Large-scale machine learning and convex optimization

Francis Bach

INRIA - Ecole Normale Supérieure, Paris, France

Allerton Conference - September 2015

Slides available at `www.di.ens.fr/~fbach/gradsto_allerton.pdf`

Context

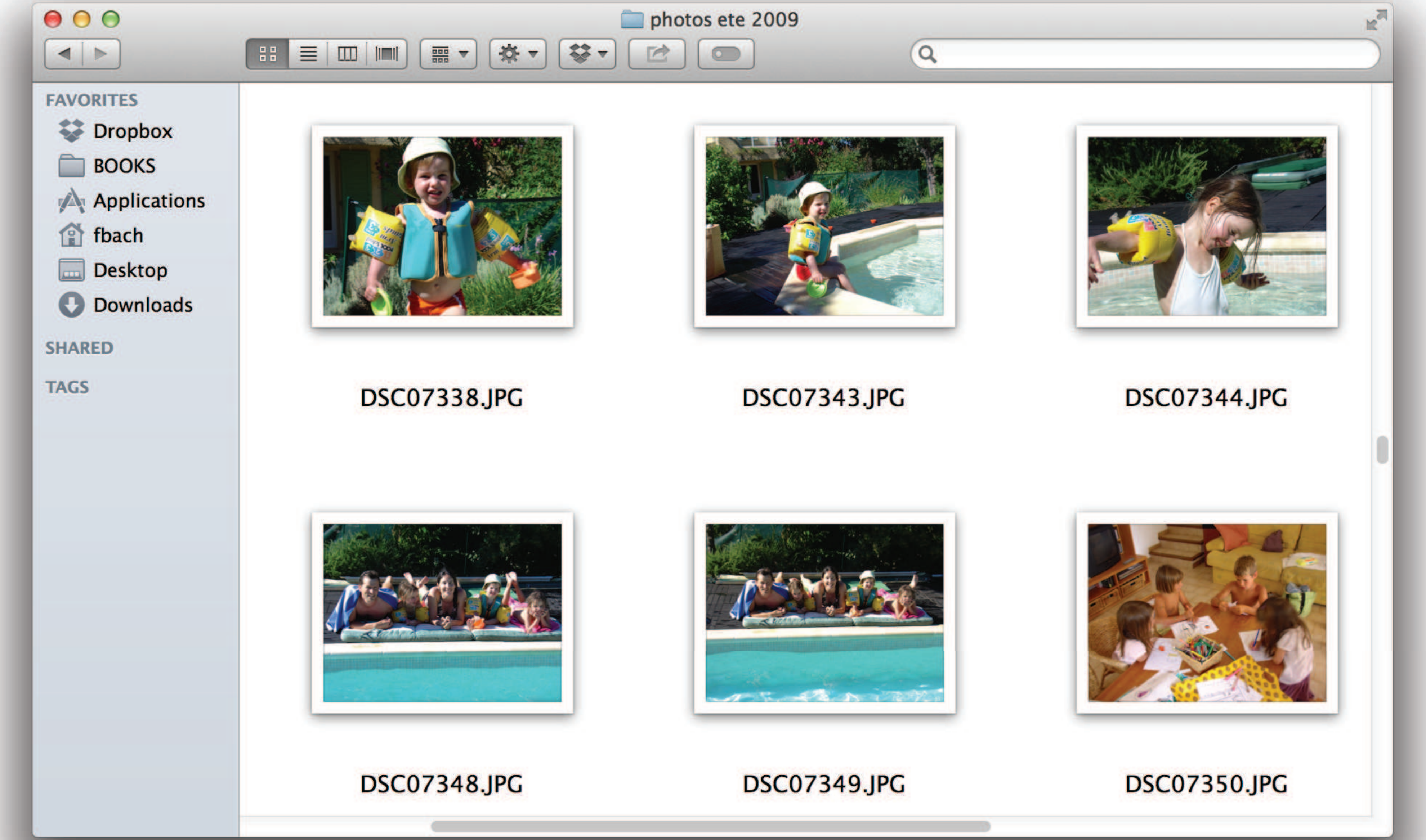
Machine learning for “big data”

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

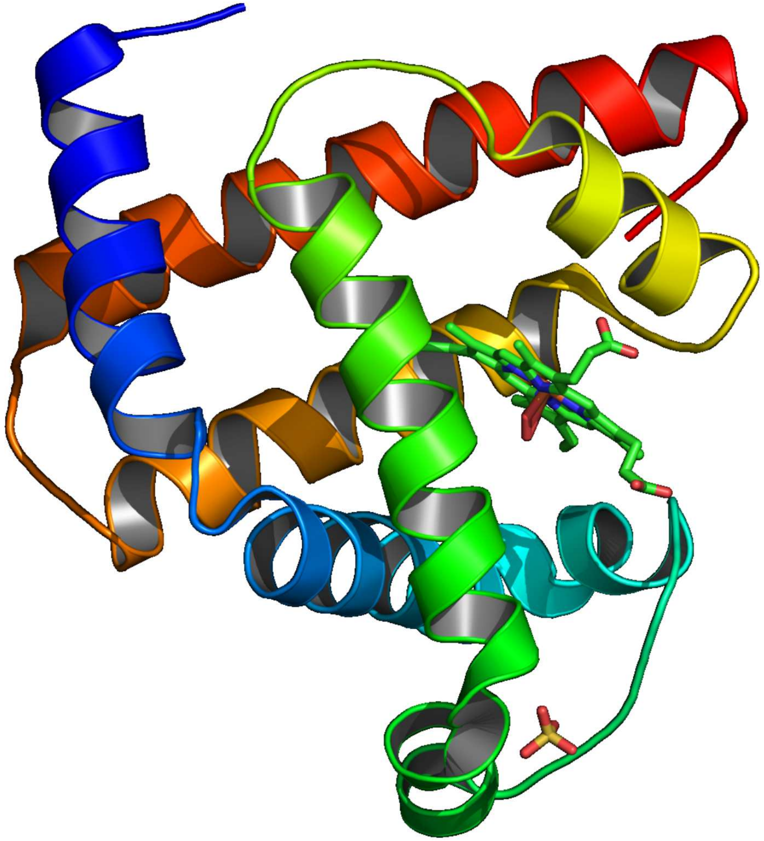
- **Examples**: computer vision, bioinformatics, text processing

Visual object recognition

Personal photos

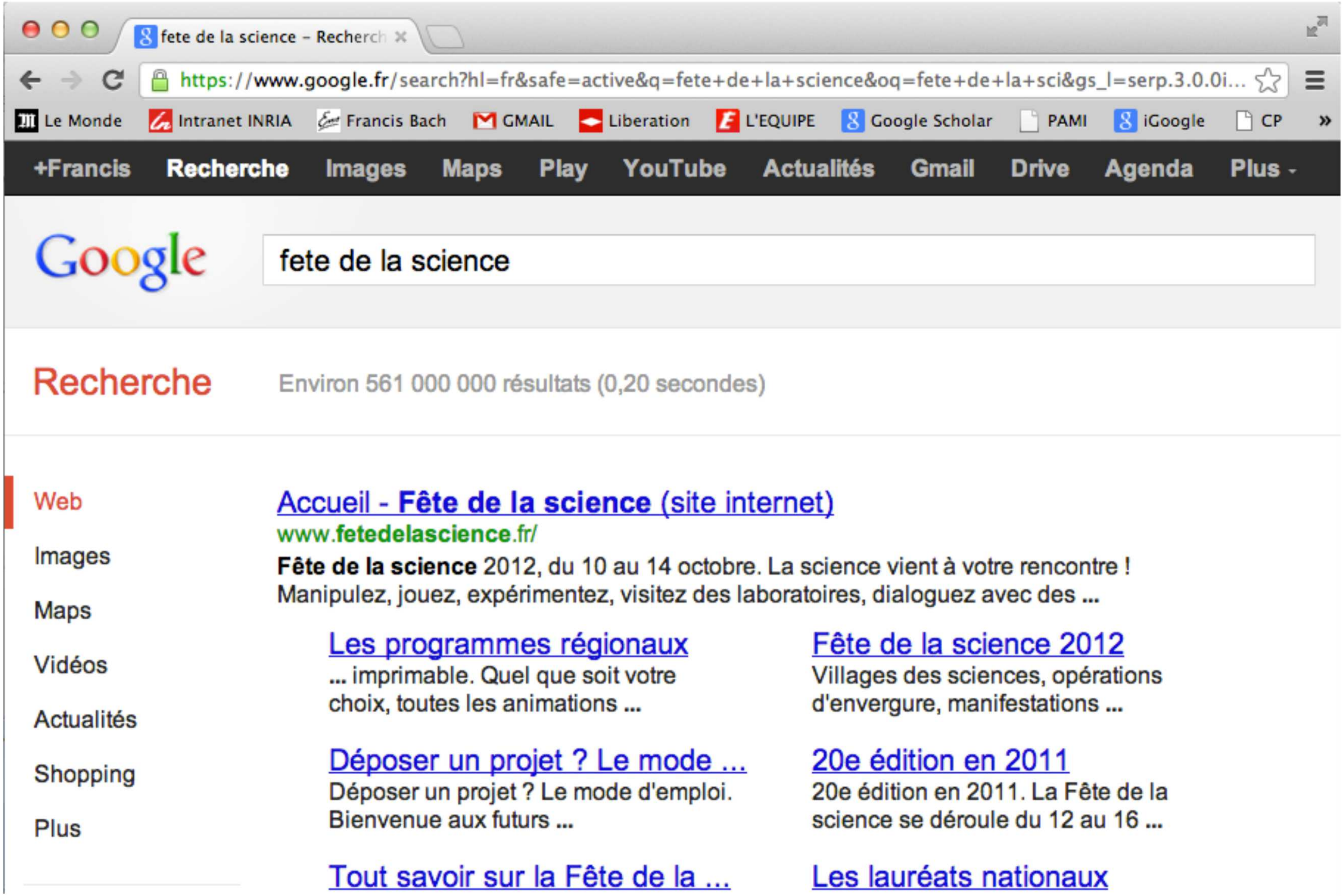


Learning for bioinformatics - Proteins



- Crucial components of cell life
- Predicting multiple functions and interactions
- **Massive data**: up to 1 millions for humans!
- **Complex data**
  - Amino-acid sequence
  - Link with DNA
  - Tri-dimensional molecule

Search engines - advertising



fete de la science - Recherch ×

https://www.google.fr/search?hl=fr&safe=active&q=fete+de+la+science&oq=fete+de+la+sci&gs_l=serp.3.0.0i...

fete de la science

Recherche

Environ 561 000 000 résultats (0,20 secondes)

Accueil - **Fête de la science** (site internet)
www.**fetedelascience**.fr/
Fête de la science 2012, du 10 au 14 octobre. La science vient à votre rencontre ! Manipulez, jouez, expérimentez, visitez des laboratoires, dialoguez avec des ...

Les programmes régionaux
... imprimable. Quel que soit votre choix, toutes les animations ...

Fête de la science 2012
Villages des sciences, opérations d'envergure, manifestations ...

Déposer un projet ? Le mode ...
Déposer un projet ? Le mode d'emploi. Bienvenue aux futurs ...

20e édition en 2011
20e édition en 2011. La Fête de la science se déroule du 12 au 16 ...

Tout savoir sur la Fête de la ...

