Practical Session on Convex Optimization: Exploiting Problem Structure

Mark Schmidt

INRIA/ENS

September 2011

Motivation: Opening up the Black Box

- Last time, we saw for non-smooth problems that using the problem structure could lead to vastly improved performance.

Motivation: Opening up the Black Box

- Last time, we saw for non-smooth problems that using the problem structure could lead to vastly improved performance.
- E.g., proximal-gradient work much better than 'black box' sub-gradient methods.

Motivation: Opening up the Black Box

- Last time, we saw for non-smooth problems that using the problem structure could lead to vastly improved performance.
- E.g., proximal-gradient work much better than 'black box' sub-gradient methods.
- This time, we talk about some more ways to take advantage of problem structure.

Other Ways of Using Problem Structure

- Block Coordinate Descent
- Stochastic Gradient
- Other Techniques

Block Coordinate Descent

- Key idea:
  1. Select some subset of the variables.
  2. Exactly or approximately minimize with respect to subset.

Block Coordinate Descent

- Key idea:
  1. Select some subset of the variables.
  2. Exactly or approximately minimize with respect to subset.
- Very effective when:
  1. Minimization is very cheap.
  2. Problem is close to separable, i.e. $f(x) = \sum_{i=1}^{n} f_i(x_i)$.

Block Coordinate Descent

- Key idea:
  1. Select some subset of the variables.
  2. Exactly or approximately minimize with respect to subset.
- Very effective when:
  1. Minimization is very cheap.
  2. Problem is close to separable, i.e. $f(x) = \sum_{i=1}^{n} f_i(x_i)$.
- Variable-selection strategy:
  1. Cyclic (cheap, works the worst).
  2. Randomized.
  3. Greedy (works the best, often expensive).

Block Coordinate Descent

- Key idea:
  1. Select some subset of the variables.
  2. Exactly or approximately minimize with respect to subset.
- Very effective when:
  1. Minimization is very cheap.
  2. Problem is close to separable, i.e. $f(x) = \sum_{i=1}^{n} f_i(x_i)$.
- Variable-selection strategy:
  1. Cyclic (cheap, works the worst).
  2. Randomized.
  3. Greedy (works the best, often expensive).
- Can show convergence if:
  1. Differentiable and minimizing subset is unique.
  2. Non-differentiable part is separable with respect to subsets.

Coordinate Descent for ℓ_1-Regularized Least Squares

- Implement a coordinate-descent strategy for ℓ_1-regularized least squares.
$$\min_x ||Ax - b||^2 + \lambda ||x||_1.$$
- You can use the sub-differential to exactly solve the sub-problem.
- This is called the 'shooting' algorithm.

Coordinate Descent for ℓ_1-Regularized Least Squares

- Implement a coordinate-descent strategy for ℓ_1-regularized least squares.

$$\min_x ||Ax - b||^2 + \lambda ||x||_1.$$

- You can use the sub-differential to exactly solve the sub-problem.
- This is called the 'shooting' algorithm.
- Extension: block-coordinate descent with direct solver.

Other Ways of Using Problem Structure

- Block Coordinate Descent
- Stochastic Gradient
- Other Techniques

Stochastic Gradient Descent

- For problems where

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x).$$

Stochastic Gradient Descent

- For problems where

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x).$$

- Key idea:
  1. Select some subset of the training examples $\mathcal{B}_k$.
  2. Take a gradient step using the approximation

$$\nabla f(x_k) \approx g(x_k) = \frac{1}{|\mathcal{B}_k|}\sum_{i\in\mathcal{B}_k} f_i(x).$$

Stochastic Gradient Descent

- For problems where

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x).$$

- Key idea:
  1. Select some subset of the training examples $\mathcal{B}_k$.
  2. Take a gradient step using the approximation

$$\nabla f(x_k) \approx g(x_k) = \frac{1}{|\mathcal{B}_k|}\sum_{i \in \mathcal{B}_k} f_i(x).$$

- Converges using a step size of $\alpha_k = O(1/k)$.
- Very effective when:
  1. Number of training examples n is very large.
  2. Gradient approximation is reasonable.

Stochastic Gradient Descent

- For problems where

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x).$$

- Key idea:
  1. Select some subset of the training examples $\mathcal{B}_k$.
  2. Take a gradient step using the approximation

$$\nabla f(x_k) \approx g(x_k) = \frac{1}{|\mathcal{B}_k|}\sum_{i \in \mathcal{B}_k} f_i(x).$$

- Converges using a step size of $\alpha_k = O(1/k)$.
- Very effective when:
  1. Number of training examples n is very large.
  2. Gradient approximation is reasonable.
- Randomized selection has faster (expected) convergence rate.

