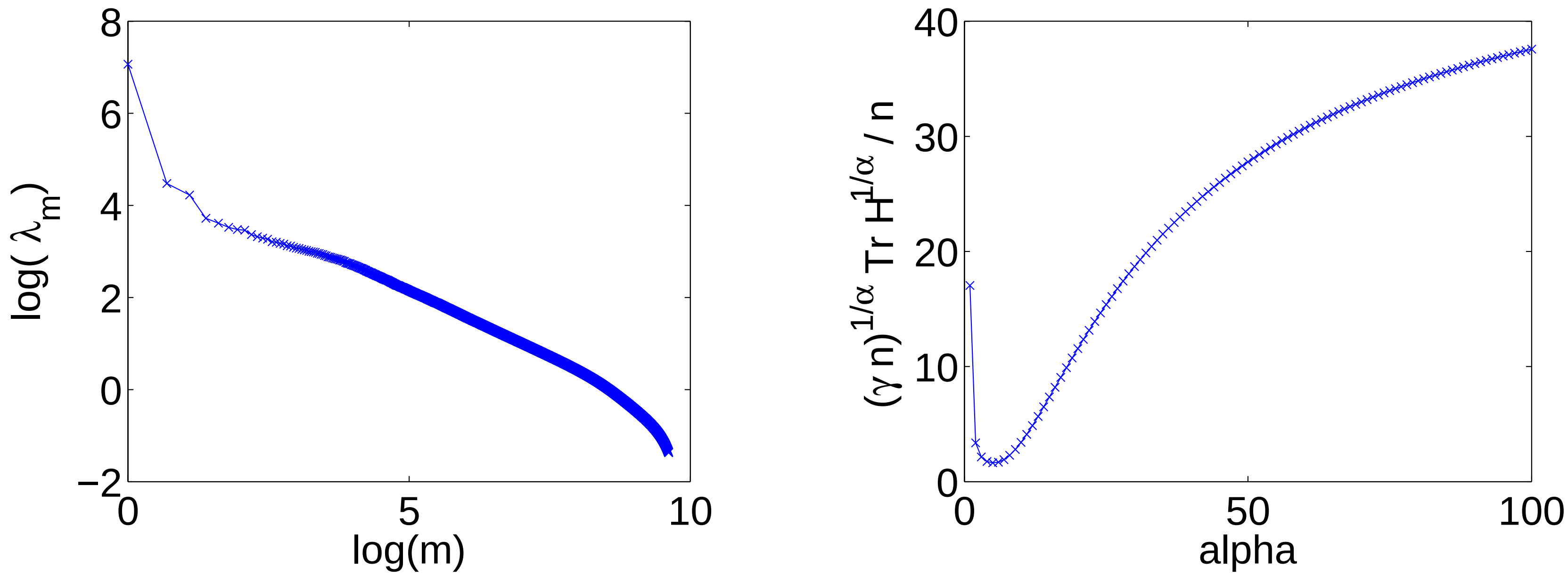
Finer assumptions (Dieuleveut and Bach, 2014)

- **Covariance eigenvalues**
  - Pessimistic assumption: all eigenvalues λ_m less than a constant
  - Actual decay as $\lambda_m = o(m^{-\alpha})$ with $\operatorname{tr} H^{1/\alpha} = \sum_m \lambda_m^{1/\alpha}$ small



Finer assumptions (Dieuleveut and Bach, 2014)

- **Covariance eigenvalues**
  - Pessimistic assumption: all eigenvalues λ_m less than a constant
  - Actual decay as $\lambda_m = o(m^{-\alpha})$ with $\operatorname{tr} H^{1/\alpha} = \sum_m \lambda_m^{1/\alpha}$ small
  - New result: replace $\dfrac{\sigma^2 d}{n}$ by $\dfrac{\sigma^2 (\gamma n)^{1/\alpha} \operatorname{tr} H^{1/\alpha}}{n}$



Finer assumptions (Dieuleveut and Bach, 2014)

- **Covariance eigenvalues**
  - Pessimistic assumption: all eigenvalues λ_m less than a constant
  - Actual decay as $\lambda_m = o(m^{-\alpha})$ with $\operatorname{tr} H^{1/\alpha} = \sum_m \lambda_m^{1/\alpha}$ small
  - New result: replace $\dfrac{\sigma^2 d}{n}$ by $\dfrac{\sigma^2 (\gamma n)^{1/\alpha} \operatorname{tr} H^{1/\alpha}}{n}$

- **Optimal predictor**
  - Pessimistic assumption: $\|\theta_0 - \theta_*\|^2$ finite
  - Finer assumption: $\|H^{1/2-r}(\theta_0 - \theta_*)\|_2$ small
  - Replace $\dfrac{\|\theta_0 - \theta_*\|^2}{\gamma n}$ by $\dfrac{4\|H^{1/2-r}(\theta_0 - \theta_*)\|_2}{\gamma^{2r} n^{2\min\{r,1\}}}$

Optimal bounds for least-squares?

- **Least-squares**: cannot beat $\sigma^2 d/n$ (Tsybakov, 2003). Really?
  - What if $d \gg n$?

- **Refined assumptions with adaptivity** (Dieuleveut and Bach, 2014)
  - Beyond strong convexity or lack thereof

$$f(\bar{\theta}_n) - f(\theta_*) \leqslant \frac{16\sigma^2 \operatorname{tr} H^{1/\alpha}}{n}(\gamma n)^{1/\alpha} + \frac{4\|H^{1/2-r}(\theta_0 - \theta_*)\|_2}{\gamma^{2r} n^{2\min\{r,1\}}}$$

  - Previous results: $\alpha = +\infty$ and $r = 1/2$
  - Valid for all α and r
  - Optimal step-size potentially decaying with n
  - Extension to non-parametric estimation (kernels) with optimal rates

From least-squares to non-parametric estimation - I

- **Extension to Hilbert spaces**: $\Phi(x), \theta \in \mathcal{H}$

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

- **If $\theta_0 = 0$, θ_n is a linear combination of** $\Phi(x_1), \dots, \Phi(x_n)$

$$\theta_n = \sum_{k=1}^{n} \alpha_k \Phi(x_k) \quad \text{and} \quad \alpha_n = -\gamma \sum_{k=1}^{n-1} \alpha_k \langle \Phi(x_k), \Phi(x_n)\rangle + \gamma y_n$$

From least-squares to non-parametric estimation - I

- **Extension to Hilbert spaces**: $\Phi(x), \theta \in \mathcal{H}$

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

- **If $\theta_0 = 0$, θ_n is a linear combination of** $\Phi(x_1), \dots, \Phi(x_n)$

$$\theta_n = \sum_{k=1}^{n} \alpha_k \Phi(x_k) \ \text{ and } \ \alpha_n = -\gamma \sum_{k=1}^{n-1} \alpha_k \langle \Phi(x_k), \Phi(x_n)\rangle + \gamma y_n$$

- **Kernel trick**: $k(x, x') = \langle \Phi(x), \Phi(x')\rangle$
  - Reproducing kernel Hilbert spaces and non-parametric estimation
  - See, e.g., Schölkopf and Smola (2001); Shawe-Taylor and Cristianini (2004); Dieuleveut and Bach (2014)
  - Still $O(n^2)$

From least-squares to non-parametric estimation - II

- **Simple example**: Sobolev space on $\mathcal{X} = [0, 1]$
  - $\Phi(x) =$ weighted Fourier basis $\Phi(x)_j = \varphi_j \cos(2j\pi x)$ (plus sine)
  - kernel $k(x, x') = \sum_j \varphi_j^2 \cos\big[2j\pi(x - x')\big]$
  - Optimal prediction function θ_* has norm $\|\theta_*\|^2 = \sum_j |\mathcal{F}(\theta_*)_j|^2 \varphi_j^{-2}$
  - Depending on smoothness, may or may not be finite

From least-squares to non-parametric estimation - II

- **Simple example**: Sobolev space on $\mathcal{X} = [0,1]$
  - $\Phi(x)$ = weighted Fourier basis $\Phi(x)_j = \varphi_j \cos(2j\pi x)$ (plus sine)
  - kernel $k(x,x') = \sum_j \varphi_j^2 \cos\big[2j\pi(x-x')\big]$
  - Optimal prediction function θ_* has norm $\|\theta_*\|^2 = \sum_j |\mathcal{F}(\theta_*)_j|^2 \varphi_j^{-2}$
  - Depending on smoothness, may or may not be finite

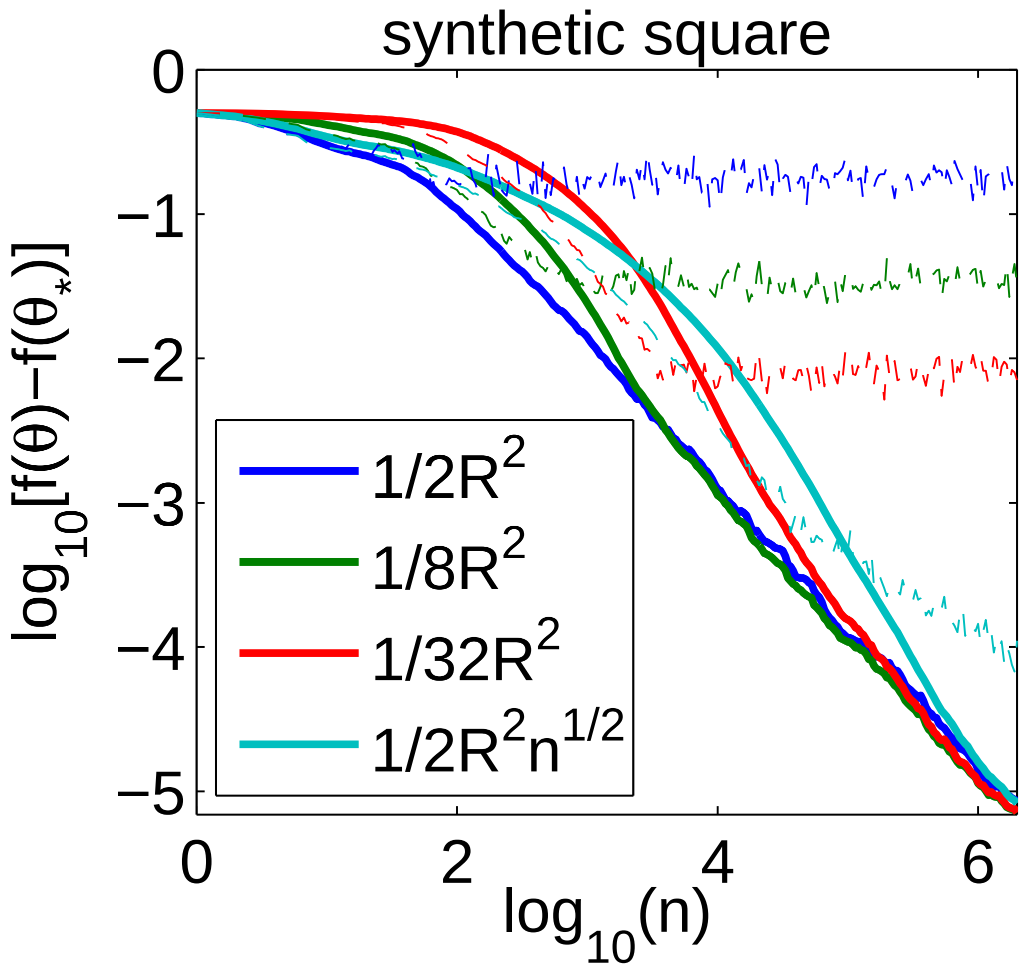
- Adapted norm $\|H^{1/2-r}\theta_*\|^2 = \sum_j |\mathcal{F}(\theta_*)_j|^2 \varphi_j^{-4r}$ may be finite