Les lauréats nationaux

Context

Machine learning for “big data”

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

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

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

Context
Machine learning for “big data”

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

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

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

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

Outline

1. **Large-scale machine learning and optimization**
    - Traditional statistical analysis
    - Classical methods for convex optimization

2. **Non-smooth stochastic approximation**
    - Stochastic (sub)gradient and averaging
    - Non-asymptotic results and lower bounds
    - Strongly convex vs. non-strongly convex

3. **Smooth stochastic approximation algorithms**
    - Asymptotic and non-asymptotic results
    - Beyond decaying step-sizes

4. **Finite data sets**

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

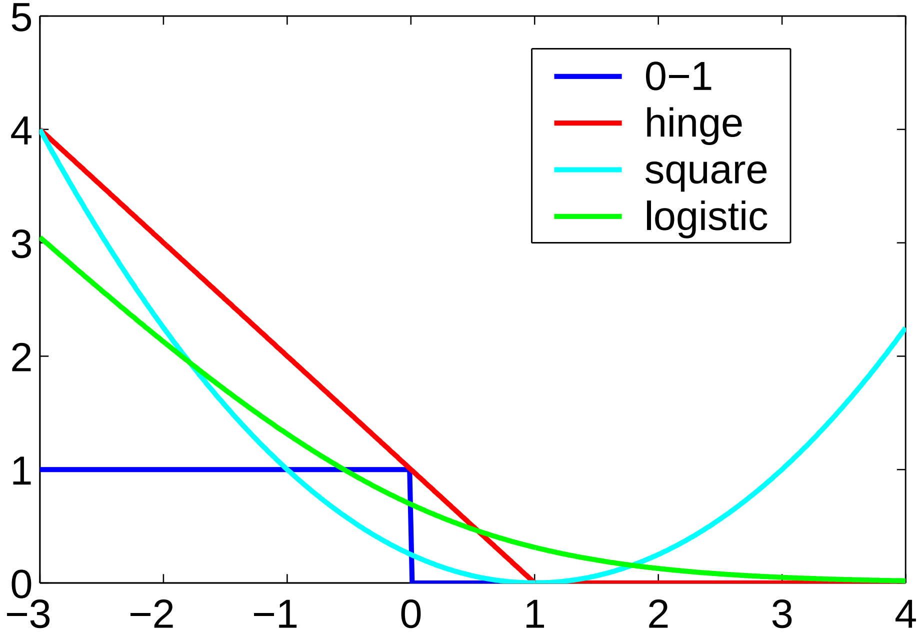
Usual losses

- **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)

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

- **Logistic regression**

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

- **Least-squares regression**

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

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

Lipschitz continuity

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

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

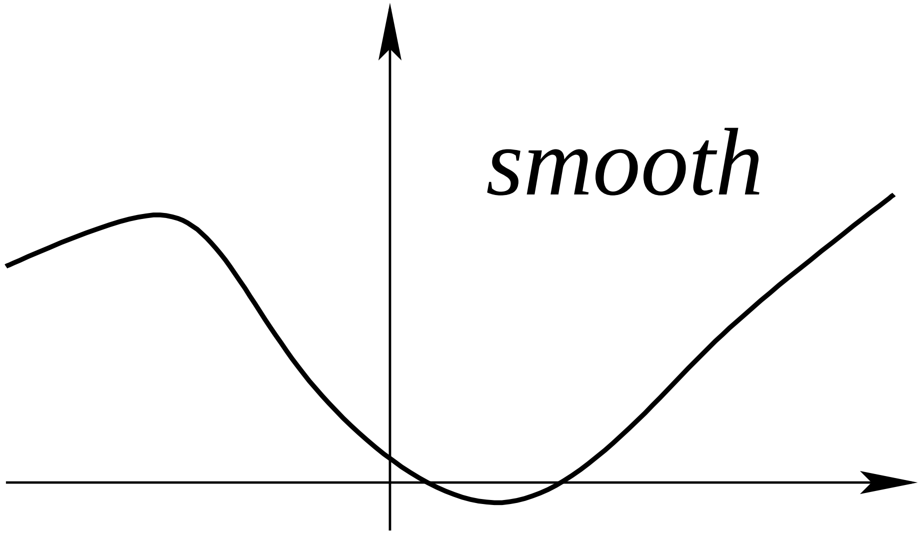
- **Machine learning**
  - with $f(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - G-Lipschitz loss and R-bounded data: $B = GR$

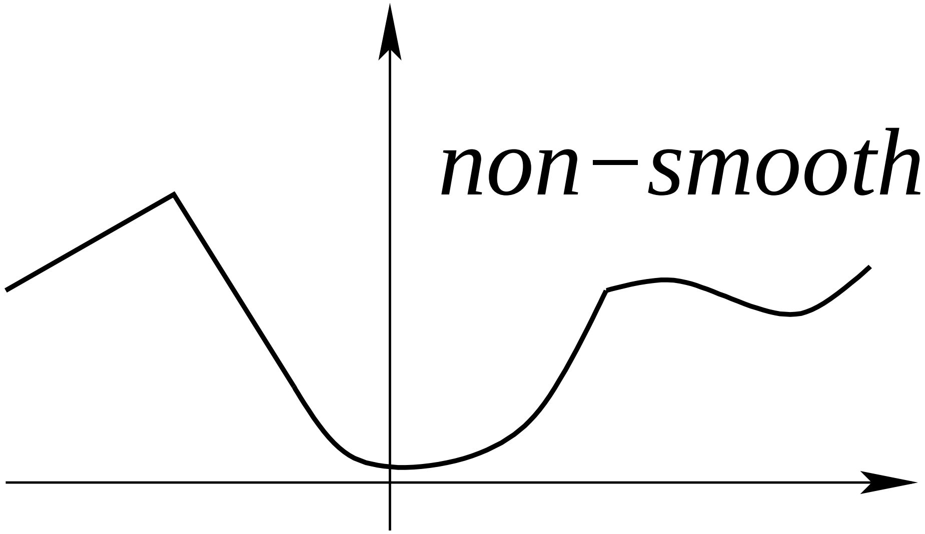
Smoothness and strong convexity

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

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

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





Smoothness and strong convexity

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

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

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

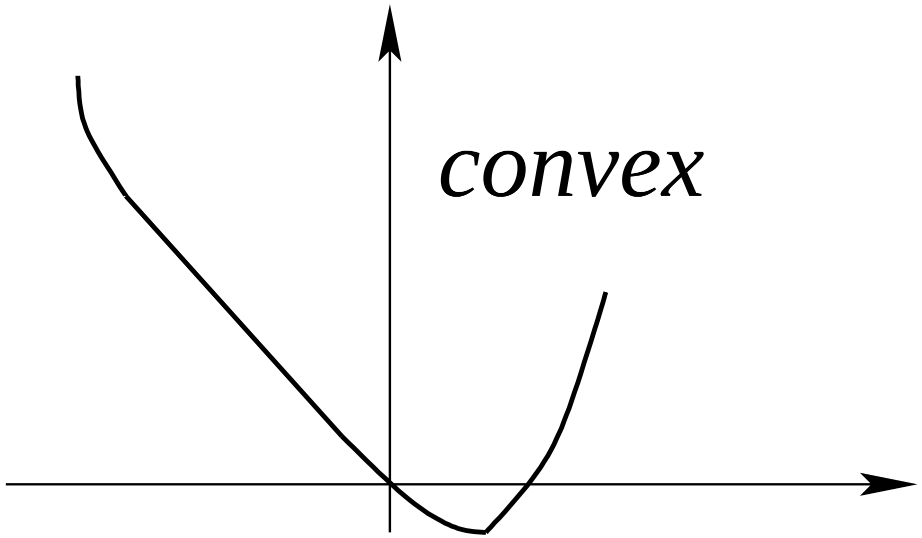
- **Machine learning**
  - with $f(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - Hessian $\approx$ covariance matrix $\frac{1}{n}\sum_{i=1}^n \Phi(x_i)\Phi(x_i)^\top$
  - ℓ-smooth loss and R-bounded data: $L = \ell R^2$

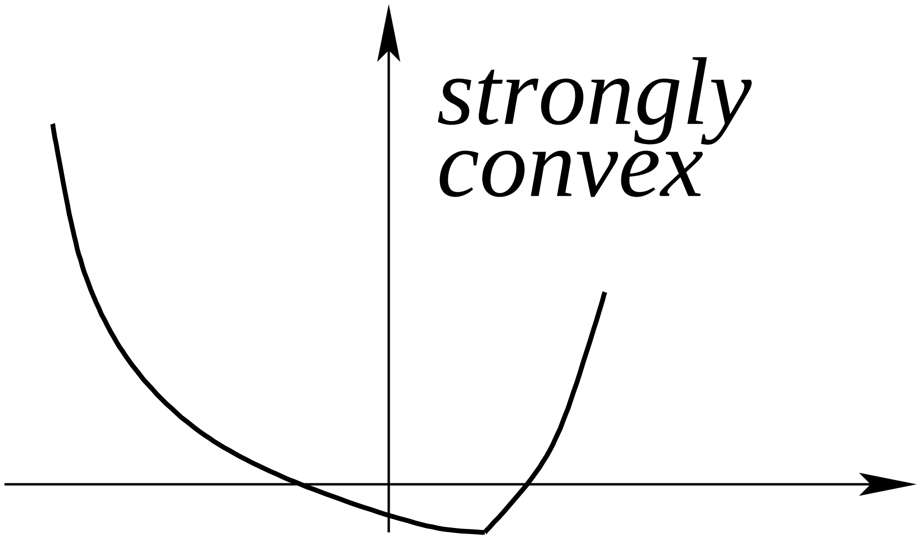
Smoothness and strong convexity

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

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

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





Smoothness and strong convexity

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

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

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

- **Machine learning**
  - with $f(\theta) = \frac{1}{n}\sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - Hessian ≈ 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 $f : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex if and only if

$$\forall \theta_1, \theta_2 \in \mathbb{R}^d, \;\; f(\theta_1) \geqslant f(\theta_2) + f'(\theta_2)^\top (\theta_1 - \theta_2) + \frac{\mu}{2} \|\theta_1 - \theta_2\|_3^2$$

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

- **Machine learning**
  - with $f(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i, \theta^\top \Phi(x_i))$
  - 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 f (Lipschitz-continuity)**: the function f if convex, differentiable and has (sub)gradients uniformly bounded by B on the ball of center 0 and radius D:

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

- **Smoothness of f**: the function f is convex, differentiable with L-Lipschitz-continuous gradient f':

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

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

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

Analysis of empirical risk minimization

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

$$f(\hat{\theta}) - \min_{\theta \in \mathbb{R}^d} f(\theta) = \underbrace{\left[f(\hat{\theta}) - \min_{\theta \in \Theta} f(\theta) \right]}_{\text{Estimation error}} + \underbrace{\left[\min_{\theta \in \Theta} f(\theta) - \min_{\theta \in \mathbb{R}^d} f(\theta) \right]}_{\text{Approximation error}}$$

  – 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**: $\Theta = \{\theta \in \mathbb{R}^d, \Omega(\theta) \leqslant D\}$

$$f(\hat{\theta}) - \min_{\theta \in \mathbb{R}^d} f(\theta) = \underbrace{\Big[f(\hat{\theta}) - \min_{\theta \in \Theta} f(\theta) \Big]}_{\text{Estimation error}} + \underbrace{\Big[\min_{\theta \in \Theta} f(\theta) - \min_{\theta \in \mathbb{R}^d} f(\theta) \Big]}_{\text{Approximation error}}$$

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

$$\begin{aligned} f(\hat{\theta}) - \min_{\theta \in \Theta} f(\theta) &= \big[f(\hat{\theta}) - \hat{f}(\hat{\theta}) \big] + \big[\hat{f}(\hat{\theta}) - \hat{f}(\theta^*_\Theta) \big] + \big[\hat{f}(\theta^*_\Theta) - f(\theta^*_\Theta) \big] \\ &\leqslant \sup_{\theta \in \Theta} |\hat{f}(\theta) - f(\theta)| + \quad 0 \quad + \sup_{\theta \in \Theta} |\hat{f}(\theta) - f(\theta)| \end{aligned}$$

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

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

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

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

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

Motivation from least-squares

- For least-squares, we have $\ell(y, \theta^\top \Phi(x)) = \frac{1}{2}(y - \theta^\top \Phi(x))^2$, and

$$\begin{aligned}
f(\theta) - \hat{f}(\theta) &= \frac{1}{2}\theta^\top \bigg(\frac{1}{n}\sum_{i=1}^n \Phi(x_i)\Phi(x_i)^\top - \mathbb{E}\Phi(X)\Phi(X)^\top \bigg)\theta \\
&\quad -\theta^\top \bigg(\frac{1}{n}\sum_{i=1}^n y_i \Phi(x_i) - \mathbb{E} Y \Phi(X) \bigg) + \frac{1}{2}\bigg(\frac{1}{n}\sum_{i=1}^n y_i^2 - \mathbb{E} Y^2 \bigg), \\
\sup_{\|\theta\|_2 \leqslant D} |f(\theta) - \hat{f}(\theta)| &\leqslant \frac{D^2}{2}\bigg\| \frac{1}{n}\sum_{i=1}^n \Phi(x_i)\Phi(x_i)^\top - \mathbb{E}\Phi(X)\Phi(X)^\top \bigg\|_{\rm op} \\
&\quad + D\bigg\| \frac{1}{n}\sum_{i=1}^n y_i \Phi(x_i) - \mathbb{E} Y \Phi(X) \bigg\|_2 + \frac{1}{2}\bigg| \frac{1}{n}\sum_{i=1}^n y_i^2 - \mathbb{E} Y^2 \bigg|, \\
\sup_{\|\theta\|_2 \leqslant D} |f(\theta) - \hat{f}(\theta)| &\leqslant O(1/\sqrt{n}) \text{ with high probability}
\end{aligned}$$