Second-Order SGD and Polyak-Ruppert Averaging

- We can show that

$$\sqrt{k}(x_k - x_*) \xrightarrow{d} \mathcal{N}(0, \Sigma),$$

where Σ depends on $\{\alpha_k\}$ and the Fisher information matrix.

Second-Order SGD and Polyak-Ruppert Averaging

- We can show that

$$\sqrt{k}(x_k - x_*) \xrightarrow{d} \mathcal{N}(0, \Sigma),$$

  where Σ depends on $\{\alpha_k\}$ and the Fisher information matrix.
- We can also consider Newton-like steps of the form

$$x_{k+1} = x_k - \alpha_k H_k g(x_k).$$

Second-Order SGD and Polyak-Ruppert Averaging

- We can show that

$$\sqrt{k}(x_k - x_*) \xrightarrow{d} \mathcal{N}(0, \Sigma),$$

  where Σ depends on $\{\alpha_k\}$ and the Fisher information matrix.
- We can also consider Newton-like steps of the form

$$x_{k+1} = x_k - \alpha_k H_k g(x_k).$$

- The optimal Σ is given by choosing $\alpha_k = O(1/k)$ and $H_k = \nabla f(x_*)$.

Second-Order SGD and Polyak-Ruppert Averaging

- We can show that
$$\sqrt{k}(x_k - x_*) \xrightarrow{d} \mathcal{N}(0, \Sigma),$$
where Σ depends on $\{\alpha_k\}$ and the Fisher information matrix.
- We can also consider Newton-like steps of the form
$$x_{k+1} = x_k - \alpha_k H_k g(x_k).$$
- The optimal Σ is given by choosing $\alpha_k = O(1/k)$ and $H_k = \nabla f(x_*)$.
- In the 1980s, Polyak and Ruppert showed that the average of the basic stochastic gradient iterations,
$$\bar{x}_{k+1} = \frac{1}{k+1}\sum_{i=1}^{k+1} x_i, \text{with } x_{k+1} = x_k - \alpha_k \nabla f(x_k),$$
achieves the optimal Σ if $\alpha_k = O(1/k^\beta)$, with $\beta \in (1/2, 1)$.

SGD for ℓ_2-Regularized Logistic Regression

- Implement SGD for ℓ_2-regularized least squares,

$$\min_x ||Ax - b||^2 + \frac{\lambda}{2}||x||^2.$$

- Compare using a step size of $\alpha_k = O(1/k)$ to using $\alpha_k = O(1/\sqrt{k})$ with averaging.

SGD for ℓ_2-Regularized Logistic Regression

- Implement SGD for ℓ_2-regularized least squares,

$$\min_x ||Ax - b||^2 + \frac{\lambda}{2}||x||^2.$$

- Compare using a step size of $\alpha_k = O(1/k)$ to using $\alpha_k = O(1/\sqrt{k})$ with averaging.
- Be careful how you handle the regularizer:
  1. You need to re-scale λ in the approximation.
  2. For sparse A, you can track the norm of x instead of updating every element.

Finite-Differencing and Simultaneous Perturbation

- Derivative-free stochastic gradient descent:

$$(1/n)\nabla_j f(x_k) \approx \nabla_j f_i(x_k) \approx \frac{f_i(x_k + \epsilon_k e_j) - f_i(x_k - \epsilon_k e_j)}{2\epsilon_k}.$$

- Requires that $\epsilon_k \to 0$ slower than α_k.

Finite-Differencing and Simultaneous Perturbation

- Derivative-free stochastic gradient descent:

$$(1/n)\nabla_j f(x_k) \approx \nabla_j f_i(x_k) \approx \frac{f_i(x_k + \epsilon_k e_j) - f_i(x_k - \epsilon_k e_j)}{2\epsilon_k}.$$

- Requires that $\epsilon_k \to 0$ slower than α_k.
- Simultaneous perturbation approximation:

$$(1/n)\nabla_j f(x_k) \approx \nabla_j f_i(x_k) \approx \frac{f_i(x_k + \epsilon_k d_k) - f_i(x_k - \epsilon_k d_k)}{2\epsilon_k d_j},$$

where d_i realizes a $\{-1, 1\}$ Bernoulli random variable.