$$f(\bar{\theta}_n) - f(\theta_*) \leqslant \frac{16\sigma^2 \operatorname{tr} H^{1/\alpha}}{n}(\gamma n)^{1/\alpha} + \frac{4\|H^{1/2-r}(\theta_0 - \theta_*)\|_2}{\gamma^{2r} n^{2\min\{r,1\}}}$$

- **Same effect than ℓ_2-regularization with weight λ equal to** $\dfrac{1}{\gamma n}$

Simulations - synthetic examples

- Gaussian distributions - $d = 20$



- **Explaining actual behavior for all n**

Bias-variance decomposition (Défossez and Bach, 2015)

- Simplification: dominating (but exact) term when $n \to \infty$ and $\gamma \to 0$
- **Variance** (e.g., starting from the solution)

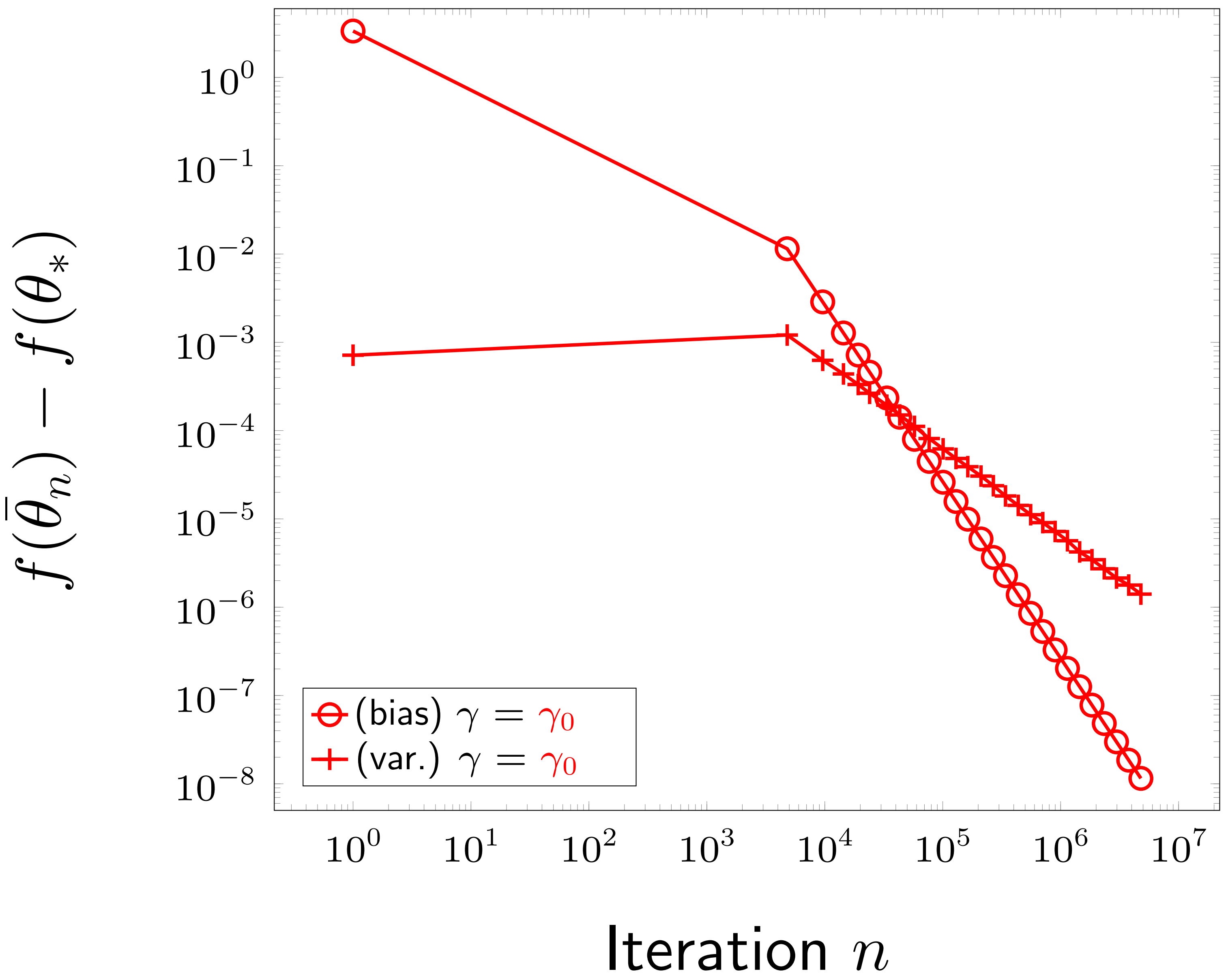
$$f(\bar{\theta}_n) - f(\theta_*) \sim \frac{1}{n} \mathbb{E}\Big[\varepsilon^2 \, \Phi(x)^\top H^{-1} \Phi(x)\Big]$$

  – NB: if noise ε is independent, then we obtain $\frac{d\sigma^2}{n}$
  – Exponentially decaying remainder terms (strongly convex problems)

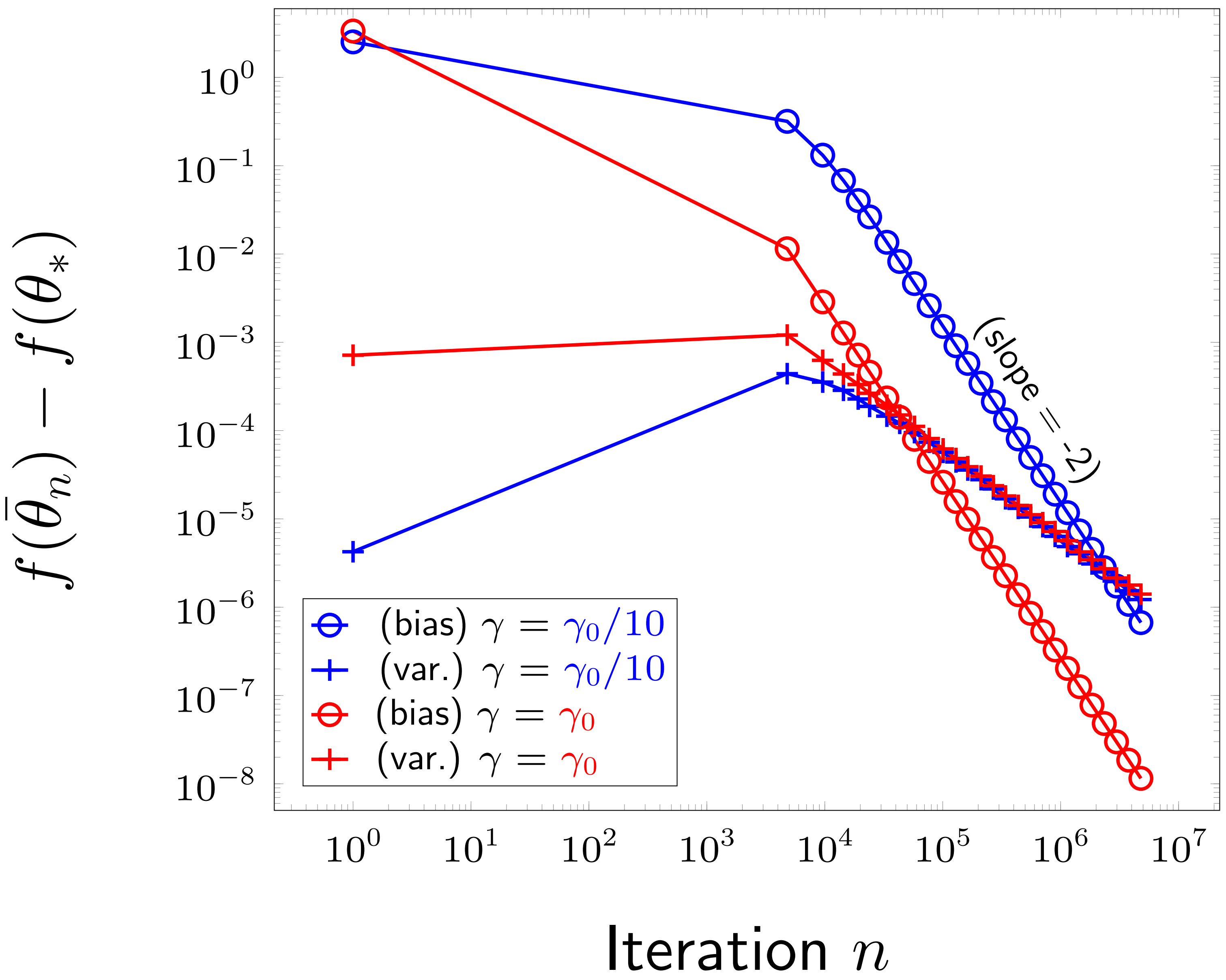
- **Bias** (e.g., no noise)

$$f(\bar{\theta}_n) - f(\theta_*) \sim \frac{1}{n^2\gamma^2} (\theta_0 - \theta_*)^\top H^{-1} (\theta_0 - \theta_*)$$

Bias-variance decomposition (synthetic data $d = 25$)



Bias-variance decomposition (synthetic data $d = 25$)



Optimal sampling (Défossez and Bach, 2015)

- **Sampling from a different distribution with importance weights**

$$\mathbb{E}_{p(x)p(y|x)}|y - \Phi(x)^\top \theta|^2 = \mathbb{E}_{q(x)p(y|x)} \frac{dp(x)}{dq(x)} |y - \Phi(x)^\top \theta|^2$$

  – Recursion: $\theta_n = \theta_{n-1} - \gamma \frac{dp(x_n)}{dq(x_n)} \big(\Phi(x_n)^\top \theta_{n-1} - y_n\big) \Phi(x_n)$

Optimal sampling (Défossez and Bach, 2015)

- **Sampling from a different distribution with importance weights**

$$\mathbb{E}_{p(x)p(y|x)}|y - \Phi(x)^\top \theta|^2 = \mathbb{E}_{q(x)p(y|x)} \frac{dp(x)}{dq(x)} |y - \Phi(x)^\top \theta|^2$$

  - Recursion: $\theta_n = \theta_{n-1} - \gamma \frac{dp(x_n)}{dq(x_n)} \big(\Phi(x_n)^\top \theta_{n-1} - y_n\big) \Phi(x_n)$
  - Specific to least-squares $= \mathbb{E}_{q(x)p(y|x)} \left| \sqrt{\frac{dp(x)}{dq(x)}} y - \sqrt{\frac{dp(x)}{dq(x)}} \Phi(x)^\top \theta \right|^2$
  - Reweighting of the data: same bounds apply!