Symmetrization with Rademacher variables

- Let $\mathcal{D}' = \{x'_1, y'_1, \dots, x'_n, y'_n\}$ an independent copy of the data $\mathcal{D} = \{x_1, y_1, \dots, x_n, y_n\}$, with corresponding loss functions $f'_i(\theta)$

$$
\begin{aligned}
\mathbb{E}\big[\sup_{\theta\in\Theta}\big|f(\theta) - \hat{f}(\theta)\big|\big] &= \mathbb{E}\big[\sup_{\theta\in\Theta}\bigg(f(\theta) - \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)\bigg)\big] \\
&= \mathbb{E}\bigg[\sup_{\theta\in\Theta}\bigg|\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\big(f'_i(\theta) - f_i(\theta)|\mathcal{D}\big)\bigg|\bigg] \\
&\leqslant \mathbb{E}\bigg[\mathbb{E}\bigg[\sup_{\theta\in\Theta}\bigg|\frac{1}{n}\sum_{i=1}^{n}\big(f'_i(\theta) - f_i(\theta)\bigg|\ \bigg|\mathcal{D}\bigg]\bigg] \\
&= \mathbb{E}\bigg[\sup_{\theta\in\Theta}\bigg|\frac{1}{n}\sum_{i=1}^{n}\big(f'_i(\theta) - f_i(\theta)\big)\bigg|\bigg] \\
&= \mathbb{E}\bigg[\sup_{\theta\in\Theta}\bigg|\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i\big(f'_i(\theta) - f_i(\theta)\big)\bigg|\bigg] \quad \text{with } \varepsilon_i \text{ uniform in } \{-1, 1\} \\
&\leqslant 2\mathbb{E}\bigg[\sup_{\theta\in\Theta}\bigg|\frac{1}{n}\sum_{i=1}^{n}\varepsilon_i f_i(\theta)\bigg|\bigg] \quad = \text{Rademacher complexity}
\end{aligned}
$$

Rademacher complexity

- Define the Rademacher complexity of the class of functions $(X, Y) \mapsto \ell(Y, \theta^\top \Phi(X))$ as

$$R_n = \mathbb{E}\bigg[\sup_{\theta \in \Theta} \bigg| \frac{1}{n} \sum_{i=1}^n \varepsilon_i f_i(\theta) \bigg| \bigg].$$

- Note two expectations, with respect to $\mathcal{D}$ *and* with respect to ε

- **Main property**:

$$\mathbb{E}\big[\sup_{\theta \in \Theta} \big| f(\theta) - \hat{f}(\theta) \big| \big] \leqslant 2 R_n$$

From Rademacher complexity to bound in high probability

- Let $Z = \sup_{\theta \in \Theta} \left| f(\theta) - \hat{f}(\theta) \right|$

- By changing the pair (x_i, y_i), Z may only change by

$$\frac{2}{n} \sup |\ell(Y, \theta^\top \Phi(X))| \leqslant \frac{2}{n}\big(\sup |\ell(Y,0)| + GRD\big) \leqslant \frac{2}{n}\big(\ell_0 + GRD\big) = c$$

with $\sup |\ell(Y,0)| = \ell_0$

- **MacDiarmid inequality**: with probability greater than $1 - \delta$,

$$Z \leqslant \mathbb{E}Z + \sqrt{\frac{n}{2}}c \cdot \sqrt{\log \frac{1}{\delta}} \leqslant 2R_n + \frac{\sqrt{2}}{\sqrt{n}}\big(\ell_0 + GRD\big)\sqrt{\log \frac{1}{\delta}}$$

Bounding the Rademacher average - I

- We have, with $\varphi_i(u) = \ell(y_i,u) - \ell(y_i,0)$ is almost surely B-Lipschitz:

$$\begin{aligned}
R_n &= \mathbb{E}\Big[\sup_{\theta\in\Theta}\Big|\frac{1}{n}\sum_{i=1}^n \varepsilon_i f_i(\theta)\Big|\Big] \\
&\leqslant \mathbb{E}\Big[\sup_{\theta\in\Theta}\Big|\frac{1}{n}\sum_{i=1}^n \varepsilon_i f_i(0)\Big|\Big] + \mathbb{E}\Big[\sup_{\theta\in\Theta}\Big|\frac{1}{n}\sum_{i=1}^n \varepsilon_i \big[f_i(\theta) - f_i(0)\big]\Big|\Big] \\
&\leqslant \frac{\ell_0}{\sqrt{n}} + \mathbb{E}\Big[\sup_{\theta\in\Theta}\frac{1}{n}\sum_{i=1}^n \varepsilon_i \big[f_i(\theta) - f_i(0)\big]\Big] \\
&= \frac{\ell_0}{\sqrt{n}} + \mathbb{E}\Big[\sup_{\theta\in\Theta}\frac{1}{n}\sum_{i=1}^n \varepsilon_i \varphi_i(\theta^\top\Phi(x_i))\Big]
\end{aligned}$$

- Using Ledoux-Talagrand concentration results for Rademacher averages (since φ_i is G-Lipschitz), we get:

$$R_n \leqslant \frac{\ell_0}{\sqrt{n}} + 2G\cdot\mathbb{E}\Big[\sup_{\|\theta\|_2\leqslant D}\Big|\frac{1}{n}\sum_{i=1}^n \varepsilon_i \theta^\top\Phi(x_i)\Big]\Big|$$

Bounding the Rademacher average - II

- We have:

$$
\begin{aligned}
R_n &\leqslant \frac{\ell_0}{\sqrt{n}} + 2G\mathbb{E}\Big[\sup_{\|\theta\|_2 \leqslant D} \Big| \frac{1}{n}\sum_{i=1}^n \varepsilon_i \theta^\top \Phi(x_i)\Big]\Big| \\
&= \frac{\ell_0}{\sqrt{n}} + 2G\mathbb{E}\Big\| D\frac{1}{n}\sum_{i=1}^n \varepsilon_i \Phi(x_i)\Big\|_2 \\
&\leqslant \frac{\ell_0}{\sqrt{n}} + 2GD\sqrt{\mathbb{E}\Big\|\frac{1}{n}\sum_{i=1}^n \varepsilon_i \Phi(x_i)\Big\|_2^2} \text{ by Jensen's inequality} \\
&\leqslant \frac{2(\ell_0 + GRD)}{\sqrt{n}} \text{ by using} \|\Phi(x)\|_2 \leqslant R
\end{aligned}
$$

- Overall, we get, with probability $1-\delta$:

$$
\sup_{\theta \in \Theta} \big| f(\theta) - \hat{f}(\theta) \big| \leqslant \frac{1}{\sqrt{n}}(\ell_0 + GRD)(4 + \sqrt{2\log\frac{1}{\delta}})
$$

Putting it all together

- We have, with probability $1 - \delta$
  - For exact minimizer $\hat{\theta} \in \arg\min_{\theta \in \Theta} \hat{f}(\theta)$, we have

$$\begin{aligned} f(\hat{\theta}) - \min_{\theta \in \Theta} f(\theta) &\leqslant 2 \cdot \sup_{\theta \in \Theta} |\hat{f}(\theta) - f(\theta)| \\ &\leqslant \frac{2}{\sqrt{n}} (\ell_0 + GRD)(4 + \sqrt{2 \log \frac{1}{\delta}}) \end{aligned}$$

  - For all $\theta \in \Theta$

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

- **Only need to optimize with precision** $\frac{2}{\sqrt{n}}(\ell_0 + GRD)$

Slow rate for supervised learning (summary)

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

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

- **Lipschitz functions ⇒ 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:
  - $f(\theta) = \frac{1}{2}\mathbb{E}(\theta - z)^2 = \frac{1}{2}(\theta - \mathbb{E}z)^2 + \frac{1}{2}\,\mathrm{var}(z) = \hat{f}(\theta) + O(1/\sqrt{n})$
  - $f(\hat{\theta}) = \frac{1}{2}(\hat{\theta} - \mathbb{E}z)^2 + \frac{1}{2}\,\mathrm{var}(z) = f(\mathbb{E}z) + O(1/\sqrt{n})$

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:

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

- Bound only at $\hat{\theta}$ + strong convexity

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 $\theta \in \mathbb{R}^d$,

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

- 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

Complexity results in convex optimization for ML

- **Assumption**: f convex on $\mathbb{R}^d$

- **Classical generic algorithms**

  - (sub)gradient method/descent
  - Accelerated gradient descent
  - Newton method

- **Key additional properties of f**

  - Lipschitz continuity, smoothness or strong convexity

- **Key insight from Bottou and Bousquet (2008)**

  - In machine learning, no need to optimize below estimation error

- **Key reference**: Nesterov (2004)

Summary of smoothness/convexity assumptions

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

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

- **Smoothness of f**: the function f is convex, differentiable with L-Lipschitz-continuous gradient f':

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

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

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

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

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

- **Algorithm**: $\theta_t = \Pi_D\bigg(\theta_{t-1} - \dfrac{2D}{B\sqrt{t}} f'(\theta_{t-1})\bigg)$
  - Π_D : orthogonal projection onto $\{\|\theta\|_2 \leqslant D\}$

- **Bound**:

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

- Three-line proof (for “constant” step-size)

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

Subgradient method/descent - proof - I

- Iteration: $\theta_t = \Pi_D(\theta_{t-1} - \gamma_t f'(\theta_{t-1}))$ with $\gamma_t = \frac{2D}{B\sqrt{t}}$

- Assumption: $\|f'(\theta)\|_2 \leqslant B$ and $\|\theta\|_2 \leqslant D$

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

- leading to

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

Subgradient method/descent - proof - II

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

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

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

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**
  - f convex and B-Lipschitz-continuous on $\{\|\theta\|_2 \leqslant D\}$
  - f μ-strongly convex
- **Algorithm**: $\theta_t = \Pi_D\Big(\theta_{t-1} - \dfrac{2}{\mu(t+1)} f'(\theta_{t-1})\Big)$
- **Bound**:

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

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

Subgradient method - strong convexity - proof - I

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

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

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

(property of subgradients and strong convexity)

- leading to

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

Subgradient method - strong convexity - proof - II

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

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

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

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

(smooth) gradient descent

- **Assumptions**
  - f convex with L-Lipschitz-continuous gradient
  - Minimum attained at θ_*

- **Algorithm**:

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

- **Bound**:

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

- Three-line proof

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

(smooth) gradient descent - strong convexity

- **Assumptions**
  - f convex with L-Lipschitz-continuous gradient
  - f μ-strongly convex

- **Algorithm**:

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

- **Bound**:

$$f(\theta_t) - f(\theta_*) \leqslant (1 - \mu/L)^t \big[f(\theta_0) - f(\theta_*) \big]$$

- Three-line proof

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

- Line search

Accelerated gradient methods (Nesterov, 1983)

- **Assumptions**
  - f convex with L-Lipschitz-cont. gradient , min. attained at θ_*

- **Algorithm**:

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

- **Bound**:

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

- Ten-line proof

- Not improvable

- Extension to strongly convex 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)

Summary: minimizing convex functions

- **Assumption**: f convex

- **Gradient descent**: $\theta_t = \theta_{t-1} - \gamma_t \, f'(\theta_{t-1})$
  - $O(1/\sqrt{t})$ 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**: $\theta_t = \theta_{t-1} - f''(\theta_{t-1})^{-1} f'(\theta_{t-1})$
  - $O\big(e^{-\rho 2^t}\big)$ convergence rate

Summary: minimizing convex functions

- **Assumption**: f convex

- **Gradient descent**: $\theta_t = \theta_{t-1} - \gamma_t \, f'(\theta_{t-1})$
  - $O(1/\sqrt{t})$ 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**: $\theta_t = \theta_{t-1} - f''(\theta_{t-1})^{-1} f'(\theta_{t-1})$
  - $O\big(e^{-\rho 2^t}\big)$ 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**

Outline

1. **Large-scale machine learning and optimization**
   - Traditional statistical analysis
   - Classical methods for convex optimization

2. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds
   - Strongly convex vs. non-strongly convex

3. **Smooth stochastic approximation algorithms**
   - Asymptotic and non-asymptotic results
   - Beyond decaying step-sizes

4. **Finite data sets**

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)

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

- **Number of iterations = number of observations**
  - "Single" line of code!