Finite-Differencing and Simultaneous Perturbation

- Derivative-free stochastic gradient descent:

$$(1/n)\nabla_j f(x_k) \approx \nabla_j f_i(x_k) \approx \frac{f_i(x_k + \epsilon_k e_j) - f_i(x_k - \epsilon_k e_j)}{2\epsilon_k}.$$

- Requires that $\epsilon_k \to 0$ slower than α_k.
- Simultaneous perturbation approximation:

$$(1/n)\nabla_j f(x_k) \approx \nabla_j f_i(x_k) \approx \frac{f_i(x_k + \epsilon_k d_k) - f_i(x_k - \epsilon_k d_k)}{2\epsilon_k d_j},$$

  where d_i realizes a $\{-1, 1\}$ Bernoulli random variable.
- These have the same asymptotic convergence rate, but simultaneous perturbation iterations only require two evaluations per iteration.

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	$O(1/k)$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	$O(1/k)$
Nesterov	LCG+SC		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	$O(1/k)$
Nesterov	LCG+SC	$O((1-\sqrt{\mu/L})^k)$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	$O(1/k)$
Nesterov	LCG+SC	$O((1-\sqrt{\mu/L})^k)$	$O(1/k)$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	$O(1/k)$
Nesterov	LCG+SC	$O((1-\sqrt{\mu/L})^k)$	$O(1/k)$
Quasi-Newton	LCG+SC+LCH		

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	$O(1/k)$
Nesterov	LCG+SC	$O((1-\sqrt{\mu/L})^k)$	$O(1/k)$
Quasi-Newton	LCG+SC+LCH	$O(\prod_{i=1}^k \rho_i), \rho_i \to 0$	

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	$O(1/k)$
Nesterov	LCG+SC	$O((1-\sqrt{\mu/L})^k)$	$O(1/k)$
Quasi-Newton	LCG+SC+LCH	$O(\prod_{i=1}^k \rho_i), \rho_i \to 0$	$O(1/k)$

Non-Asymptotic Convergence for Convex Optimization

Algorithm	Assumptions	Deterministic	Stochastic
Sub-Gradient	BSG	$O(1/\sqrt{k})$	$O(1/\sqrt{k})$
Gradient	LCG	$O(1/k)$	$O(1/\sqrt{k})$
Nesterov	LCG	$O(1/k^2)$	$O(1/\sqrt{k})$
Nesterov	Smooth to LCG	$O(1/k)$	$O(1/\sqrt{k})$
Sub-Gradient	SC	$O(\frac{\log k}{k})$	$O(\frac{\log k}{k})$
Average(SGD)	SC	$O(1/k)$	$O(1/k)$
Gradient	LCG+SC	$O((1-\mu/L)^k)$	$O(1/k)$
Nesterov	LCG+SC	$O((1-\sqrt{\mu/L})^k)$	$O(1/k)$
Quasi-Newton	LCG+SC+LCH	$O(\prod_{i=1}^k \rho_i), \rho_i \to 0$	$O(1/k)$

- Deterministic methods only advantageous with continuity.
- Smoothness does not help stochastic methods.
- Stochastic methods achieve the deterministic rate up to some fixed accuracy, and can achieve deterministic rates if noise decreases appropriately.

$O(1/k)$ rate for SGD

- Consider the stochastic gradient method

$$x_{k+1} = x_k - \alpha_k g(x_k),$$

  with $\alpha_k = \frac{1}{\mu k}$.
- Assume that $\mu I \preceq \nabla^2 f(x) \preceq LI$ and that

$$M^2 \geq \sup_x \mathbb{E}[||g(x)||^2],$$

  for some M.
- Show that

$$\mathbb{E}[f(x_k) - f(x_*)] = O(1/k).$$

Other Ways of Using Problem Structure

- Block Coordinate Descent
- Stochastic Gradient
- Other Techniques

References

- For coordinate descent methods, see “Nonlinear Programming” by Dimitri Bertsekas, the papers of Paul Tseng, and the recent report by Yuri Nesterov.
- For stochastic gradient methods, see Dimitri Bertsekas’ “Neurodynamic Programming” book for convergence, ”Introduction to Stochastic Search and Optimization” by James Spall for asymptotic rates, and for non-asymptotic rates see Arkadi Nemirovki’s “Efficient Methods in Convex Programming”.