Optimal sampling (Défossez and Bach, 2015)

- **Sampling from a different distribution with importance weights**

$$\mathbb{E}_{p(x)p(y|x)}|y - \Phi(x)^\top\theta|^2 = \mathbb{E}_{q(x)p(y|x)}\frac{dp(x)}{dq(x)}|y - \Phi(x)^\top\theta|^2$$

  - Recursion: $\theta_n = \theta_{n-1} - \gamma\frac{dp(x_n)}{dq(x_n)}\big(\Phi(x_n)^\top\theta_{n-1} - y_n\big)\Phi(x_n)$
  - Specific to least-squares $= \mathbb{E}_{q(x)p(y|x)}\bigg|\sqrt{\frac{dp(x)}{dq(x)}}y - \sqrt{\frac{dp(x)}{dq(x)}}\Phi(x)^\top\theta\bigg|^2$
  - Reweighting of the data: same bounds apply!

- **Optimal for variance**: $\displaystyle\frac{dq(x)}{dp(x)} \propto \sqrt{\Phi(x)^\top H^{-1}\Phi(x)}$
  - Same density as active learning (Kanamori and Shimodaira, 2003)
  - Limited gains: different between first and second moments
  - Caveat: need to know H

Optimal sampling (Défossez and Bach, 2015)

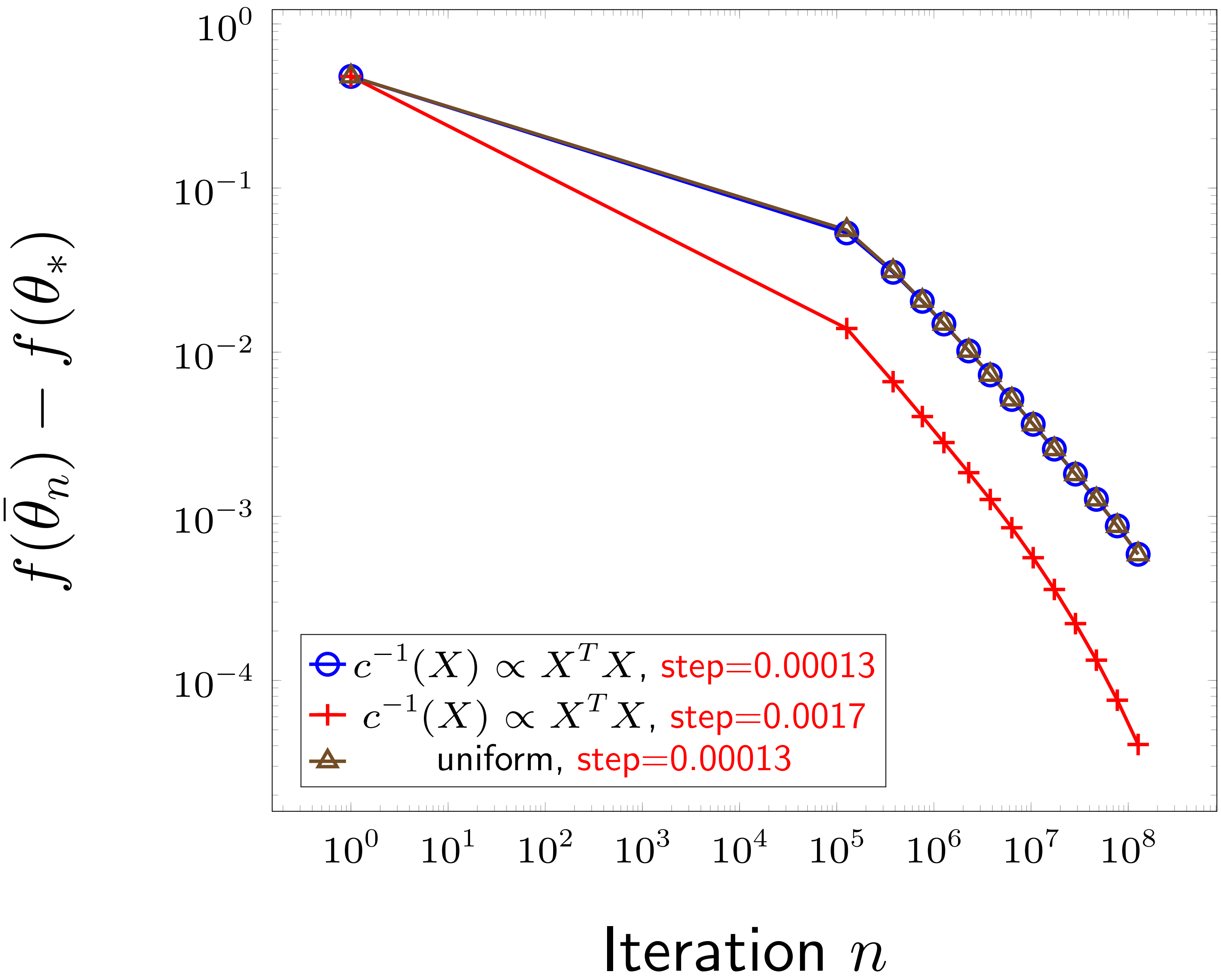
- **Sampling from a different distribution with importance weights**

$$\mathbb{E}_{p(x)p(y|x)}|y - \Phi(x)^\top\theta|^2 = \mathbb{E}_{q(x)p(y|x)}\frac{dp(x)}{dq(x)}|y - \Phi(x)^\top\theta|^2$$

  - Recursion: $\theta_n = \theta_{n-1} - \gamma\frac{dp(x_n)}{dq(x_n)}\big(\Phi(x_n)^\top\theta_{n-1} - y_n\big)\Phi(x_n)$
  - Specific to least-squares $= \mathbb{E}_{q(x)p(y|x)}\Big|\sqrt{\frac{dp(x)}{dq(x)}}y - \sqrt{\frac{dp(x)}{dq(x)}}\Phi(x)^\top\theta\Big|^2$
  - Reweighting of the data: same bounds apply!

- **Optimal for bias**: $\dfrac{dq(x)}{dp(x)} \propto \|\Phi(x)\|^2$
  - Simpy allows biggest possible step size $\gamma < \frac{2}{\operatorname{tr} H}$
  - Large gains in practice
  - Corresponds to normalized least-mean-squares

Convergence on *Sido* dataset ($d = 4932$)



Achieving optimal bias and variance terms

- **Current results with averaged SGD**
  - **Variance** (starting from optimal θ_*) $= \dfrac{\sigma^2 d}{n}$
  - **Bias** (no noise) $= \min\left\{\dfrac{R^2\|\theta_0-\theta_*\|^2}{n}, \dfrac{R^4\langle\theta_0-\theta_*, H^{-1}(\theta_0-\theta_*)\rangle}{n^2}\right\}$

Achieving optimal bias and variance terms

- **Current results with averaged SGD** (ill-conditioned problems)
  - **Variance** (starting from optimal θ_*) $= \dfrac{\sigma^2 d}{n}$
  - **Bias** (no noise) $= \dfrac{R^2\|\theta_0 - \theta_*\|^2}{n}$

Achieving optimal bias and variance terms

- **Current results with averaged SGD** (ill-conditioned problems)
  - **Variance** (starting from optimal θ_*) $= \dfrac{\sigma^2 d}{n}$
  - **Bias** (no noise) $= \dfrac{R^2\|\theta_0 - \theta_*\|^2}{n}$

	Bias	Variance
Averaged gradient descent (Bach and Moulines, 2013)	$\dfrac{R^2\|\theta_0 - \theta_*\|^2}{n}$	$\dfrac{\sigma^2 d}{n}$

Achieving optimal bias and variance terms

	Bias	Variance
Averaged gradient descent (Bach and Moulines, 2013)	$\frac{R^2\|\theta_0 - \theta_*\|^2}{n}$	$\frac{\sigma^2 d}{n}$

Achieving optimal bias and variance terms

	Bias	Variance
Averaged gradient descent (Bach and Moulines, 2013)	$\dfrac{R^2\|\theta_0 - \theta_*\|^2}{n}$	$\dfrac{\sigma^2 d}{n}$
Accelerated gradient descent (Nesterov, 1983)	$\dfrac{R^2\|\theta_0 - \theta_*\|^2}{n^2}$	$\sigma^2 d$

- **Acceleration is notoriously non-robust to noise** (d'Aspremont, 2008; Schmidt et al., 2011)
  - For non-structured noise, see Lan (2012)

Achieving optimal bias and variance terms

	Bias	Variance
Averaged gradient descent (Bach and Moulines, 2013)	$\frac{R^2\|\theta_0 - \theta_*\|^2}{n}$	$\frac{\sigma^2 d}{n}$
Accelerated gradient descent (Nesterov, 1983)	$\frac{R^2\|\theta_0 - \theta_*\|^2}{n^2}$	$\sigma^2 d$
"Between" averaging and acceleration (Flammarion and Bach, 2015)	$\frac{R^2\|\theta_0 - \theta_*\|^2}{n^{1+\alpha}}$	$\frac{\sigma^2 d}{n^{1-\alpha}}$