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: $\boxed{\gamma_n = Cn^{-\alpha}}$

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 $\{\|\theta\|_2 \leqslant D\}$

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

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

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

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**: $\theta_n = \Pi_D\left(\theta_{n-1} - \frac{2}{\mu(n+1)} f_n'(\theta_{n-1})\right)$

- **Bound**:

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

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

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

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

Outline

1. **Large-scale machine learning and optimization**
   - Traditional statistical analysis
   - Classical methods for convex optimization

2. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds
   - Strongly convex vs. non-strongly convex

3. **Smooth stochastic approximation algorithms**
   - Asymptotic and non-asymptotic results
   - Beyond decaying step-sizes

4. **Finite data sets**

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

Classical proof sketch (no averaging)

$$\begin{aligned}
\|\theta_n - \theta_*\|_2^2 &= \|\theta_{n-1} - \gamma_n f_n'(\theta_{n-1}) - \theta_*\|_2^2 \\
&= \|\theta_{n-1} - \theta_*\|_2^2 - 2\gamma_n(\theta_{n-1} - \theta_*)^\top f_n'(\theta_{n-1}) + \gamma_n^2 \|f_n'(\theta_{n-1})\|_2^2 \\
&\leqslant \|\theta_{n-1} - \theta_*\|_2^2 - 2\gamma_n(\theta_{n-1} - \theta_*)^\top f_n'(\theta_{n-1}) \\
&\qquad + 2\gamma_n^2\|f_n'(\theta_*)\|_2^2 + 2\gamma_n^2\|f_n'(\theta_{n-1}) - f_n'(\theta_*)\|_2^2 \\
&\leqslant \|\theta_{n-1} - \theta_*\|_2^2 - 2\gamma_n(\theta_{n-1} - \theta_*)^\top f_n'(\theta_{n-1}) \\
&\qquad + 2\gamma_n^2\|f_n'(\theta_*)\|_2^2 + 2\gamma_n^2 L[f_n'(\theta_{n-1}) - f_n'(\theta_*)]^\top(\theta_{n-1} - \theta_*) \\
\mathbb{E}\big[\|\theta_n - \theta_*\|_2^2 | \mathcal{F}_{n-1}\big] &\leqslant \|\theta_{n-1} - \theta_*\|_2^2 - 2\gamma_n(\theta_{n-1} - \theta_*)^\top f'(\theta_{n-1}) \\
&\qquad + 2\gamma_n^2 \mathbb{E}\|f_n'(\theta_*)\|_2^2 + 2\gamma_n^2 L[f'(\theta_{n-1}) - 0]^\top(\theta_{n-1} - \theta_*) \\
&\leqslant \|\theta_{n-1} - \theta_*\|_2^2 - 2\gamma_n(1 - \gamma_n L)(\theta_{n-1} - \theta_*)^\top f'(\theta_{n-1}) + 2\gamma_n^2\sigma^2 \\
&\leqslant \|\theta_{n-1} - \theta_*\|_2^2 - 2\gamma_n(1 - \gamma_n L)\frac{1}{2}\mu\|\theta_{n-1} - \theta_*\|_2^2 + 2\gamma_n^2\sigma^2 \\
&= \big[1 - \mu\gamma_n(1 - \gamma_n L)\big]\|\theta_{n-1} - \theta_*\|_2^2 + 2\gamma_n^2\sigma^2 \\
\mathbb{E}\big[\|\theta_{n-1} - \theta_*\|_2^2\big] &\leqslant \big[1 - \mu\gamma_n(1 - \gamma_n L)\big]\mathbb{E}\big[\|\theta_{n-1} - \theta_*\|_2^2\big] + 2\gamma_n^2\sigma^2
\end{aligned}$$

Proof sketch (averaging)

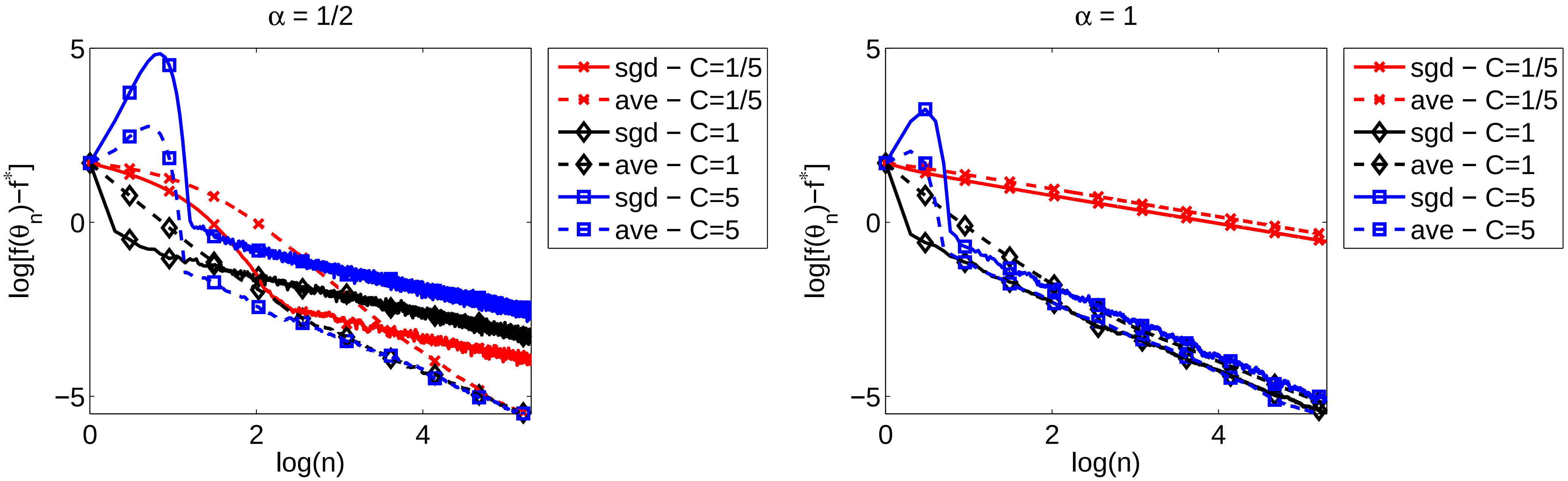
- From Polyak and Juditsky (1992):

$$
\begin{aligned}
& \theta_n = \theta_{n-1} - \gamma_n f_n'(\theta_{n-1}) \\
\Leftrightarrow \quad & f_n'(\theta_{n-1}) = \frac{1}{\gamma_n}(\theta_{n-1} - \theta_n) \\
\Leftrightarrow \quad & f_n'(\theta_*) + f_n''(\theta_*)(\theta_{n-1} - \theta_*) = \frac{1}{\gamma_n}(\theta_{n-1} - \theta_n) + O(\|\theta_{n-1} - \theta_*\|^2) \\
\Leftrightarrow \quad & f_n'(\theta_*) + f''(\theta_*)(\theta_{n-1} - \theta_*) = \frac{1}{\gamma_n}(\theta_{n-1} - \theta_n) + O(\|\theta_{n-1} - \theta_*\|^2) \\
& \qquad\qquad\qquad\qquad\qquad\qquad + O(\|\theta_{n-1} - \theta_*\|)\varepsilon_n \\
\Leftrightarrow \quad & \theta_{n-1} - \theta_* = -f''(\theta_*)^{-1} f_n'(\theta_*) + \frac{1}{\gamma_n} f''(\theta_*)^{-1}(\theta_{n-1} - \theta_n) \\
& \qquad\qquad + O(\|\theta_{n-1} - \theta_*\|^2) + O(\|\theta_{n-1} - \theta_*\|)\varepsilon_n
\end{aligned}
$$

- Averaging to cancel the term $\frac{1}{\gamma_n} f''(\theta_*)^{-1}(\theta_{n-1} - \theta_n)$

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 $\alpha \in [1/3, 1]$ (worse than with averaging)

- **Take-home message**
  - Use $\alpha = 1/2$ with averaging to be adaptive to strong convexity

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)

Outline

1. **Large-scale machine learning and optimization**
   - Traditional statistical analysis
   - Classical methods for convex optimization

2. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds
   - Strongly convex vs. non-strongly convex

3. **Smooth stochastic approximation algorithms**
   - Asymptotic and non-asymptotic results
   - Beyond decaying step-sizes

4. **Finite data sets**

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?**

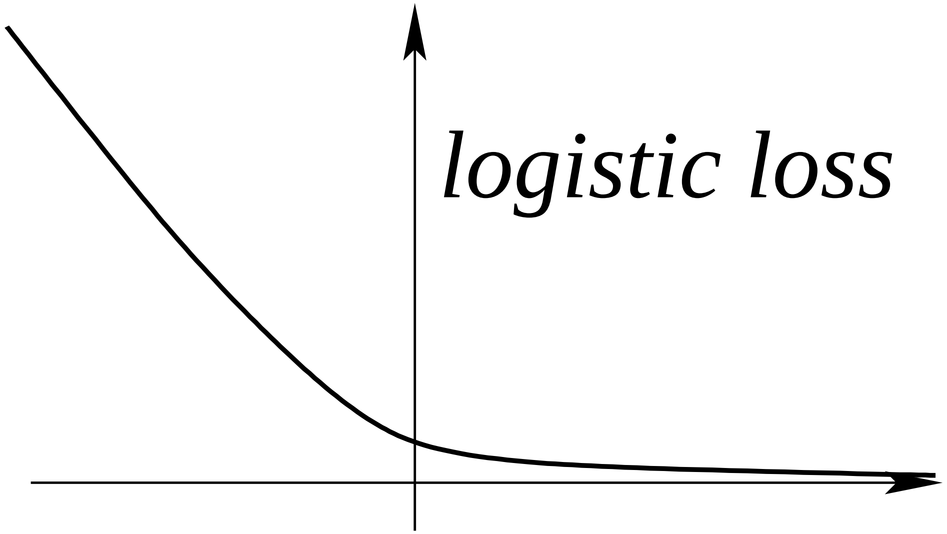
Adaptive algorithm for logistic regression

- **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 $|\theta^\top \Phi(x_n)| \leqslant M$ (and with constants e^M)
  - $\mu =$ lowest eigenvalue of the Hessian at the optimum $f''(\theta_*)$



Adaptive algorithm for logistic regression

- **Logistic regression**: $(\Phi(x_n), y_n) \in \mathbb{R}^d \times \{-1, 1\}$
  - 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** $1/\big(2R^2\sqrt{n}\big)$
  - 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)

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)

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

Adaptive algorithm for logistic regression

Proof sketch

- Step 1: use existing result $f(\bar\theta_n) - f(\theta_*) + \frac{R^2}{\sqrt{n}}\|\theta_0-\theta_*\|_2^2 = O(1/\sqrt{n})$

- Step 2: $f_n'(\theta_{n-1}) = \frac{1}{\gamma}(\theta_{n-1}-\theta_n) \Rightarrow \frac{1}{n}\sum_{k=1}^n f_k'(\theta_{k-1}) = \frac{1}{n\gamma}(\theta_0-\theta_n)$

- Step 3: $\Big\| f'\big(\frac{1}{n}\sum_{k=1}^n \theta_{k-1}\big) - \frac{1}{n}\sum_{k=1}^n f'(\theta_{k-1})\Big\|_2$
  $= O\big(f(\bar\theta_n) - f(\theta_*)\big) = O(1/\sqrt{n})$ using self-concordance

- Step 4a: if f μ-strongly convex, $f(\bar\theta_n) - f(\theta_*) \leqslant \frac{1}{2\mu}\big\|f'(\bar\theta_n)\big\|_2^2$

- Step 4b: if f self-concordant, “locally true” with $\mu = \lambda_{\min}(f''(\theta_*))$

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 $|\theta^\top \Phi(x_n)| \leqslant M$ (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** $1/\big(2R^2\sqrt{n}\big)$
  - 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)

