Algorithmic Challenges of Big Data

Yurii Nesterov, CORE/INMA (UCL)

August 13-15, 2014

Max Planck Institute

Outline

Lecture 1: *Huge-Scale Optimization by Coordinate Updates*

- Problems with sparse data
- Implementation of coordinate moves
- Worst-case efficiency bounds
- Page-rank problem (Google problem)
- Numerical experiments

Lecture 2: *Subgradient methods for Huge-Scale Optimization Problems*

Lecture 3: *Finding primal-dual solutions of Huge-Scale Problems*

Reason for success: intelligent use of *problem structure*

Exercises 1,2: Training on implementation details

Nonlinear Optimization: problems sizes

Class	Operations	Dimension	Iter.Cost	Memory	
Small-size	All	$10^0 - 10^2$	$n^4 \to n^3$	Kilobyte:	10^3
Medium-size	A^{-1}	$10^3 - 10^4$	$n^3 \to n^2$	Megabyte:	10^6
Large-scale	Ax	$10^5 - 10^7$	$n^2 \to n$	Gigabyte:	10^9
Huge-scale	$x + y$	$10^8 - 10^{12}$	$n \to \log n$	Terabyte:	10^{12}

Sources of Huge-Scale problems

- Internet (New)
- Telecommunications (New)
- Finite-element schemes (Old)
- PDE, Weather prediction (Old)

Main hope: Sparsity.

Our plans for today

- Take a very old optimization method.
- Explain why it is very bad.
- Prove that (sometimes) it is very good.
- Check this by numerical experiments.

NB: This will work for two other lectures too.

Very old optimization idea: Coordinate Search

Problem: $\min_{x \in R^n} f(x)$ (f is convex and differentiable).

Coordinate relaxation algorithm

For $k \geq 0$ iterate

1. Choose active coordinate i_k.
2. Update $x_{k+1} = x_k - h_k \nabla_{i_k} f(x_k) e_{i_k}$ ensuring $f(x_{k+1}) \leq f(x_k)$.
   (e_i is ith coordinate vector in R^n.)

Main advantage: Very simple implementation.

Possible strategies

1. Cyclic moves. (Difficult to analyze.)
2. Random choice of coordinate (Why?)
3. Choose coordinate with the maximal directional derivative.

Complexity estimate: assume

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad x, y \in R^n.$$

Let us choose $h_k = \frac{1}{L}$. Then

$$\begin{aligned} f(x_k) - f(x_{k+1}) &\geq \frac{1}{2L}|\nabla_{i_k} f(x_k)|^2 \geq \frac{1}{2nL}\|\nabla f(x_k)\|^2 \\ &\geq \frac{1}{2nLR^2}(f(x_k) - f^*)^2. \end{aligned}$$

Hence, $f(x_k) - f^* \leq \frac{2nLR^2}{k}$, $k \geq 1$. (For pure GM, drop n.)

This was the only known theoretical result known for CDM!

Criticism

Theoretical justification:

- Complexity bounds are not known for the most of the schemes.
- The only justified scheme needs computation of the whole gradient. (Why don't use GM?)

Computational complexity:

- Fast differentiation: if function is defined by a sequence of operations, then $C(\nabla f) \leq 4C(f)$.
- Can we do anything without computing the function's values?

Result: CDM were almost out of computational practice during decades.

Google problem

Let $E \in R^{n \times n}$ be an incidence matrix of a graph. Denote $e = (1, \dots, 1)^T$ and

$$\bar{E} = E \cdot \mathrm{diag}\,(E^T e)^{-1}.$$

Thus, $\bar{E}^T e = e$. Our problem is as follows:

$$\text{Find } x^* \geq 0 : \quad \bar{E} x^* = x^*.$$

Optimization formulation:

$$f(x) \stackrel{\text{def}}{=} \tfrac{1}{2} \| \bar{E} x - x \|^2 + \tfrac{\gamma}{2} [\langle e, x \rangle - 1]^2 \quad \rightarrow \quad \min_{x \in R^n}$$

Huge-scale problems

Main features

- The size is very big ($n \geq 10^7$).
- The data is distributed in space.
- The requested parts of data are not always available.
- The data is changing in time.

Consequences

Simplest operations are expensive or infeasible:

- Update of the full vector of variables.
- Matrix-vector multiplication.
- Computation of the objective function's value, etc.