Achieving optimal bias and variance terms

	Bias	Variance
Averaged gradient descent (Bach and Moulines, 2013)	$\frac{R^2\|\theta_0-\theta_*\|^2}{n}$	$\frac{\sigma^2 d}{n}$
Accelerated gradient descent (Nesterov, 1983)	$\frac{R^2\|\theta_0-\theta_*\|^2}{n^2}$	$\sigma^2 d$
"Between" averaging and acceleration (Flammarion and Bach, 2015)	$\frac{R^2\|\theta_0-\theta_*\|^2}{n^{1+\alpha}}$	$\frac{\sigma^2 d}{n^{1-\alpha}}$
Averaging and acceleration (Dieuleveut, Flammarion, and Bach, 2016)	$\frac{R^2\|\theta_0-\theta_*\|^2}{n^2}$	$\frac{\sigma^2 d}{n}$

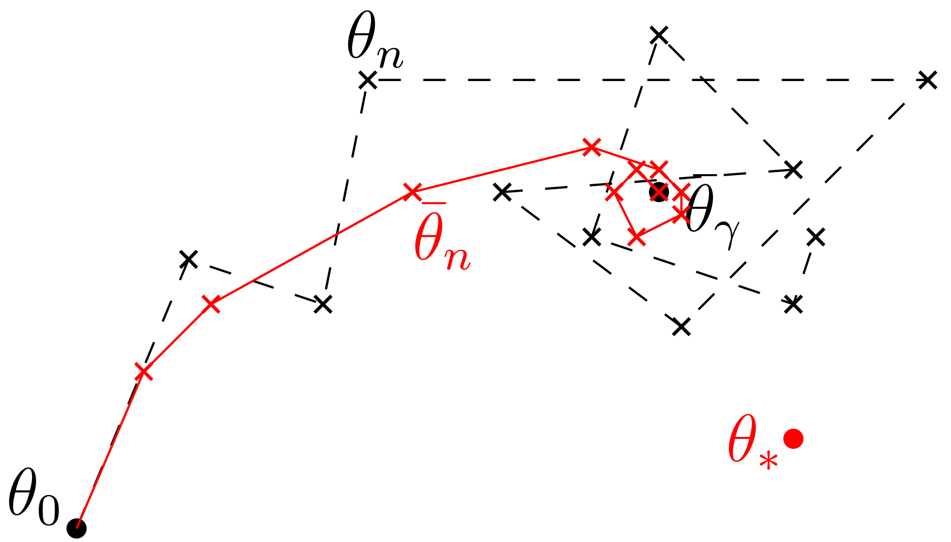
Beyond least-squares - Markov chain interpretation

- 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 $\bar{\theta}_\gamma \neq \theta_*$**



Beyond least-squares - Markov chain interpretation

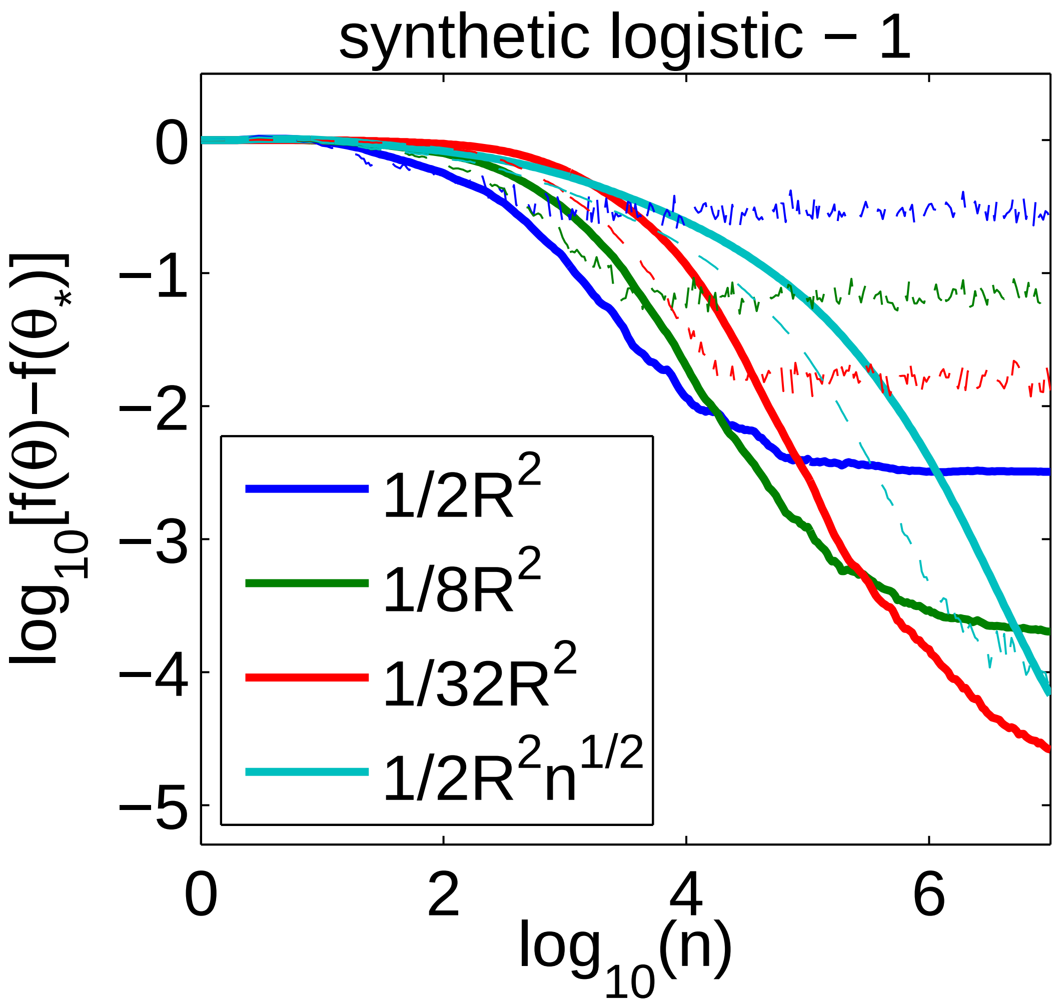
- Recursion $\theta_n = \theta_{n-1} - \gamma f'_n(\theta_{n-1})$ also defines a Markov chain
  - Stationary distribution π_γ such that $\int f'(\theta)\pi_\gamma(\mathrm{d}\theta) = 0$
  - When f' is not linear, $f'(\int \theta\pi_\gamma(\mathrm{d}\theta)) \neq \int f'(\theta)\pi_\gamma(\mathrm{d}\theta)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

- θ_n **oscillates around the wrong value** $\bar{\theta}_\gamma \neq \theta_*$
  - moreover, $\|\theta_* - \theta_n\| = O_p(\sqrt{\gamma})$
  - Linear convergence up to the noise level for strongly-convex problems (Nedic and Bertsekas, 2000)

- **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 - $d = 20$



Restoring convergence through online Newton steps

- **Known facts**

  1. Averaged SGD with $\gamma_n \propto n^{-1/2}$ leads to *robust* rate $O(n^{-1/2})$ for all convex functions
  2. Averaged SGD with γ_n constant leads to *robust* rate $O(n^{-1})$ for all convex *quadratic* functions
  3. Newton's method squares the error at each iteration for smooth functions
  4. A single step of Newton's method is equivalent to minimizing the quadratic Taylor expansion

Restoring convergence through online Newton steps

- **Known facts**

  1. Averaged SGD with $\gamma_n \propto n^{-1/2}$ leads to *robust* rate $O(n^{-1/2})$ for all convex functions
  2. Averaged SGD with γ_n constant leads to *robust* rate $O(n^{-1})$ for all convex *quadratic* functions $\Rightarrow O(n^{-1})$
  3. Newton's method squares the error at each iteration for smooth functions $\Rightarrow O((n^{-1/2})^2)$
  4. A single step of Newton's method is equivalent to minimizing the quadratic Taylor expansion

- **Online Newton step**

  – Rate: $O((n^{-1/2})^2 + n^{-1}) = O(n^{-1})$
  – Complexity: $O(d)$ per iteration

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 $O(d)$**

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

  - $f_n''(\tilde{\theta}) = \ell''(y_n, \langle\tilde{\theta}, \Phi(x_n)\rangle)\Phi(x_n) \otimes \Phi(x_n)$ 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 $O(d/n)$ for logistic regression
  - Additional assumptions but no strong convexity

Choice of support point for online Newton step

- **Two-stage procedure**

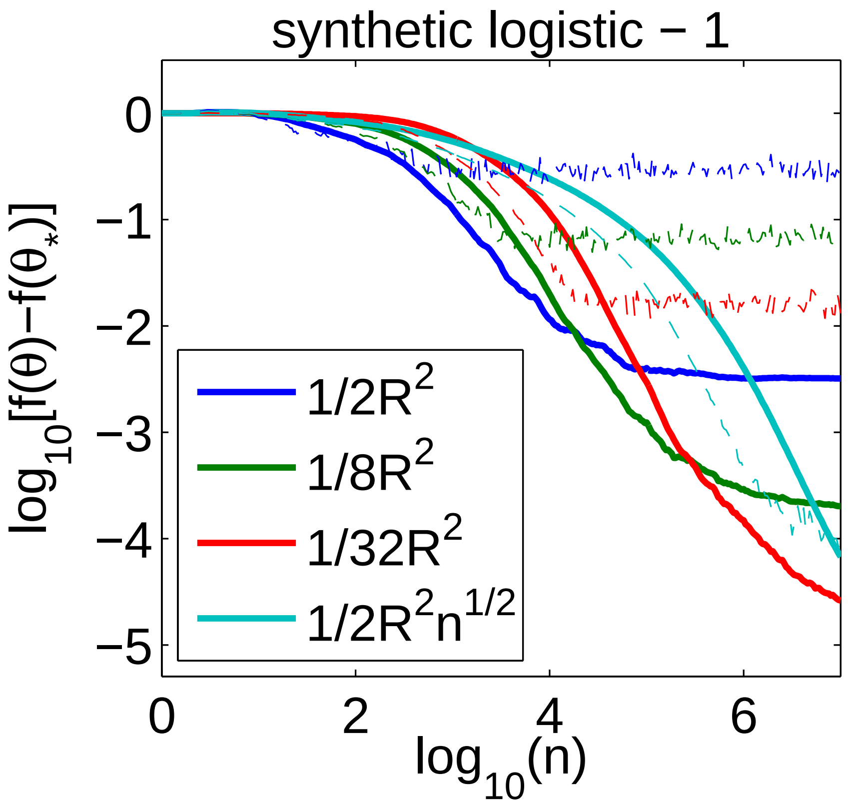
(1) Run $n/2$ iterations of averaged SGD to obtain $\tilde{\theta}$
(2) Run $n/2$ iterations of averaged constant step-size LMS