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

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

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

Least-squares - Proof technique

- 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.,

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

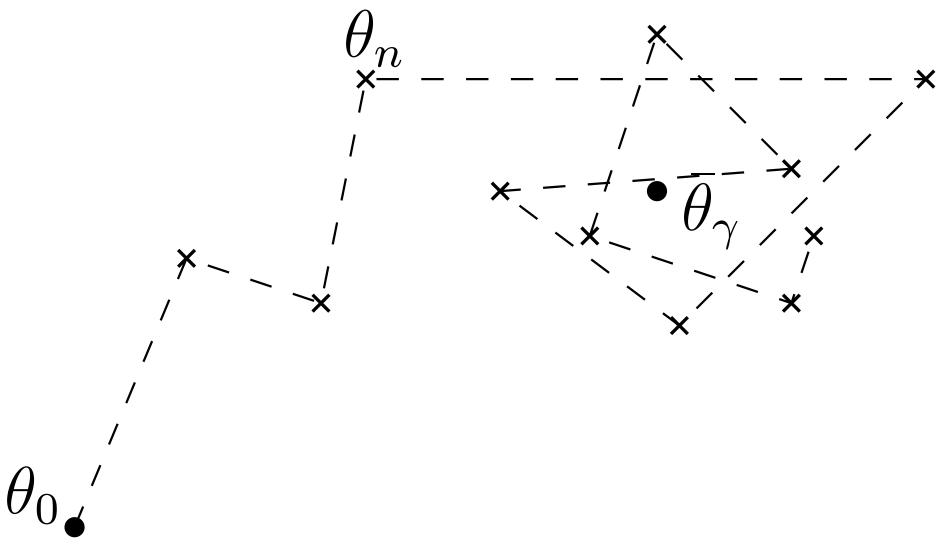
- Infinite expansion of Aguech, Moulines, and Priouret (2000) in powers of γ

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



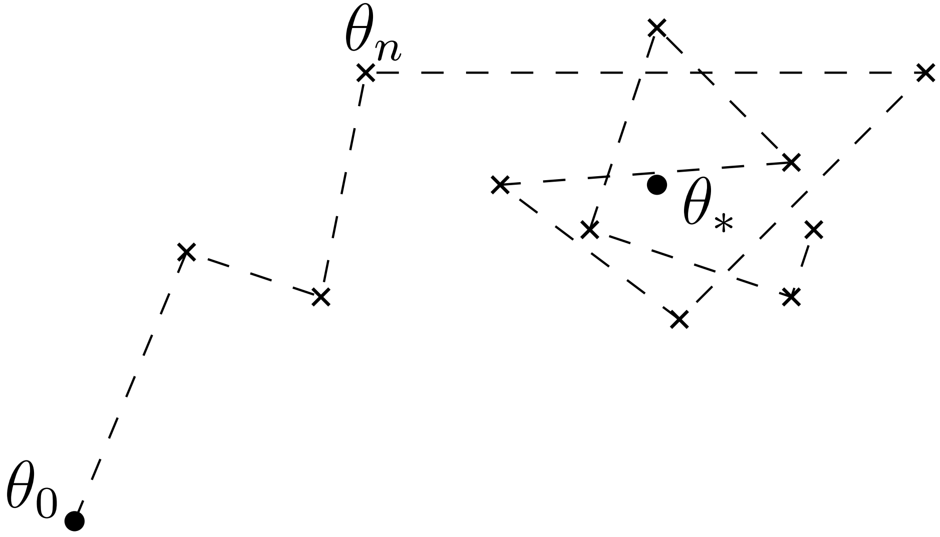
Markov chain interpretation of constant step sizes

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

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



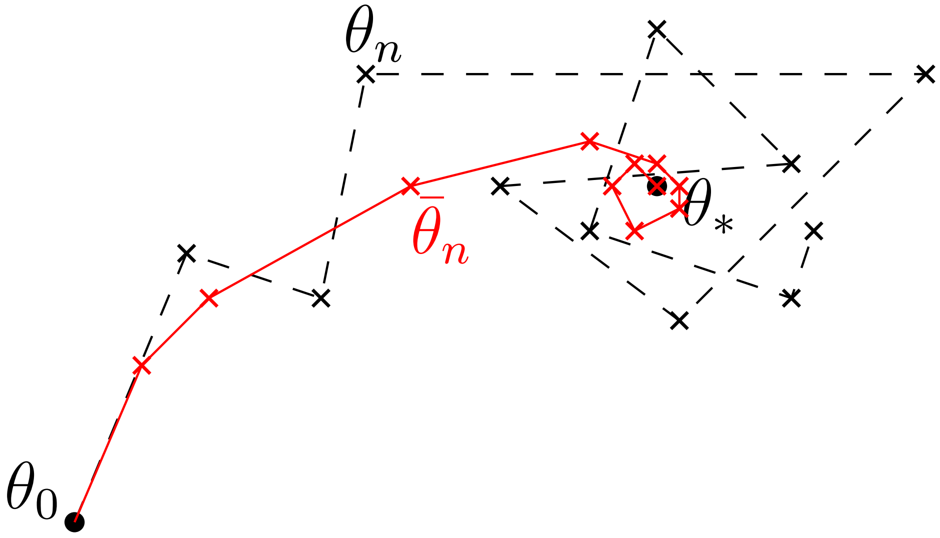
Markov chain interpretation of constant step sizes

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

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



Markov chain interpretation of constant step sizes

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

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

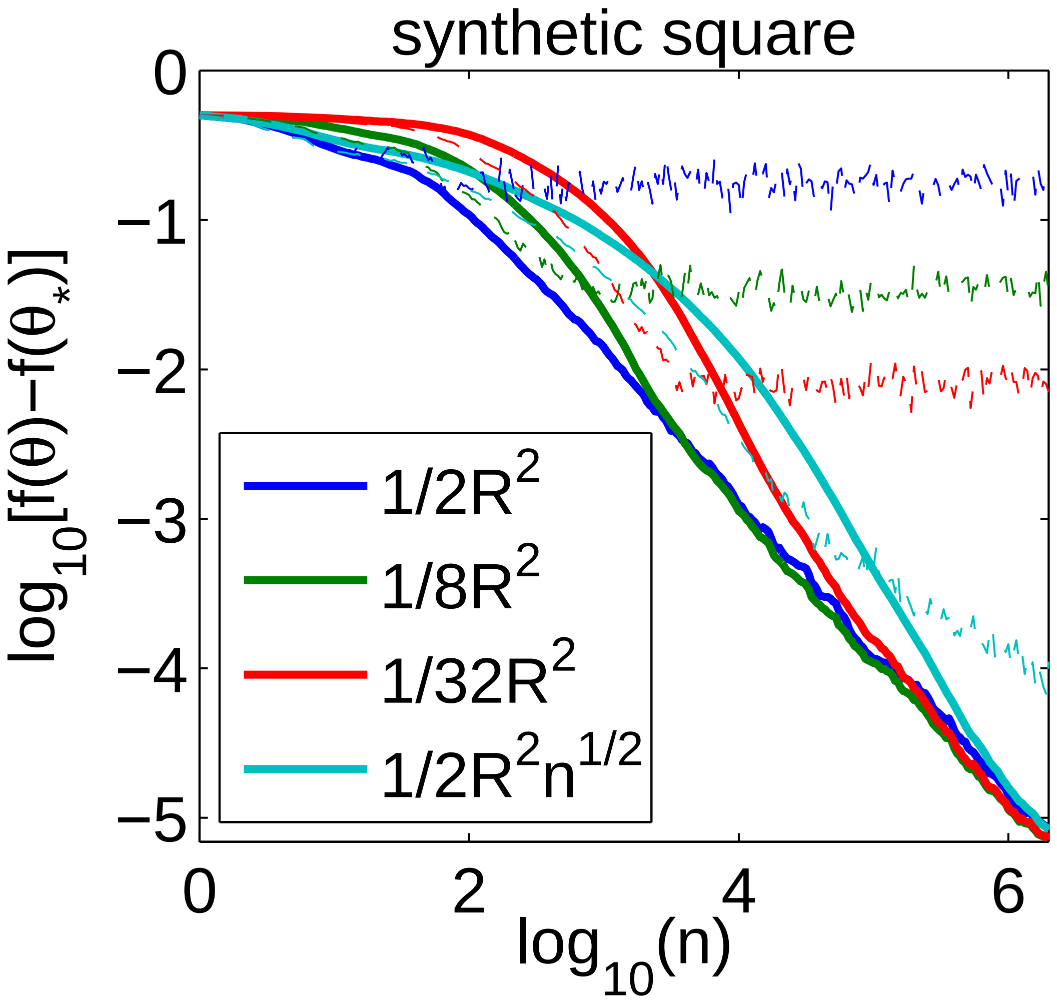
- The sequence $(\theta_n)_n$ is a homogeneous Markov chain
  - convergence to a stationary distribution π_γ
  - with expectation $\bar{\theta}_\gamma \overset{\text{def}}{=} \int \theta \pi_\gamma(\mathrm{d}\theta)$

- **For least-squares,** $\bar{\theta}_\gamma = \theta_*$
  - θ_n does not converge to θ_* but oscillates around it
  - oscillations of order $\sqrt{\gamma}$

- **Ergodic theorem:**
  - Averaged iterates converge to $\bar{\theta}_\gamma = \theta_*$ at rate $O(1/n)$

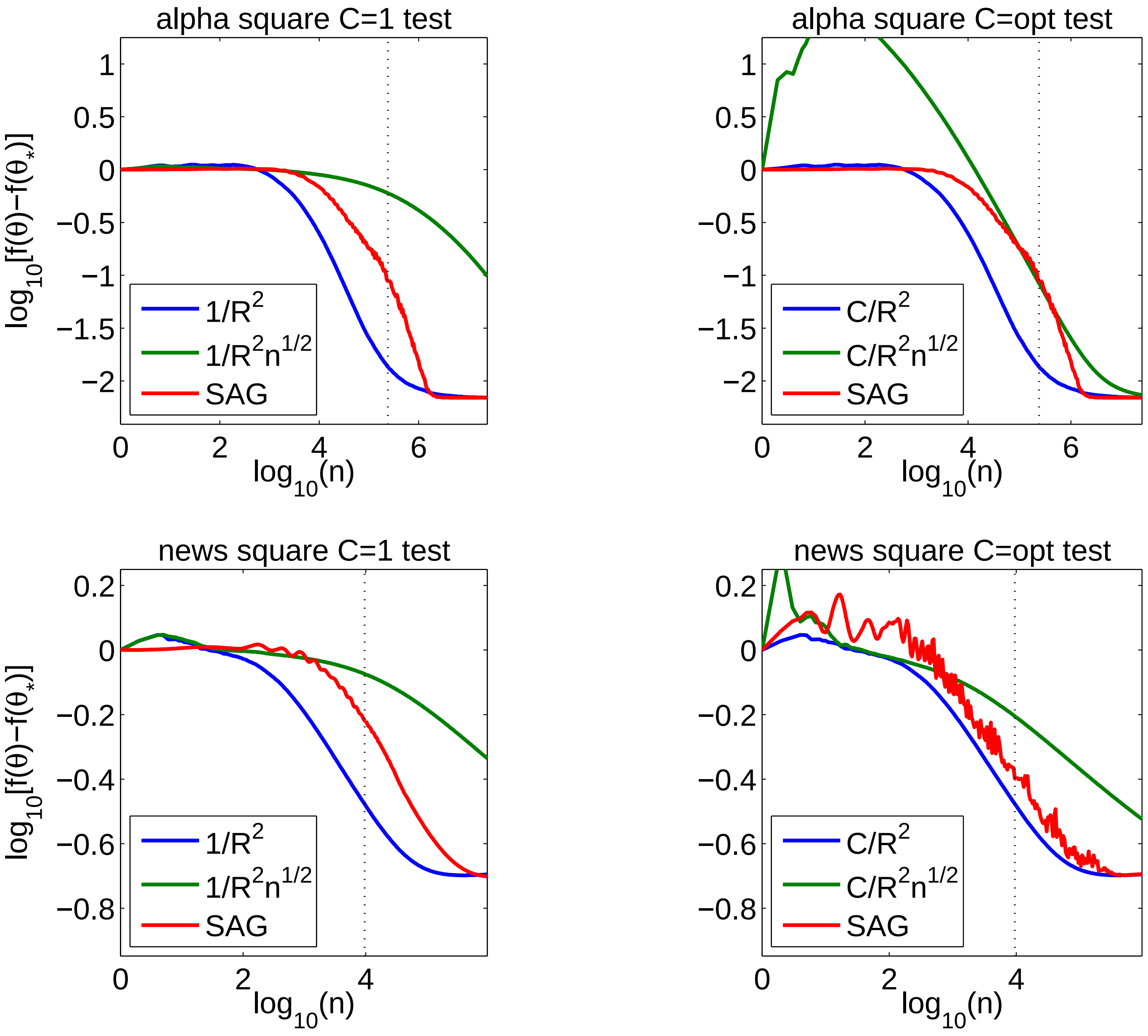
Simulations - synthetic examples

- Gaussian distributions - $p = 20$



Simulations - benchmarks

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



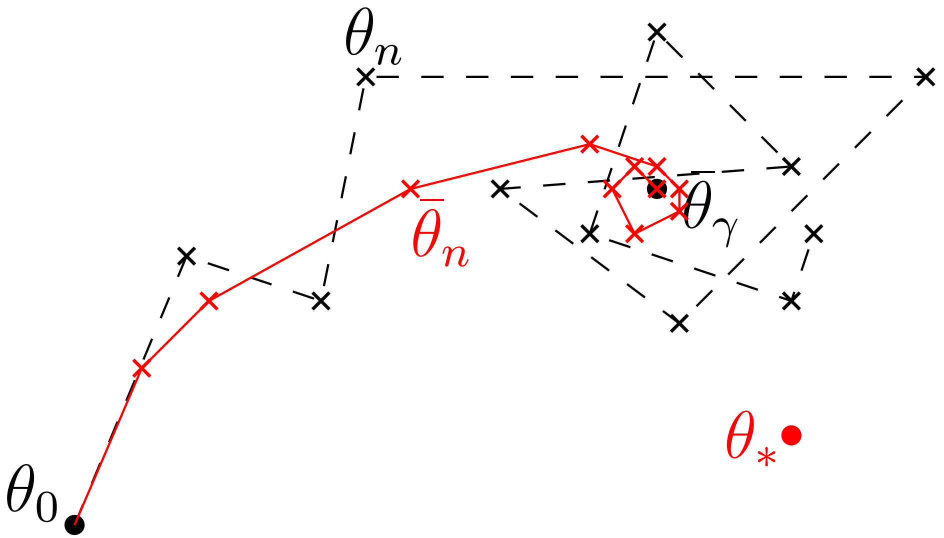
Beyond least-squares - Markov chain interpretation