Structure of the Google Problem

Let us look at the gradient of the objective:

$$\begin{aligned} \nabla_i f(x) &= \langle a_i, g(x) \rangle + \gamma[\langle e, x \rangle - 1], \; i = 1, \dots, n, \\ g(x) &= \bar{E}x - x \in R^n, \quad (\bar{E} = (a_1, \dots, a_n)). \end{aligned}$$

Main observations:

- The coordinate move $x_+ = x - h_i \nabla_i f(x) e_i$ needs $O(p_i)$ a.o. (p_i is the number of nonzero elements in a_i.)
- $d_i \overset{\text{def}}{=} \text{diag}\left(\nabla^2 f \overset{\text{def}}{=} \bar{E}^T \bar{E} + \gamma e e^T \right)_i = \gamma + \frac{1}{p_i}$ are available. We can use them for choosing the step sizes ($h_i = \frac{1}{d_i}$).

Reasonable coordinate choice strategy? Random!

Random coordinate descent methods (RCDM)

$$\min_{x \in R^N} f(x), \qquad (f \text{ is convex and differentiable})$$

Let us decompose the space: $R^N = \bigotimes_{i=1}^{n} R^{n_i}, \;\; N = \sum_{i=1}^{n} n_i,$

$$\begin{aligned} I_N &= (U_1, \dots, U_n) \in R^{N \times N}, \quad U_i \in R^{N \times n_i}, \\ x &= (x^{(1)}, \dots, x^{(n)})^T \; = \sum_{i=1}^{n} U_i x^{(i)}, \quad x^{(i)} \in R^{n_i}. \end{aligned}$$

Partial gradient of $f(x)$ in $x^{(i)}$ is $f'_i(x) = U_i^T \nabla f(x) \in R^{n_i}$.

For R^{n_i}, we fix norms $\|x\|_{(i)}, \;\; \|s\|^*_{(i)} = \max_{\|h\|_{(i)}=1} \langle s, h \rangle$.

If $h(s)$ is the optimal solution, then $s_* \stackrel{\mathrm{def}}{=} \|s\|^*_{(i)} \cdot h(s)$.

Main inequalities

Main Assumption:

$$\|f_i'(x+U_ih_i) - f_i'(x)\|_{(i)}^* \le L_i\|h_i\|_{(i)}, \quad h_i \in R^{n_i},\ i=1,\dots,n.$$

Then

$$f(x+U_ih_i) \le f(x) + \langle f_i'(x), h_i\rangle + \tfrac{L_i}{2}\|h_i\|_{(i)}^2, \quad x \in R^N,\ h_i \in R^{n_i}.$$

Define the coordinate steps: $T_i(x) \overset{\text{def}}{=} x - \frac{1}{L_i}U_i f_i'(x)_*$. Then,

$$f(x) - f(T_i(x)) \ \ \ge\ \ \tfrac{1}{2L_i}\left(\|f_i'(x)\|_{(i)}^*\right)^2, \quad i=1,\dots,n.$$

Proof: Minimize the upper bound.

Random coordinate choice

We need a special random counter $\mathcal{R}_\alpha$, $\alpha \in [0,1]$:

$$\mathbf{Prob}\,[i] \;=\; p_\alpha^{(i)} \;=\; L_i^\alpha \cdot \left[\sum_{j=1}^n L_j^\alpha\right]^{-1}, \quad i = 1,\dots,n.$$

Note: $\mathcal{R}_0$ generates uniform distribution.

Method $RCDM(\alpha, x_0)$

For $k \geq 0$ iterate:

1) Choose $i_k = \mathcal{R}_\alpha$.

2) Update $x_{k+1} = T_{i_k}(x_k)$.

Complexity bounds for RCDM

We need to introduce the following norms for $x, g \in R^N$:

$$\|x\|_\alpha = \left[\sum_{i=1}^n L^\alpha \|x^{(i)}\|^2_{(i)}\right]^{1/2}, \quad \|g\|^*_\alpha = \left[\sum_{i=1}^n \frac{1}{L^\alpha}\left(\|g^{(i)}\|^*_{(i)}\right)^2\right]^{1/2}.$$

After k iterations, $RCDM(\alpha, x_0)$ generates random output x_k, which depends on $\xi_k = \{i_0, \dots, i_k\}$. Denote $\phi_k = E_{\xi_{k-1}} f(x_k)$.

Theorem. For any $k \geq 1$ we have

$$\phi_k - f^* \leq \frac{2}{k} \cdot \left[\sum_{j=1}^n L_j^\alpha\right] \cdot R^2_{1-\alpha}(x_0),$$

where $R_\beta(x_0) = \max_x \left\{ \max_{x_* \in X^*} \|x - x_*\|_\beta : \ f(x) \leq f(x_0) \right\}$.

Interpretation I

1. $\alpha = 0$. Then $S_0 = n$, and we get

$$\phi_k - f^* \quad \leq \quad \tfrac{2n}{k} \cdot R_1^2(x_0).$$

Note

- We use the metric $\|x\|_1^2 = \sum\limits_{i=1}^{n} L_i \|x^{(i)}\|_{(i)}^2$.
- For matrix with diagonal $\{L_i\}_{i=1}^n$ its norm can reach n.
- Hence, for GM we can guarantee the same bound.