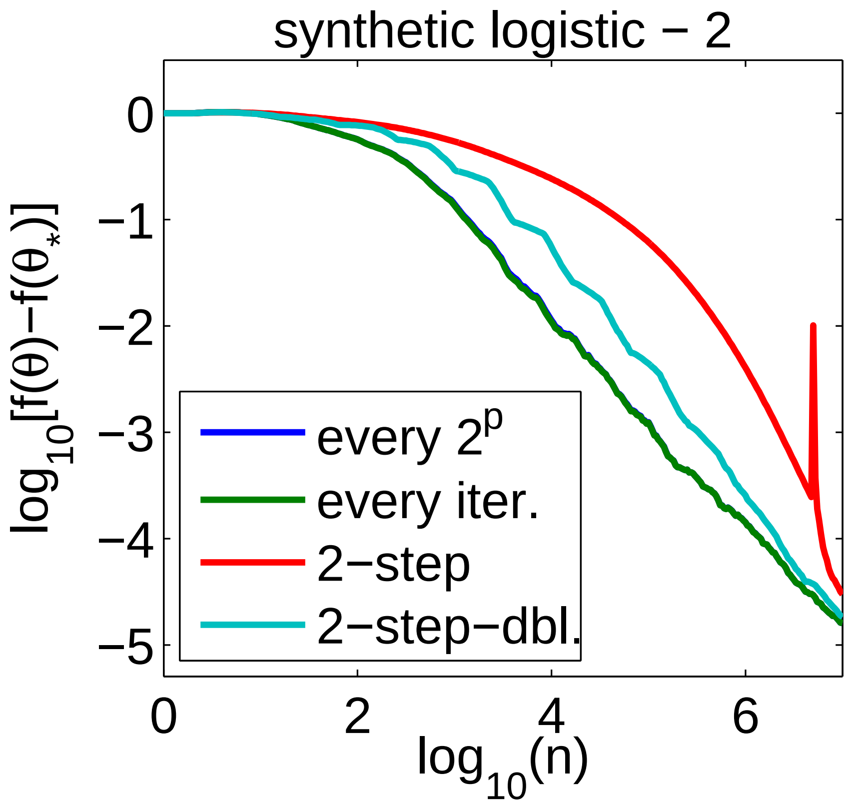
  - Reminiscent of one-step estimators (see, e.g., Van der Vaart, 2000)
  - Provable convergence rate of $O(d/n)$ for logistic regression
  - Additional assumptions but no strong convexity

- **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 (yet) but best practical behavior
  - Note (dis)similarity with regular SGD: $\theta_n = \theta_{n-1} - \gamma f_n'(\theta_{n-1})$

Simulations - synthetic examples

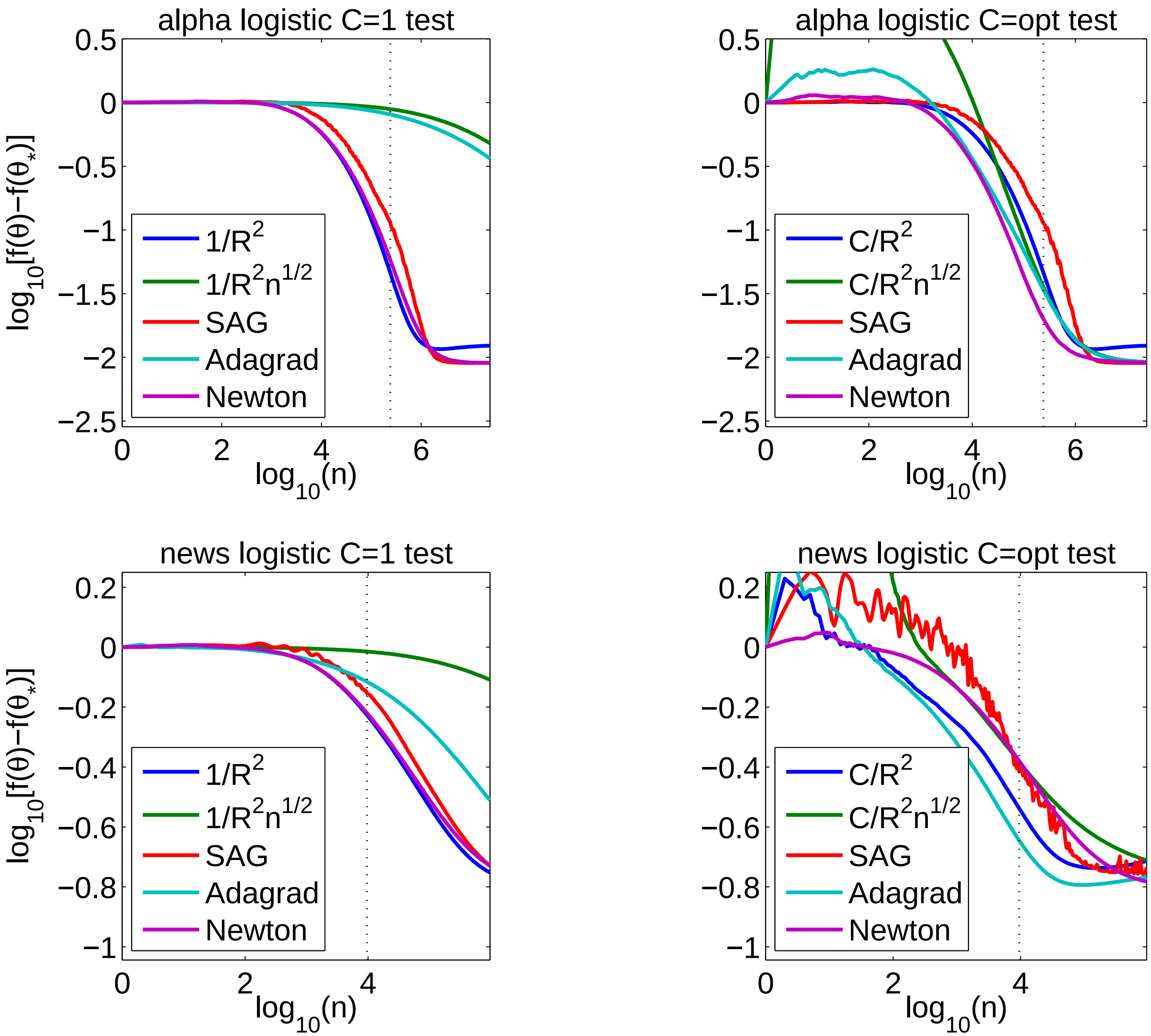
- Gaussian distributions - $d = 20$





Simulations - benchmarks

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



Summary of rates of convergence

- Problem parameters
  - D diameter of the domain
  - B Lipschitz-constant
  - L smoothness constant
  - μ strong convexity constant

	convex	strongly convex
nonsmooth	deterministic: $BD/\sqrt{t}$ stochastic: $BD/\sqrt{n}$	deterministic: $B^2/(t\mu)$ stochastic: $B^2/(n\mu)$
smooth	deterministic: LD^2/t^2 stochastic: $LD^2/\sqrt{n}$	deterministic: $\exp(-t\sqrt{\mu/L})$ stochastic: $L/(n\mu)$
quadratic	deterministic: LD^2/t^2 stochastic: $d/n + LD^2/n$	deterministic: $\exp(-t\sqrt{\mu/L})$ stochastic: $d/n + LD^2/n$

Summary of rates of convergence

- Problem parameters
  - D diameter of the domain
  - B Lipschitz-constant
  - L smoothness constant
  - μ strong convexity constant

	convex	strongly convex
nonsmooth	deterministic: $BD/\sqrt{t}$ stochastic: $BD/\sqrt{n}$	deterministic: $B^2/(t\mu)$ stochastic: $B^2/(n\mu)$
smooth	deterministic: LD^2/t^2 stochastic: $LD^2/\sqrt{n}$ finite sum: n/t	deterministic: $\exp(-t\sqrt{\mu/L})$ stochastic: $L/(n\mu)$ finite sum: $\exp(-\min\{1/n,\mu/L\}t)$
quadratic	deterministic: LD^2/t^2 stochastic: $d/n + LD^2/n$	deterministic: $\exp(-t\sqrt{\mu/L})$ stochastic: $d/n + LD^2/n$

Outline - I

1. **Introduction**
   - Large-scale machine learning and optimization
   - Classes of functions (convex, smooth, etc.)
   - Traditional statistical analysis through Rademacher complexity

2. **Classical methods for convex optimization**
   - Smooth optimization (gradient descent, Newton method)
   - Non-smooth optimization (subgradient descent)

3. **Classical stochastic approximation** (not covered)
   - Robbins-Monro algorithm (1951)

Outline - II

4. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds

5. **Smooth stochastic approximation algorithms**
   - Non-asymptotic analysis for smooth functions
   - Least-squares regression without decaying step-sizes

6. **Finite data sets**
   - Gradient methods with exponential convergence rates

Going beyond a single pass over the data

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

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

- **Goal**: minimize $g(\theta) = \dfrac{1}{n} \displaystyle\sum_{i=1}^n f_i(\theta)$

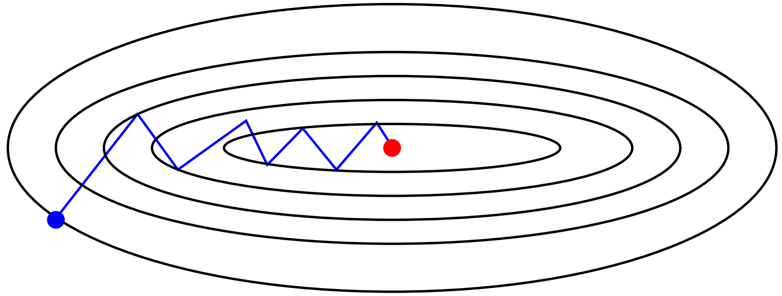
Stochastic vs. deterministic methods

- 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 in $O(e^{-\alpha t})$
  - Iteration complexity is linear in n *(with line search)*