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

Beyond least-squares - Markov chain interpretation

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

- **θ_n oscillates around the wrong value $\bar{\theta}_\gamma \neq \theta_*$**



Beyond least-squares - Markov chain interpretation

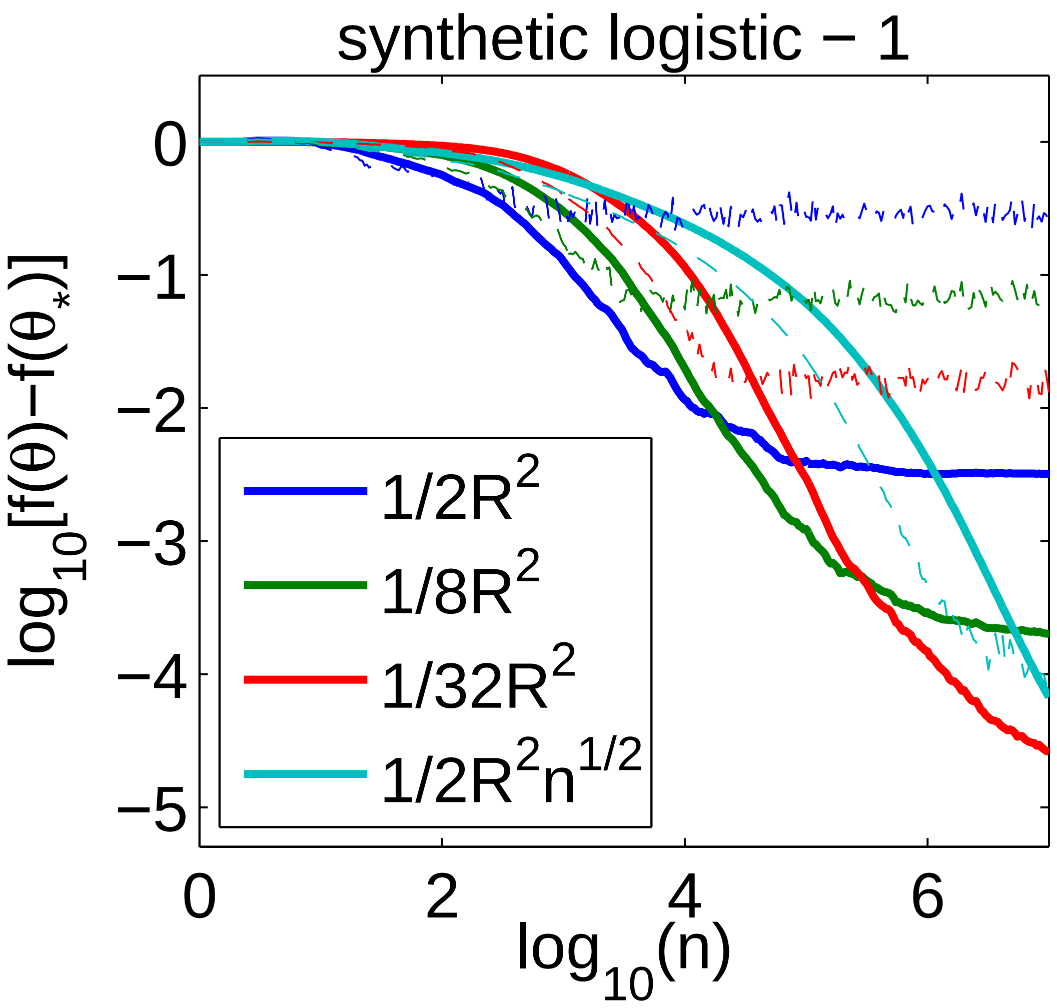
- Recursion $\theta_n = \theta_{n-1} - \gamma f'_n(\theta_{n-1})$ also defines a Markov chain
  - Stationary distribution π_γ such that $\int f'(\theta)\pi_\gamma(\mathrm{d}\theta) = 0$
  - When f' is not linear, $f'(\int \theta\pi_\gamma(\mathrm{d}\theta)) \neq \int f'(\theta)\pi_\gamma(\mathrm{d}\theta)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

- **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 - $p = 20$



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: $O(p)$ 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 $O(p)$**

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

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

Choice of support point for online Newton step

- **Two-stage procedure**

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

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

Logistic regression - Proof technique

- Using generalized self-concordance of $\varphi : u \mapsto \log(1 + e^{-u})$:

$$|\varphi'''(u)| \leqslant \varphi''(u)$$

  – NB: difference with regular self-concordance: $|\varphi'''(u)| \leqslant 2\varphi''(u)^{3/2}$

- Using novel high-probability convergence results for regular averaged stochastic gradient descent

- Requires assumption on the kurtosis in every direction, i.e.,

$$\mathbb{E}\langle \Phi(x_n), \eta \rangle^4 \leqslant \kappa \big[\mathbb{E}\langle \Phi(x_n), \eta \rangle^2 \big]^2$$

Choice of support point for online Newton step

- **Two-stage procedure**

(1) Run $n/2$ iterations of averaged SGD to obtain $\tilde{\theta}$

(2) Run $n/2$ iterations of averaged constant step-size LMS

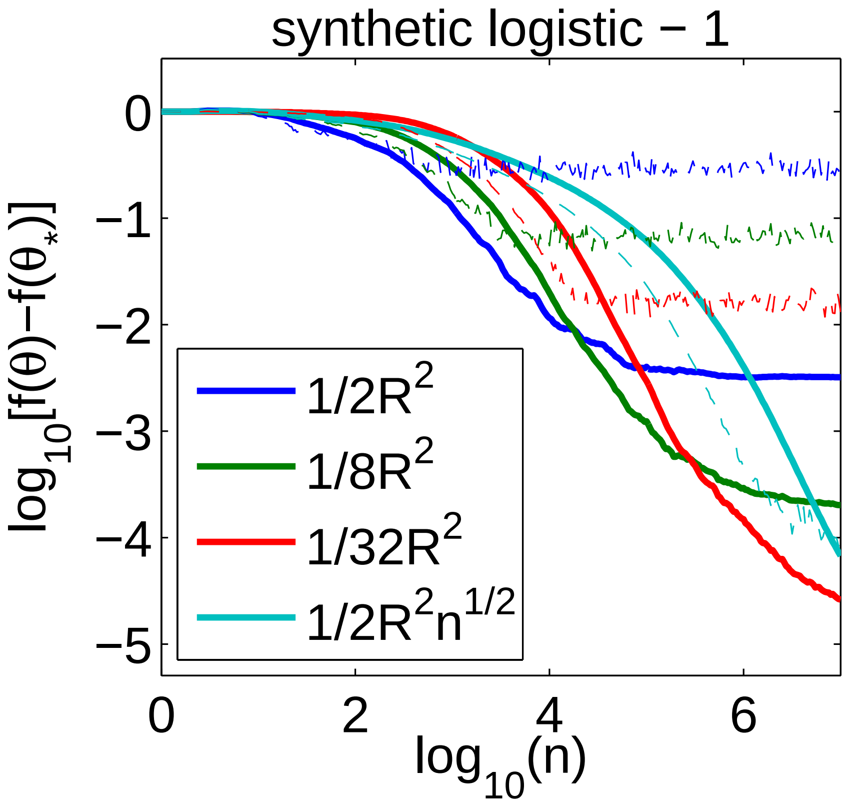
  - Reminiscent of one-step estimators (see, e.g., Van der Vaart, 2000)
  - Provable convergence rate of $O(p/n)$ for logistic regression
  - Additional assumptions but no strong convexity

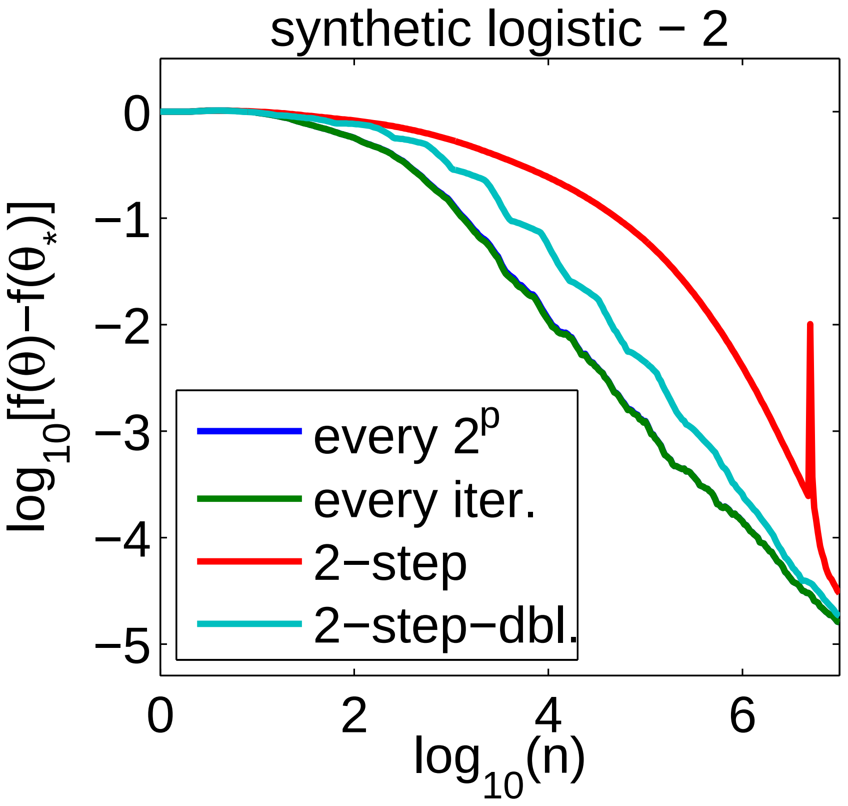
- **Update at each iteration using the current averaged iterate**