  But its cost of iteration is much higher!

Interpretation II

2. $\alpha = \frac{1}{2}$. Let $n_i = 1$, $i = 1, \dots, n$. Denote

$$D_\infty(x_0) = \max_x \left\{ \max_{y \in X^*} \max_{1 \le i \le n} |x^{(i)} - y^{(i)}| : \; f(x) \le f(x_0) \right\}.$$

Then, $R^2_{1/2}(x_0) \le S_{1/2} D^2_\infty(x_0)$, and we obtain

$$\phi_k - f^* \quad \le \quad \frac{2}{k} \cdot \left[\sum_{i=1}^n L_i^{1/2} \right]^2 \cdot D^2_\infty(x_0).$$

Note:

- For the first order methods, the worst-case complexity of minimizing over a box depends on n.
- Since $S_{1/2}$ can be bounded, RCDM can be applied in situations where the usual GM fail.

Interpretation III

3. $\alpha = 1$. Let all norms $\| \cdot \|_{(i)}$ are standard Euclidean. Then $R_0(x_0)$ is the size of the initial level set, and

$$\phi_k - f^* \;\leq\; \frac{2}{k} \cdot \left[\sum_{i=1}^{n} L_i \right] \cdot R_0^2(x_0) \;\equiv\; \frac{2n}{k} \cdot \left[\frac{1}{n} \sum_{i=1}^{n} L_i \right] \cdot R_0^2(x_0).$$

Rate of convergence of GM can be estimated as

$$f(x_k) - f^* \leq \frac{\gamma}{k} R_0^2(x_0),$$

where γ satisfies condition $f''(x) \preceq \gamma \cdot I$, $x \in R^N$.
Note: maximal eigenvalue of symmetric matrix can reach its trace.

In the worst case, the rate of convergence of GM is the same as that of $RCDM$.

Minimizing strongly convex functions

Theorem. Let $f(x)$ be strongly convex with respect to $\| \cdot \|_{1-\alpha}$ with convexity parameter $\sigma_{1-\alpha} > 0$.
Then, for $\{x_k\}$ generated by $RCDM(\alpha, x_0)$ we have

$$\phi_k - \phi^* \quad \leq \quad \left(1 - \tfrac{\sigma_{1-\alpha}}{S_\alpha}\right)^k (f(x_0) - f^*).$$

Proof: Let x_k be generated by $RCDM$ after k iterations. Let us estimate the expected result of the next iteration.

$$f(x_k) - E_{i_k}(f(x_{k+1})) = \sum_{i=1}^{n} p_\alpha^{(i)} \cdot [f(x_k) - f(T_i(x_k))]$$
$$\geq \sum_{i=1}^{n} \tfrac{p_\alpha^{(i)}}{2L_i} \left(\| f_i'(x_k) \|_{(i)}^* \right)^2 \; = \; \tfrac{1}{2S_\alpha} (\| f'(x_k) \|_{1-\alpha}^*)^2$$
$$\geq \tfrac{\sigma_{1-\alpha}}{S_\alpha} (f(x_k) - f^*).$$

It remains to compute expectation in ξ_{k-1}. □

Confidence level of the answers

Note: We have proved that the expected values of random $f(x_k)$ are good.

Can we guarantee anything after a single run?

Confidence level: Probability $\beta \in (0,1)$, that some statement about random output is correct.

Main tool: Markov inequality $(\xi, T > 0)$:

$$\mathbf{Prob}\,[\xi \geq T] \;\leq\; \frac{E(\xi)}{T}.$$

Our situation:

$$\mathbf{Prob}\,[f(x_k) - f^* \geq \epsilon] \;\leq \tfrac{1}{\epsilon}[\phi_k - f^*] \;\leq\; 1 - \beta.$$

We need $\phi_k - f^* \leq \epsilon \cdot (1 - \beta)$. Too expensive for $\beta \to 1$?

Regularization technique

Consider $f_\mu(x) = f(x) + \frac{\mu}{2}\|x - x_0\|^2_{1-\alpha}$. It is strongly convex.