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

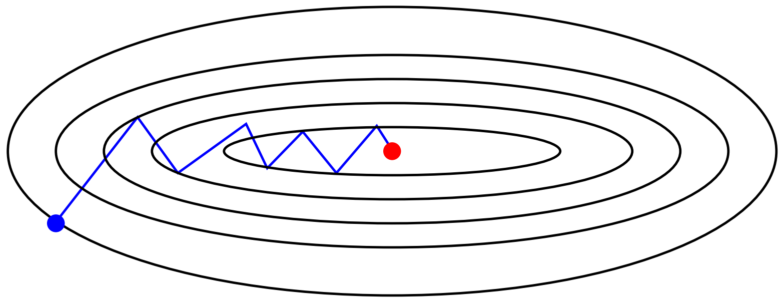
- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma_t g'(\theta_{t-1}) = \theta_{t-1} - \dfrac{\gamma_t}{n}\sum_{i=1}^{n} f_i'(\theta_{t-1})$
  - Linear (e.g., exponential) convergence rate in $O(e^{-\alpha t})$
  - Iteration complexity is linear in n *(with line search)*

- 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 *(step size selection?)*

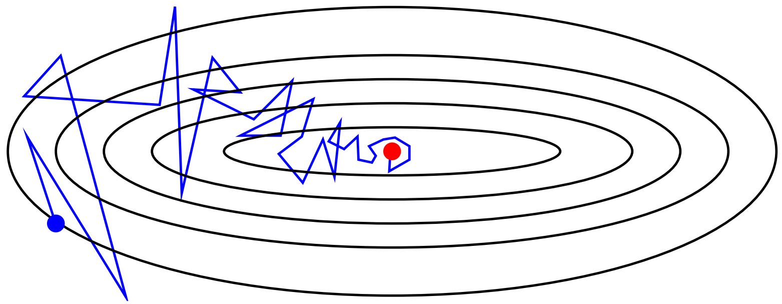
Stochastic vs. deterministic methods

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

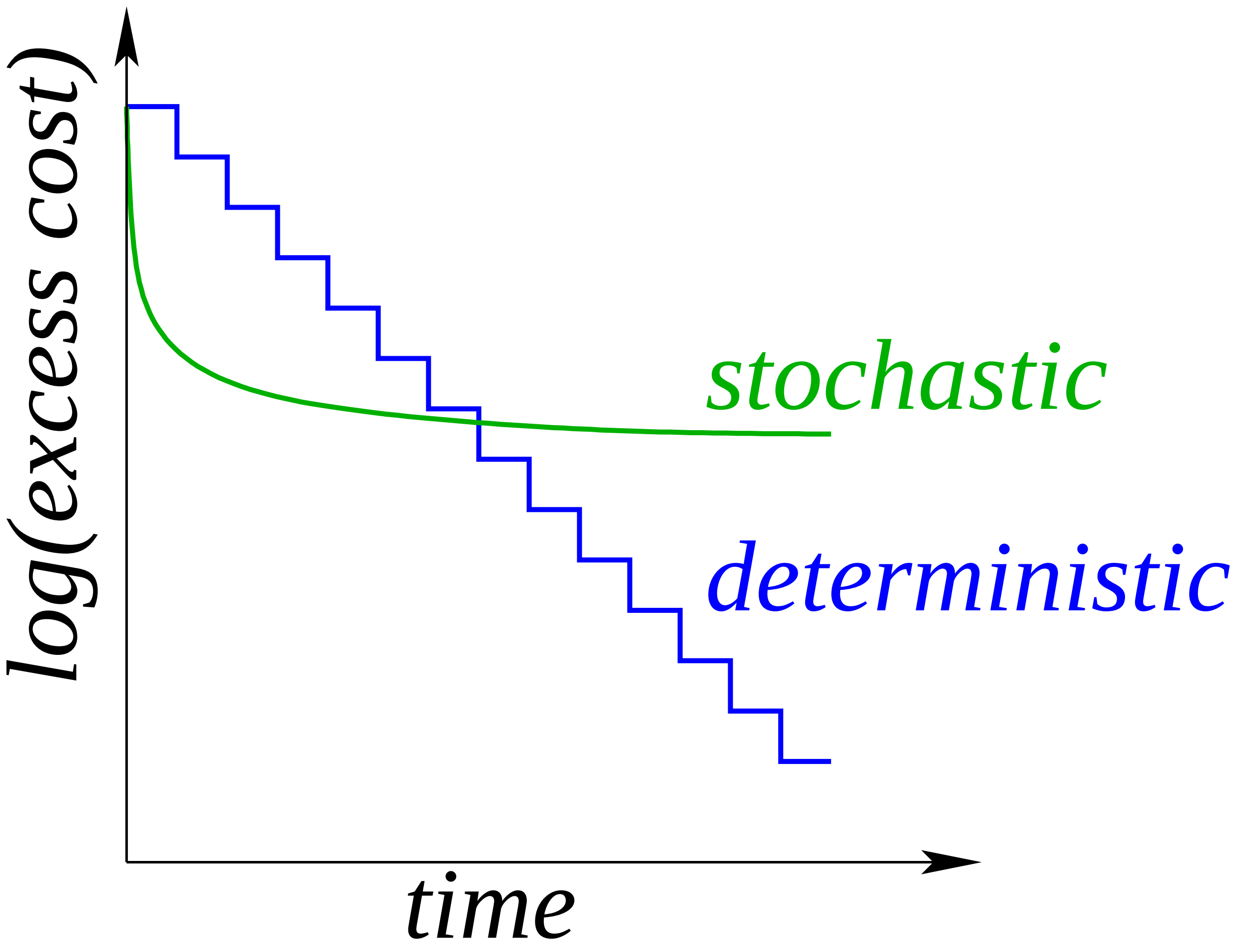


- Stochastic gradient descent: $\theta_t = \theta_{t-1} - \gamma_t f'_{i(t)}(\theta_{t-1})$



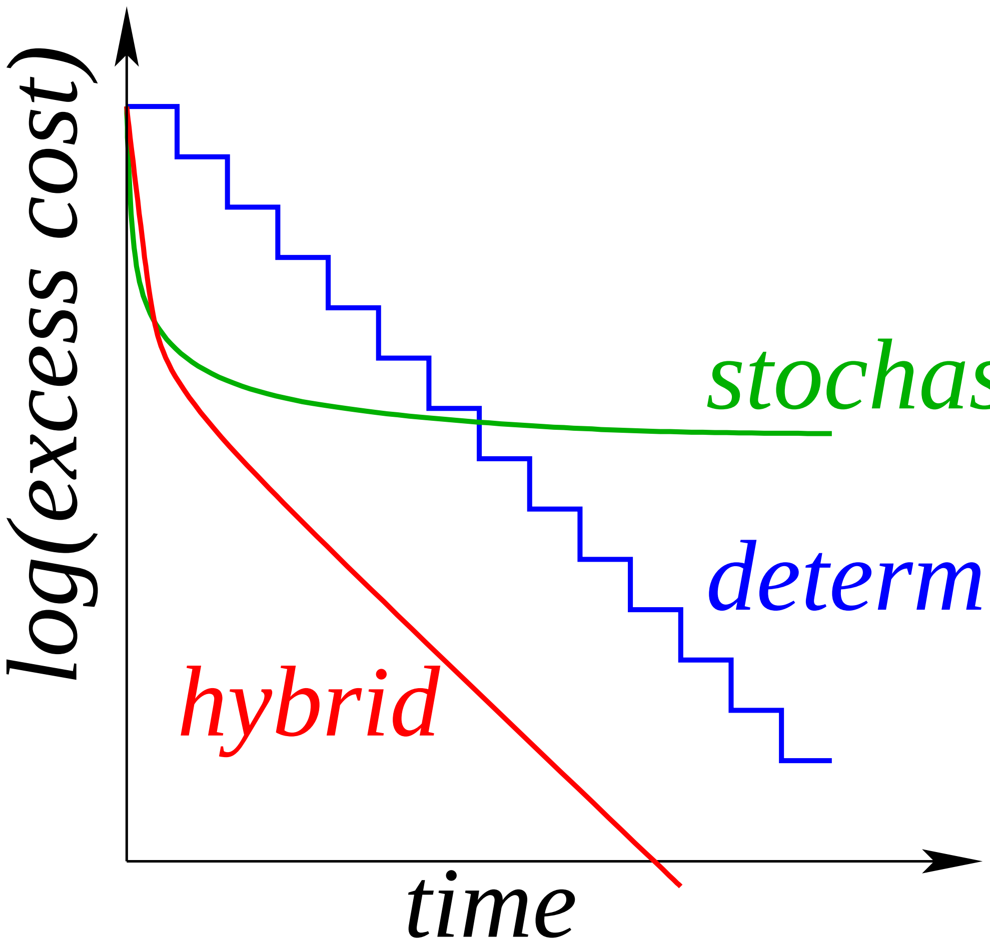
Stochastic vs. deterministic methods

- **Goal** = **best of both worlds**: Linear rate with $O(1)$ iteration cost
  Robustness to step size



Stochastic vs. deterministic methods

- **Goal** = **best of both worlds**: Linear rate with $O(1)$ iteration cost
  Robustness to step size



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 R^2-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 $\gamma_t = 1/(16R^2)$
  - initialization with one pass of averaged SGD

Stochastic average gradient - Convergence analysis

- **Assumptions**
  - Each f_i is R^2-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 $\gamma_t = 1/(16R^2)$
  - initialization with one pass of averaged SGD

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

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

  - Linear (exponential) convergence rate with $O(1)$ iteration cost
  - After one pass, reduction of cost by $\exp\Big(-\min\Big\{\frac{1}{8}, \frac{n\mu}{16R^2}\Big\}\Big)$

Stochastic average gradient - Convergence analysis

- **Assumptions**
  - Each f_i is R^2-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 $\gamma_t = 1/(16R^2)$
  - initialization with one pass of averaged SGD

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

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

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

Convergence analysis - Proof sketch

- **Main step**: find "good" Lyapunov function $J(\theta_t, y_1^t, \dots, y_n^t)$
  - such that $\mathbb{E}\big[J(\theta_t, y_1^t, \dots, y_n^t)|\mathcal{F}_{t-1}\big] < J(\theta_{t-1}, y_1^{t-1}, \dots, y_n^{t-1})$
  - no natural candidates

- **Computer-aided proof**
  - Parameterize function $J(\theta_t, y_1^t, \dots, y_n^t) = g(\theta_t) - g(\theta_*) + \text{quadratic}$
  - Solve semidefinite program to obtain candidates (that depend on n, μ, L)
  - Check validity with symbolic computations