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

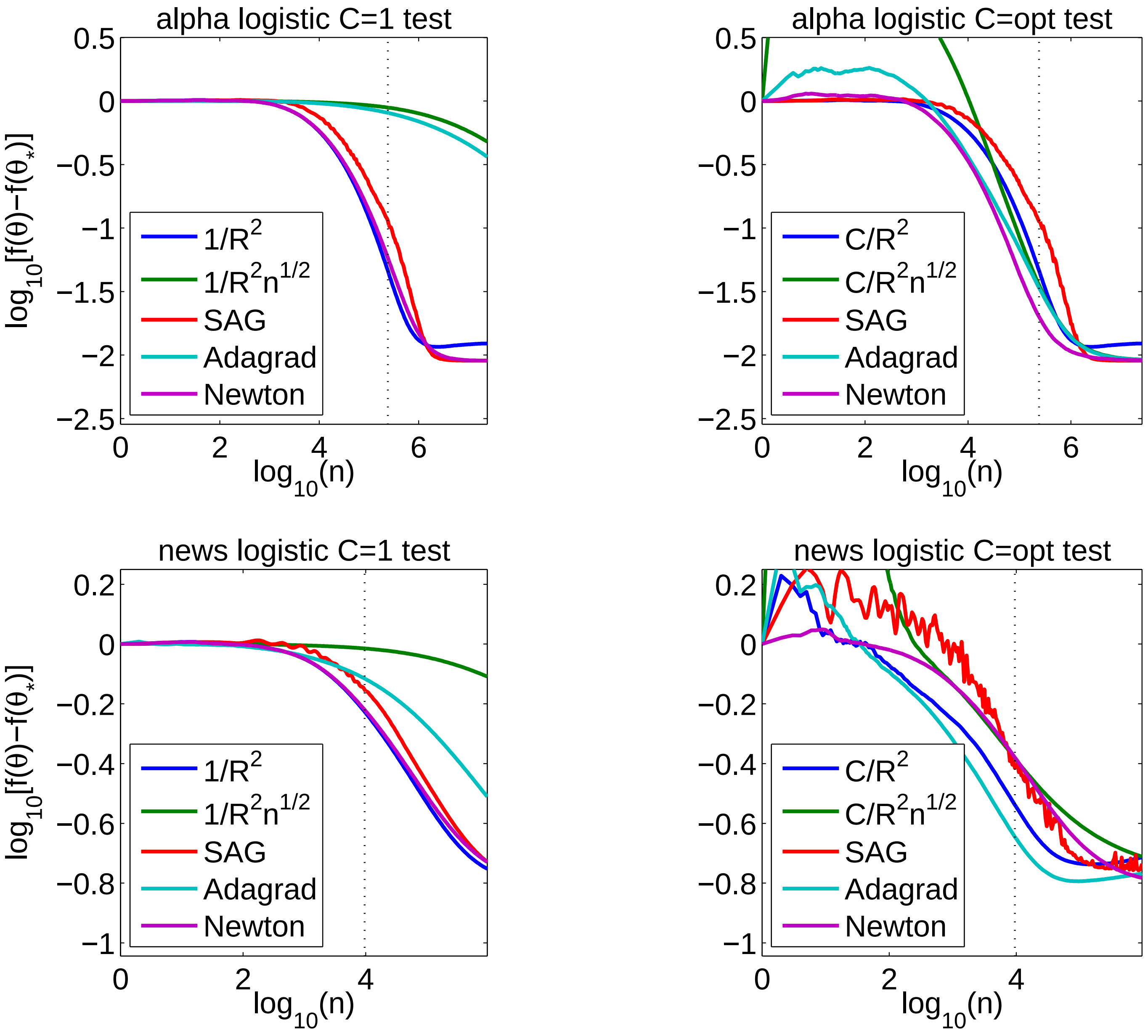
- Gaussian distributions - $p = 20$





Simulations - benchmarks

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



Optimal bounds for least-squares?

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

- **Refined assumptions with adaptivity** (Dieuleveut and Bach, 2014)

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

  - Extension to non-parametric estimation
  - Previous results: $\alpha = 0$ and $r = 1/2$

- **Asymptotic analysis** (Défossez and Bach, 2015)
  - Bias vs. variance: which one is dominating?

- **Acceleration** (Flammarion and Bach, 2015)

Bias-variance decomposition (Défossez and Bach, 2015)

- Simplification: dominating term when $n \to \infty$ and $\gamma \to 0$

- **Variance** (e.g., starting from the solution)

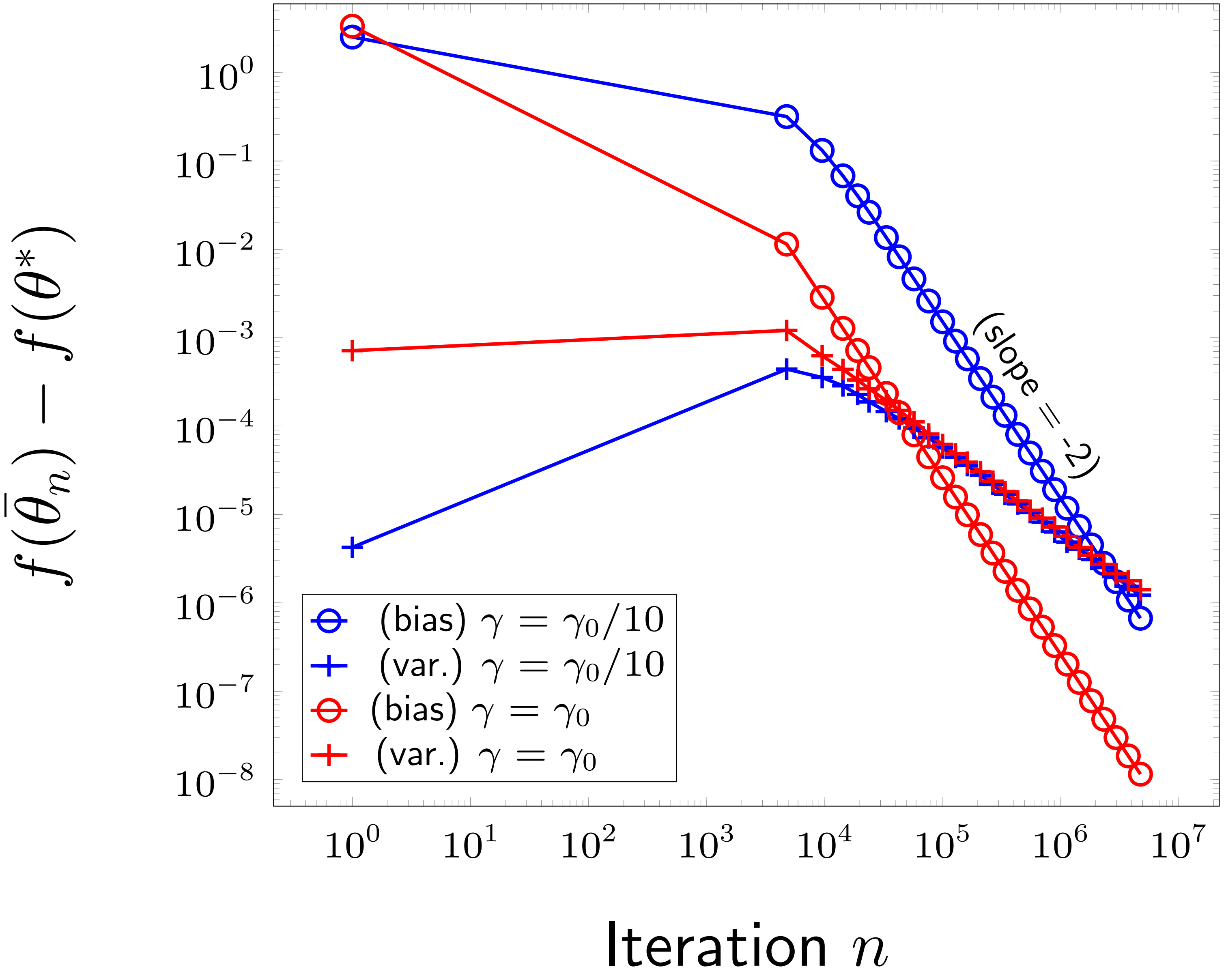
$$f(\bar{\theta}_n) - f(\theta_*) \sim \frac{1}{n} \mathbb{E}\Big[\varepsilon^2 \, \Phi(x)^\top H^{-1} \Phi(x)\Big]$$

  - NB: f 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



Acceleration (Flammarion and Bach, 2015)

- **Existing results** (Bach and Moulines, 2013)
  - **Variance** $= \dfrac{\sigma^2 d}{n}$
  - **Bias** $\leqslant \min \left\{ \dfrac{R^2\|\theta_0-\theta_*\|^2}{n}, \dfrac{R^4\langle\theta_0-\theta_*, H^{-1}(\theta_0-\theta_*)\rangle}{n^2} \right\}$

- **Is it possible to get a bias term in** $\dfrac{R^2\|\theta_0-\theta_*\|^2}{n^2}$**?**
  - Corresponds to acceleration (Nesterov, 1983)
  - Best (current) result:

$$\frac{\sigma^2 d}{n^{1-\alpha}} + \frac{R^2\|\theta_0-\theta_*\|^2}{n^{1+\alpha}}$$

Outline

1. **Large-scale machine learning and optimization**
    - Traditional statistical analysis
    - Classical methods for convex optimization

2. **Non-smooth stochastic approximation**
    - Stochastic (sub)gradient and averaging
    - Non-asymptotic results and lower bounds
    - Strongly convex vs. non-strongly convex

3. **Smooth stochastic approximation algorithms**
    - Asymptotic and non-asymptotic results
    - Beyond decaying step-sizes

4. **Finite data sets**

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

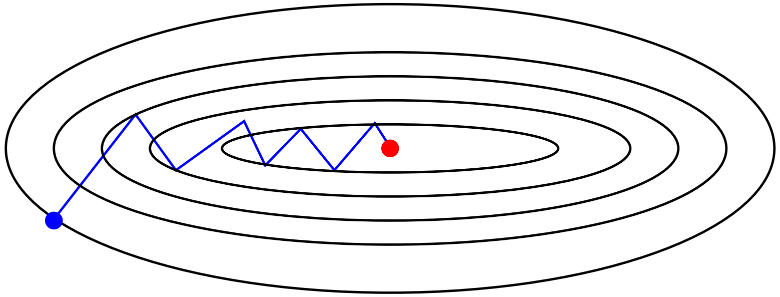
- Minimizing $g(\theta) = \frac{1}{n}\sum_{i=1}^{n} f_i(\theta)$ with $f_i(\theta) = \ell\big(y_i, \theta^\top \Phi(x_i)\big) + \mu\Omega(\theta)$

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

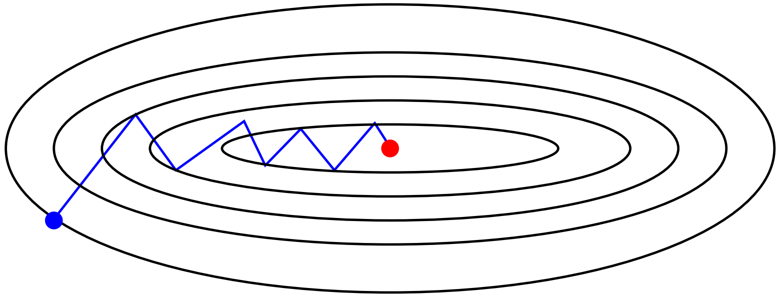
- Batch gradient descent: $\theta_t = \theta_{t-1} - \gamma_t g'(\theta_{t-1}) = \theta_{t-1} - \frac{\gamma_t}{n}\sum_{i=1}^{n} f_i'(\theta_{t-1})$
  - Linear (e.g., exponential) convergence rate in $O(e^{-\alpha t})$
  - Iteration complexity is linear in n *(with line search)*