Therefore, we can obtain $\phi_k - f^*_\mu \leq \epsilon \cdot (1-\beta)$ in

$$O\left(\frac{1}{\mu} S_\alpha \ln \frac{1}{\epsilon\cdot(1-\beta)}\right) \text{ iterations.}$$

Theorem. Define $\alpha = 1$, $\mu = \frac{\epsilon}{4R_0^2(x_0)}$, and choose

$$k \;\; \geq \;\; 1 + \frac{8S_1 R_0^2(x_0)}{\epsilon}\left[\ln \frac{2S_1 R_0^2(x_0)}{\epsilon} + \ln \frac{1}{1-\beta}\right].$$

Let x_k be generated by $RCDM(1, x_0)$ as applied to f_μ. Then

$$\textbf{Prob}\left(f(x_k) - f^* \leq \epsilon\right) \geq \beta.$$

Note: $\beta = 1 - 10^{-p} \quad \Rightarrow \quad \ln 10^p = 2.3p.$

Extensions

1. Problems with constraints:

$$\min_{x \in Q} \quad f(x),$$

where $Q = \bigotimes_{i=1}^{n} Q_i$, $Q_i \subseteq R^{n_i}$, $i = 1, \dots, n$, are closed and convex.
Define the constrained coordinate update:

$$\begin{array}{rcl} u^{(i)}(x) & = & \arg \min\limits_{u^{(i)} \in Q_i} \left[\langle f'_i(x), u^{(i)} - x^{(i)} \rangle + \frac{L_i}{2} \| u^{(i)} - x^{(i)} \|^2_{(i)} \right], \\ T_i(x) & = & x + U_i^T (u^{(i)} - x^{(i)}), \quad i = 1, \dots, n. \end{array}$$

Then

$$f(x) - f(T_i(x)) \geq \frac{L_i}{2} \| u^{(i)} - x^{(i)} \|^2_{(i)}, \quad i = 1, \dots, n.$$

Uniform coordinate decent method with constraints

For $k \geq 0$ iterate:

1) Choose randomly i_k by uniform distribution on $\{1 \dots n\}$.

2) Update $x_{k+1} = T_{i_k}(x_k)$.

Theorem. For any $k \geq 0$ we have

$$\phi_k - f^* \quad \leq \quad \tfrac{n}{n+k} \cdot \left[\tfrac{1}{2} R_1^2(x_0) + f(x_0) - f^* \right].$$

If f is strongly convex in $\| \cdot \|_1$ with constant σ, then

$$\phi_k - f^* \quad \leq \quad \left(1 - \tfrac{2\sigma}{n(1+\sigma)} \right)^k \cdot \left(\tfrac{1}{2} R_1^2(x_0) + f(x_0) - f^* \right).$$

Implementation details: Random Counter

Given the values L_i, $i = 1, \dots, n$, generate efficiently random $i \in \{1, \dots, n\}$ with probabilities $\mathbf{Prob}\,[i = k] = L_k / \sum\limits_{j=1}^{n} L_j$.

Solution: **a)** Trivial $\Rightarrow$ $O(n)$ operations.

b) Assume $n = 2^p$. Define $p+1$ vectors $S_k \in R^{2^{p-k}}$, $k = 0, \dots, p$:

$$S_0^{(i)} = L_i, \; i = 1, \dots, n.$$

$$S_k^{(i)} = S_{k-1}^{(2i)} + S_{k-1}^{(2i-1)}, \; i = 1, \dots, 2^{p-k}, \quad k = 1, \dots, p.$$

Algorithm: Make the choice in p steps, from top to bottom.

- If the element i of S_k is chosen, then choose in S_{k-1} either $2i$ or $2i-1$ in accordance to probabilities $\frac{S_{k-1}^{(2i)}}{S_k^{(i)}}$ or $\frac{S_{k-1}^{(2i-1)}}{S_k^{(i)}}$.

Difference: for $n = 2^{20} > 10^6$ we have $p = \log_2 n = 20$.

Numerical experiments: Google problem

$$f(x) \stackrel{\rm def}{=} \frac{1}{2}\|\bar{E}x - x\|^2 + \frac{\gamma}{2}[\langle e, x\rangle - 1]^2 \to \min_{x \in R^n},$$

where $\gamma > 0$ is a penalty parameter, the norm is Euclidean.

Termination criterion: $\|\bar{E}x - x\|_{(2)} \leq \epsilon \cdot \|x\|_{(2)}$ with $\epsilon = 0.01$.

Computer: Notebook Pentium-4 1.6GHz.

n	p	γ	k	Time (sec)
65536	10	$\frac{1}{n}$	47	7.41
	10	$\frac{1}{\sqrt{n}}$	65	10.5
262144	10	$\frac{1}{n}$	47	42.7
	10	$\frac{1}{\sqrt{n}}$	72	76.5
1048576	10	$\frac{1}{n}$	49	247
	10	$\frac{1}{\sqrt{n}}$	82	486

NB: Moderate growth of computational time.

Conclusion

1. We presented a technique for solving huge-scale *smooth* optimization problems with simple constraints.

2. Data can be distributed in space.

3. Data can be changing in time.

Next lecture: Huge-scale *nonsmooth* optimization problems.