Rate of convergence comparison

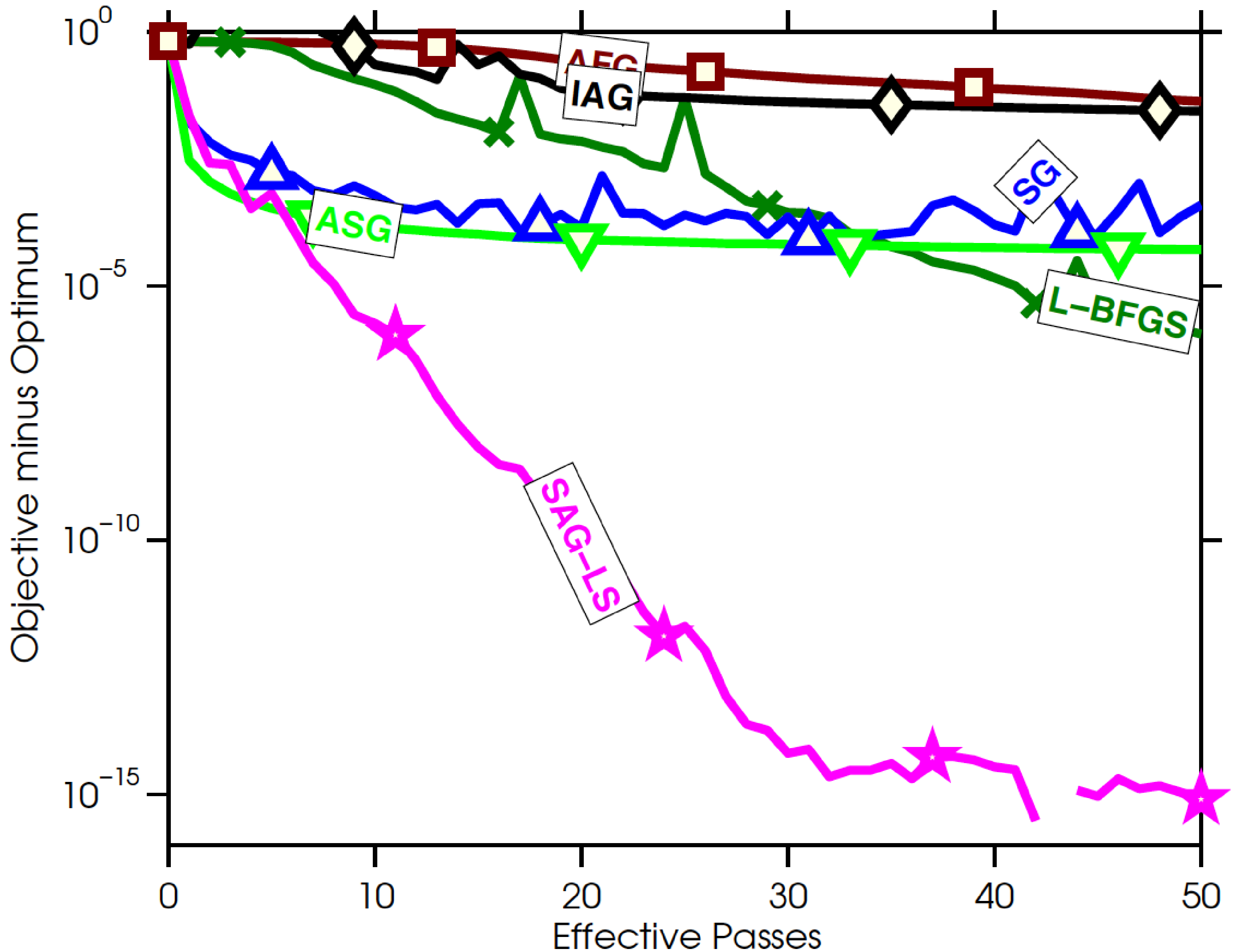
- Assume that $L = 100$, $\mu = .01$, and $n = 80000$ ($L \neq R^2$)
  - Full gradient method has rate
  $$\left(1 - \frac{\mu}{L}\right) = 0.9999$$
  - Accelerated gradient method has rate
  $$\left(1 - \sqrt{\frac{\mu}{L}}\right) = 0.9900$$
  - Running n iterations of SAG for the same cost has rate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

- **Beating two lower bounds** (with additional assumptions)
  - (1) stochastic gradient and (2) full gradient

Stochastic average gradient
Implementation details and extensions

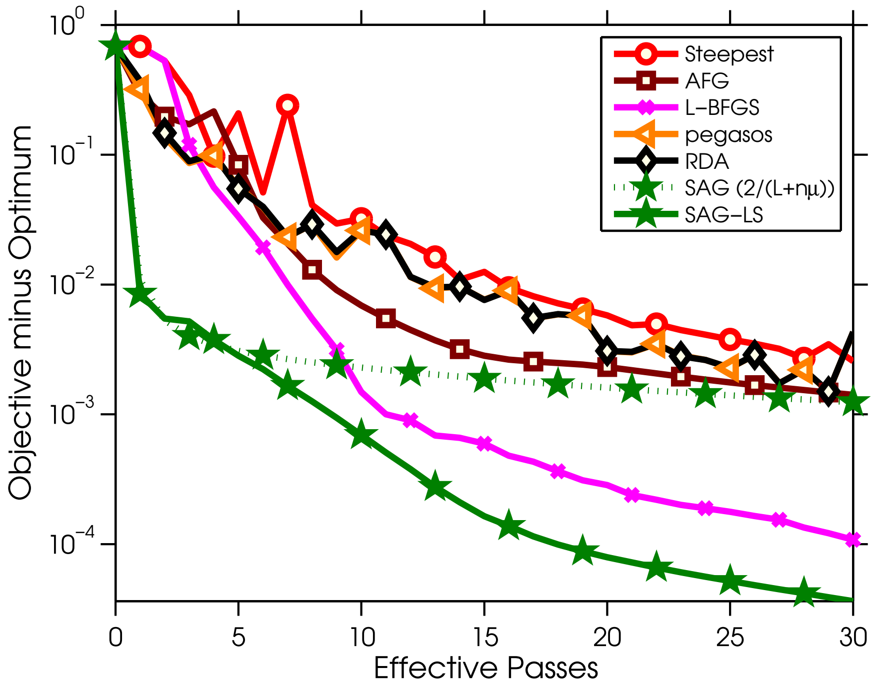
- The algorithm can use sparsity in the features to reduce the storage and iteration cost
- Grouping functions together can further reduce the memory requirement
- We have obtained good performance when R^2 is not known with a heuristic line-search
- Algorithm allows non-uniform sampling
- Possibility of making proximal, coordinate-wise, and Newton-like variants

`spam` dataset (n = 92 189, d = 823 470)

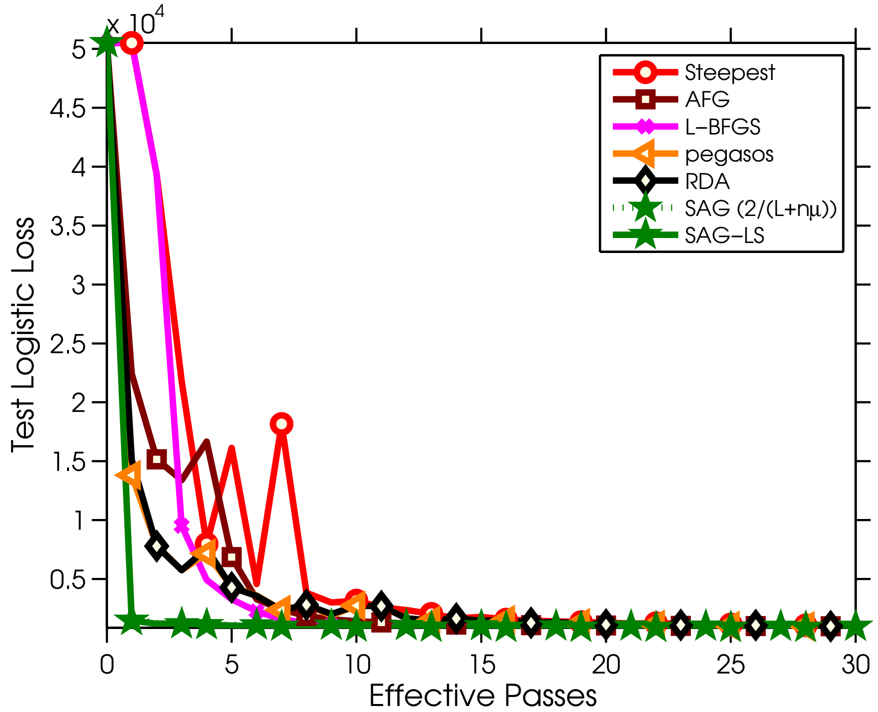


protein dataset ($n = 145751$, $d = 74$)

- Dataset split in two (training/testing)



Training cost



Testing cost

Extensions and related work

- **Exponential convergence rate for strongly convex problems**
- **Need to store gradients**
  - SVRG (Johnson and Zhang, 2013)
- **Adaptivity to non-strong convexity**
  - SAGA (Defazio, Bach, and Lacoste-Julien, 2014)
- **Simple proof**
  - SVRG, SAGA, random coordinate descent (Nesterov, 2012; Shalev-Shwartz and Zhang, 2012)
- **Lower bounds**
  - Agarwal and Bottou (2014)

Variance reduction

- **Principle**: reducing variance of sample of X by using a sample from another random variable Y with known expectation

$$Z_\alpha = \alpha(X - Y) + \mathbb{E}Y$$

  - $\mathbb{E}Z_\alpha = \alpha\mathbb{E}X + (1-\alpha)\mathbb{E}Y$
  - $\operatorname{var} Z_\alpha = \alpha^2\big[\operatorname{var} X + \operatorname{var} Y - 2\operatorname{cov}(X,Y)\big]$
  - $\alpha = 1$: no bias, $\alpha < 1$: potential bias (but reduced variance)
  - Useful if Y positively correlated with X

Variance reduction

- **Principle**: reducing variance of sample of X by using a sample from another random variable Y with known expectation

$$Z_\alpha = \alpha(X - Y) + \mathbb{E}Y$$

  – $\mathbb{E}Z_\alpha = \alpha\mathbb{E}X + (1-\alpha)\mathbb{E}Y$
  – $\operatorname{var} Z_\alpha = \alpha^2\big[\operatorname{var} X + \operatorname{var} Y - 2\operatorname{cov}(X,Y)\big]$
  – $\alpha = 1$: no bias, $\alpha < 1$: potential bias (but reduced variance)
  – Useful if Y positively correlated with X