- Stochastic 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?)*

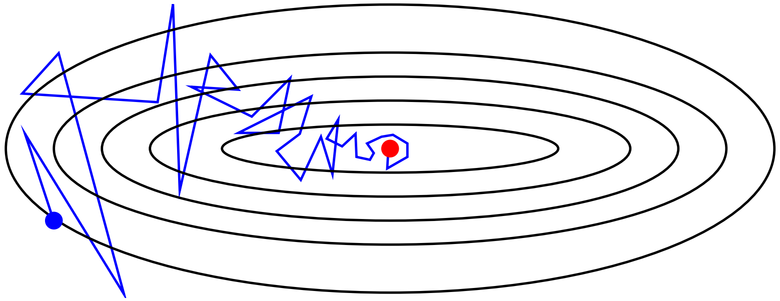
Stochastic vs. deterministic methods

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

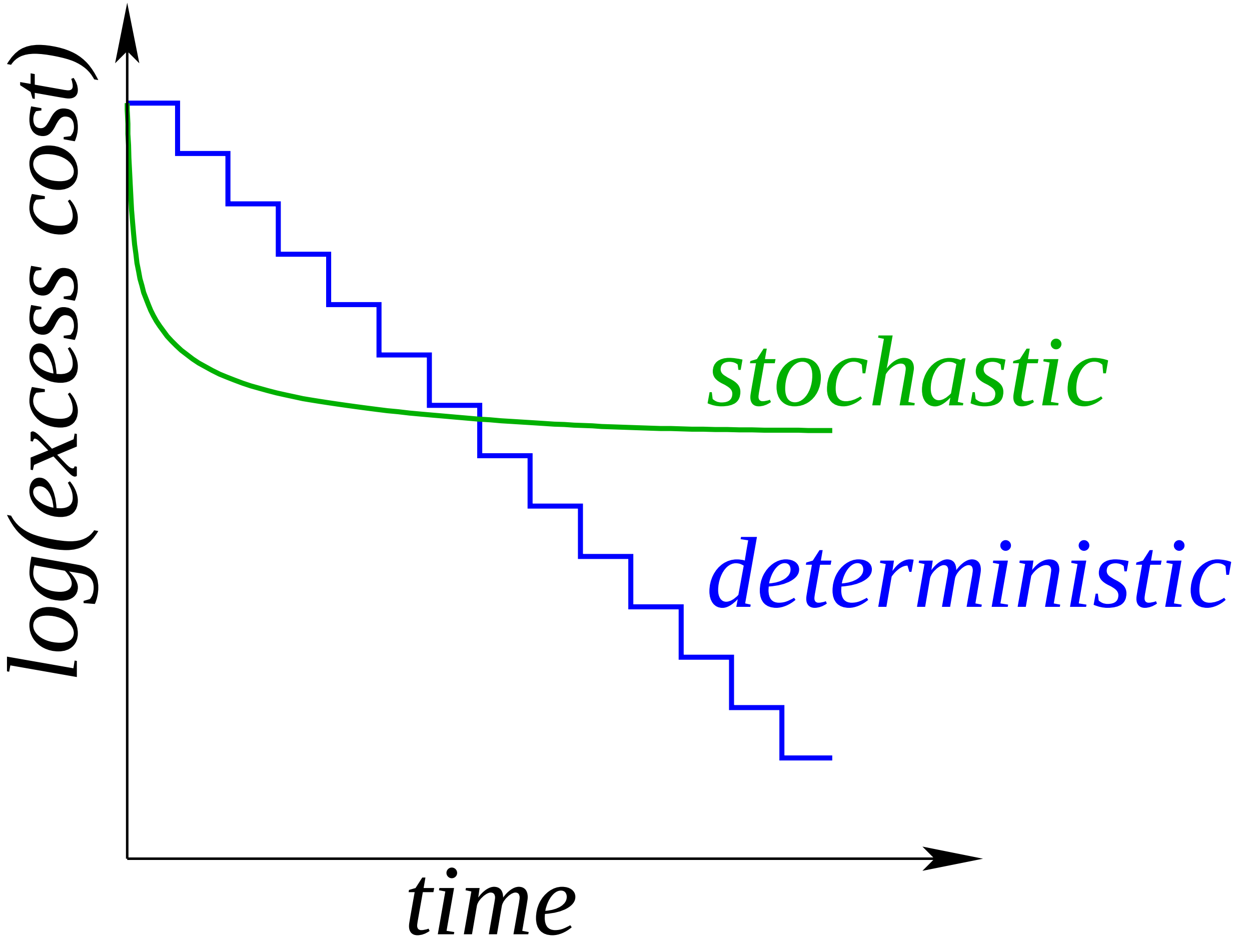


- Stochastic gradient descent: $\theta_t = \theta_{t-1} - \gamma_t f'_{i(t)}(\theta_{t-1})$



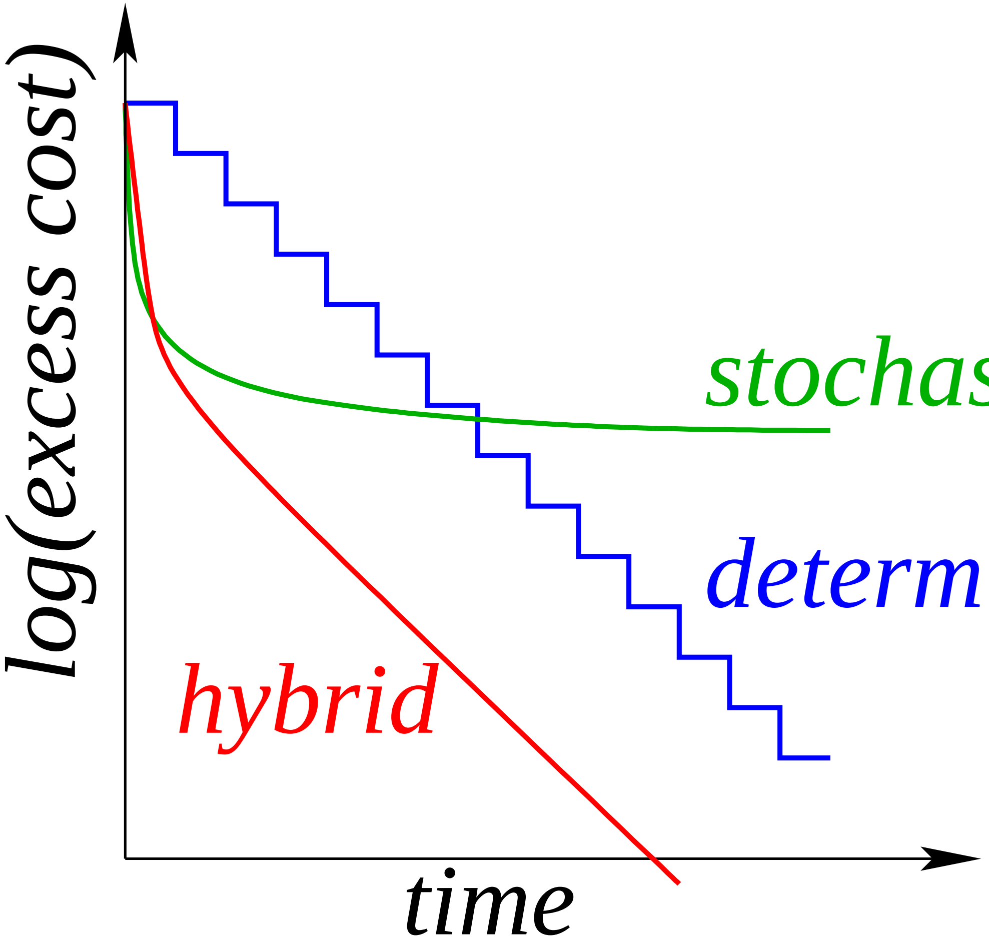
Stochastic vs. deterministic methods

- **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



Accelerating gradient methods - Related work

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

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

Accelerating gradient methods - Related work

- **Momentum, gradient/iterate averaging, stochastic version of accelerated batch gradient methods**
  - Polyak and Juditsky (1992); Tseng (1998); Sunehag et al. (2009); Ghadimi and Lan (2010); Xiao (2010)
  - Can improve constants, but still have sublinear $O(1/t)$ rate
- **Constant step-size stochastic gradient (SG), accelerated SG**
  - Kesten (1958); Delyon and Juditsky (1993); Solodov (1998); Nedic and Bertsekas (2000)
  - Linear convergence, but only up to a fixed tolerance.
- **Stochastic methods in the dual**
  - Shalev-Shwartz and Zhang (2012)
  - Similar linear rate but limited choice for the f_i's

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

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

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

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

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

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

Stochastic average gradient - Convergence analysis

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

Stochastic average gradient - Convergence analysis

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

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

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

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

Stochastic average gradient - Convergence analysis

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

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

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

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

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

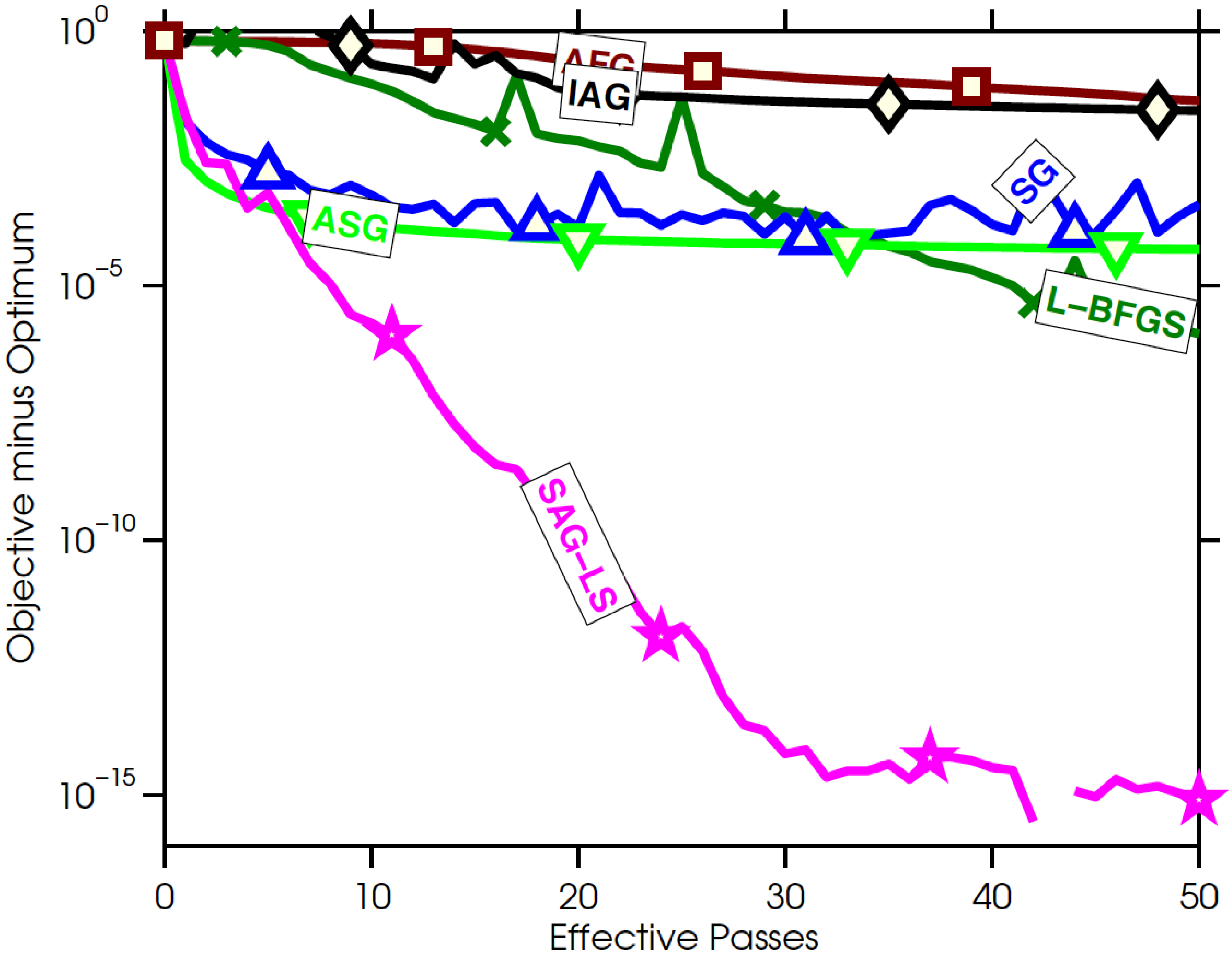
Rate of convergence comparison

- Assume that $L = 100$, $\mu = .01$, and $n = 80000$
  - 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 L 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)



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

- **Lower bounds**
  - Agarwal and Bottou (2014)

Summary and future work

- **Constant-step-size averaged stochastic gradient descent**
  - Reaches convergence rate $O(1/n)$ in all regimes
  - Improves on the $O(1/\sqrt{n})$ lower-bound of non-smooth problems
  - Efficient online Newton step for non-quadratic problems
  - Robustness to step-size selection

- **Going beyond a single pass through the data**

Summary and future work

- **Constant-step-size averaged stochastic gradient descent**
  - Reaches convergence rate $O(1/n)$ in all regimes
  - Improves on the $O(1/\sqrt{n})$ lower-bound of non-smooth problems
  - Efficient online Newton step for non-quadratic problems
  - Robustness to step-size selection

- **Going beyond a single pass through the data**

- **Extensions and future work**
  - Pre-conditioning
  - Proximal extensions fo non-differentiable terms
  - kernels and non-parametric estimation
  - line-search
  - parallelization

Outline

1. **Large-scale machine learning and optimization**
   - Traditional statistical analysis
   - Classical methods for convex optimization

2. **Non-smooth stochastic approximation**
   - Stochastic (sub)gradient and averaging
   - Non-asymptotic results and lower bounds
   - Strongly convex vs. non-strongly convex

3. **Smooth stochastic approximation algorithms**
   - Asymptotic and non-asymptotic results
   - Beyond decaying step-sizes

4. **Finite data sets**

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
  - Going beyond a single pass over the data (testing performance)
  - Characterization of implicit regularization of online methods
  - Distributed processing

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

D. Blatt, A. O. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *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.

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.

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.

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

868–881, 1993.

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

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.

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

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

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.

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

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.

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.

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.

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.

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

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

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

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

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.