- **Application to gradient estimation : SVRG** (Johnson and Zhang, 2013)

  – Estimating the averaged gradient $g'(\theta) = \frac{1}{n}\sum_{i=1}^n f_i'(\theta)$
  – Using the gradients of a previous iterate $\tilde{\theta}$

Stochastic variance reduced gradient (SVRG)

- **Algorithm divide into "epochs"**

- At each epoch, starting from $\theta_0 = \tilde{\theta}$, perform the iteration
  - Sample i_t uniformly at random
  - Gradient step: $\theta_t = \theta_{t-1} - \gamma \Big[f'_{i_t}(\theta_{t-1}) - f'_{i_t}(\tilde{\theta}) + g'(\tilde{\theta}) \Big]$

- **Proposition**: If each f_i is R^2-smooth and $g = \frac{1}{n}\sum_{i=1}^n f_i$ is μ-strongly convex, then after $k = 20R^2/\mu$ steps and with $\gamma = 1/10R^2$, then $f(\theta) - f(\theta_*)$ is reduced by 10%

Outline - I

1. **Introduction**
   - Large-scale machine learning and optimization
   - Classes of functions (convex, smooth, etc.)
   - Traditional statistical analysis through Rademacher complexity

2. **Classical methods for convex optimization**
   - Smooth optimization (gradient descent, Newton method)
   - Non-smooth optimization (subgradient descent)

3. **Classical stochastic approximation** (not covered)
   - Robbins-Monro algorithm (1951)

Outline - II

4. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds

5. **Smooth stochastic approximation algorithms**
   - Non-asymptotic analysis for smooth functions
   - Least-squares regression without decaying step-sizes

6. **Finite data sets**
   - Gradient methods with exponential convergence rates

Subgradient descent for machine learning

- **Assumptions** (f is the expected risk, $\hat{f}$ the empirical risk)
  - "Linear" predictors: $\theta(x) = \theta^\top \Phi(x)$, with $\|\Phi(x)\|_2 \leqslant R$ a.s.
  - $\hat{f}(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \Phi(x_i)^\top \theta)$
  - G-Lipschitz loss: f and $\hat{f}$ are GR-Lipschitz on $\Theta = \{\|\theta\|_2 \leqslant D\}$
- **Statistics**: with probability greater than $1-\delta$

$$\sup_{\theta \in \Theta} |\hat{f}(\theta) - f(\theta)| \leqslant \frac{GRD}{\sqrt{n}}\left[2 + \sqrt{2 \log \frac{2}{\delta}}\,\right]$$

- **Optimization**: after t iterations of subgradient method

$$\hat{f}(\hat{\theta}) - \min_{\eta \in \Theta} \hat{f}(\eta) \leqslant \frac{GRD}{\sqrt{t}}$$

- $t = n$ iterations, with total running-time complexity of $O(n^2 d)$

Stochastic subgradient "descent"/method

- **Assumptions**
  - f_n convex and B-Lipschitz-continuous on $\{\|\theta\|_2 \leqslant D\}$
  - (f_n) i.i.d. functions such that $\mathbb{E} f_n = f$
  - θ_* global optimum of f on $\{\|\theta\|_2 \leqslant D\}$

- **Algorithm**: $\theta_n = \Pi_D\left(\theta_{n-1} - \frac{2D}{B\sqrt{n}} f_n'(\theta_{n-1})\right)$

- **Bound**:

$$\mathbb{E} f\left(\frac{1}{n}\sum_{k=0}^{n-1}\theta_k\right) - f(\theta_*) \leqslant \frac{2DB}{\sqrt{n}}$$

- "Same" three-line proof as in the deterministic case

- Minimax rate (Nemirovsky and Yudin, 1983; Agarwal et al., 2012)

- Running-time complexity: $O(dn)$ after n iterations

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: $\frac{\operatorname{tr} H(\theta_*)^{-1}}{n} + \mu^{-1}O(n^{-2\alpha} + n^{-2+\alpha}) + O\Big(\frac{\|\theta_0 - \theta_*\|^2}{\mu^2 n^2}\Big)$

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 $\theta \in \mathbb{R}^d$
  - 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: $\boxed{\mathbb{E} f(\bar{\theta}_{n-1}) - f(\theta_*) \leqslant \frac{4\sigma^2 d}{n} + \frac{4R^2\|\theta_0 - \theta_*\|^2}{n}}$

- **Matches statistical lower bound** (Tsybakov, 2003)
  - Non-asymptotic robust version of Györfi and Walk (1996)

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 $O(d/n)$ 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 (yet) but best practical behavior
  - Note (dis)similarity with regular SGD: $\theta_n = \theta_{n-1} - \gamma f_n'(\theta_{n-1})$

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 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

Summary of rates of convergence

- Problem parameters
  - D diameter of the domain
  - B Lipschitz-constant
  - L smoothness constant
  - μ strong convexity constant

	convex	strongly convex
nonsmooth	deterministic: $BD/\sqrt{t}$ stochastic: $BD/\sqrt{n}$	deterministic: $B^2/(t\mu)$ stochastic: $B^2/(n\mu)$
smooth	deterministic: LD^2/t^2 stochastic: $LD^2/\sqrt{n}$ finite sum: n/t	deterministic: $\exp(-t\sqrt{\mu/L})$ stochastic: $L/(n\mu)$ finite sum: $\exp(-\min\{1/n, \mu/L\}t)$
quadratic	deterministic: LD^2/t^2 stochastic: $d/n + LD^2/n$	deterministic: $\exp(-t\sqrt{\mu/L})$ stochastic: $d/n + LD^2/n$

Conclusions
Machine learning and convex optimization

- **Statistics with or without optimization?**
  - Significance of mixing algorithms with analysis
  - Benefits of mixing algorithms with analysis

- **Open problems**
  - Non-parametric stochastic approximation
  - Characterization of implicit regularization of online methods
  - Structured prediction
  - Going beyond a single pass over the data (testing performance)
  - Further links between convex optimization and online learning/bandits
  - Parallel and distributed optimization

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.

Alekh Agarwal and Leon Bottou. A lower bound for the optimization of finite sums. *arXiv preprint arXiv:1410.0723*, 2014.

R. Aguech, E. Moulines, and P. Priouret. On a perturbation approach for the analysis of stochastic tracking algorithms. *SIAM J. Control and Optimization*, 39(3):872–899, 2000.

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-asymptotic analysis of stochastic approximation algorithms for machine learning. In *Adv. NIPS*, 2011.

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

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

Francis Bach, Rodolphe Jenatton, Julien Mairal, and Guillaume Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends® in Machine Learning*, 4(1):1–106, 2012b.

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

Albert Benveniste, Michel Métivier, and Pierre Priouret. *Adaptive algorithms and stochastic approximations*. Springer Publishing Company, Incorporated, 2012.

D. P. Bertsekas. *Nonlinear programming*. Athena scientific, 1999.

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.

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.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2003.

S. Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3-4):231–357, 2015. ISSN 1935-8237. doi: 10.1561/2200000050. URL http://dx.doi.org/10.1561/2200000050.

N. Cesa-Bianchi, A. Conconi, and C. Gentile. On the generalization ability of on-line learning algorithms. *Information Theory, IEEE Transactions on*, 50(9):2050–2057, 2004.

A. d'Aspremont. Smooth optimization with approximate gradient. *SIAM J. Optim.*, 19(3):1171–1183, 2008.

Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, pages 1646–1654, 2014.

A. Défossez and F. Bach. Constant step size least-mean-square: Bias-variance trade-offs and optimal sampling distributions. 2015.

A. Dieuleveut and F. Bach. Non-parametric Stochastic Approximation with Large Step sizes. Technical report, ArXiv, 2014.

A. Dieuleveut, N. Flammarion, and F. Bach. Harder, better, faster, stronger convergence rates for least-squares regression. Technical Report 1602.05419, arXiv, 2016.

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.

N. Flammarion and F. Bach. From averaging to acceleration, there is only a step-size. *arXiv preprint arXiv:1504.01577*, 2015.

Saeed Ghadimi and Guanghui Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23(4):2061–2089, 2013.

L. Györfi and H. Walk. On the averaged stochastic approximation for linear regression. *SIAM Journal on Control and Optimization*, 34(1):31–61, 1996.

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

Chonghai Hu, James T Kwok, and Weike Pan. Accelerated gradient methods for stochastic optimization and online learning. In *NIPS*, volume 22, pages 781–789, 2009.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pages 315–323, 2013.

Takafumi Kanamori and Hidetoshi Shimodaira. Active learning algorithm using the maximum weighted log-likelihood estimator. *Journal of statistical planning and inference*, 116(1):149–162, 2003.

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

S. Lacoste-Julien, M. Schmidt, and F. Bach. A simpler approach to obtaining an o (1/t) convergence rate for projected stochastic subgradient descent. Technical Report 1212.2002, ArXiv, 2012.

Simon Lacoste-Julien, Martin Jaggi, Mark Schmidt, and Patrick Pletscher. Block-coordinate {Frank-Wolfe} optimization for structural {SVMs}. In *Proceedings of The 30th International Conference on Machine Learning*, pages 53–61, 2013.

G. Lan. An optimal method for stochastic composite optimization. *Math. Program.*, 133(1-2, Ser. A): 365–397, 2012.

Guanghui Lan, Arkadi Nemirovski, and Alexander Shapiro. Validation analysis of mirror descent stochastic approximation method. *Mathematical programming*, 134(2):425–458, 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. 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. 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. Nemirovsky and D. B. Yudin. *Problem complexity and method efficiency in optimization.* Wiley & Sons, 1983.

Y. Nesterov. A method for solving a convex programming problem with rate of convergence $O(1/k^2)$. *Soviet Math. Doklady*, 269(3):543–547, 1983.

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

Y. Nesterov. Gradient methods for minimizing composite objective function. *Center for Operations Research and Econometrics (CORE), Catholic University of Louvain, Tech. Rep*, 76, 2007.

Y. Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical programming*, 120 (1):221–259, 2009.

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.

Yu Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

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 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 *proc. COLT*, 2009.

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

Naum Zuselevich Shor, Krzysztof C. Kiwiel, and Andrzej Ruszcay?ski. *Minimization methods for non-differentiable functions*. Springer-Verlag New York, Inc., 1985.

K. Sridharan, N. Srebro, and S. Shalev-Shwartz. Fast rates for regularized objectives. 2008.

I. Tsochantaridis, Thomas Joachims, T., Y. Altun, and Y. Singer. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005.

A. B. Tsybakov. Optimal rates of aggregation. 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.