STA 243 Final Project: Coordinate Descent Algorithms

Ninghui Li[*] Chenghan Sun[†] Han Chen[‡]

June 12, 2020

1 Introduction

Coordinate Descent(CD) algorithms are iterative methods in which each iterate is obtained by fixing most components of the variable vector x at their values from the current iteration, and approximately minimizing the objective with respect to the remaining components. This method is related closely to the Gauss-Seidel and SOR methods for equation solving Ortega and Rheinboldt 2000. The main advantage of these methods is the simplicity of each iteration, both in generating the search direction and in performing the update of variables. They have been used in applications for many years, and their popularity continues to grow because of their usefulness in data analysis, machine learning, and other areas of current interest. Given its importance and practicability, this article describes the recent methodology for Coordinate Descent algorithms

Convergence of the Coordinate Descent method requires typically that f be strictly convex (or quasiconvex or hemivariate) differentiable and, taking into account the bound constraints, has bounded level sets. Zadeh 1970 relaxed the strict convexity assumption to pseudoconvexity, which allows f to have a non-unique minimum along coordinate directions. If f is not (pseudo)convex, then an example of Powell 1973 shows that the method may cycle without approaching any stationary point of f. Nonetheless, convergence can be shown for special cases of non(pseudo)convex f. If f is not differentiable, the coordinate descent method may get stuck at a non-stationary point even when f is convex see Auslender 1976. For this reason, it is perceived generally that the method is unsuitable when f is nondifferentiable. However, an exception occurs when the nondifferentiable part of f is seperable. Such a structure for f was considered first by Auslender 1976 in the case where f is strongly convex. This structure arises also in least-square problems where an l_1-penalty is placed on a subset of the parameters in order to minimize the support. In this case, here we only consider to problem setup, the unconstrained convex f and convex objective function with separable regularization term

This paper discusses the iteration complexity of Randomized Coordinate Descent(RCD), Accelerated Randomized CD(ARCD) and Cyclic CD(CCD) under convexity and strong convexity respectively. It also extends CD to the Separable Regularized Coordinate Descent(SpCD) case. In section 2, the performance of different CD methods are summarized. The modifications and contributions of each algorithms are emphasized as well. In section 3, we provide the theoretical proofs corresponding to the iteration complexity of each algorithms. In the last section, the implementations are presented where we focus on the most popular quadratic form least square objective

[*] Department of Economics, UC Davis and nhli@ucdavis.edu

[†] Department of Chemical Engineering, UC Davis and chesun@ucdavis.edu

[‡] Department of Statistics, UC Davis and nahchen@ucdavis.edu

function. We use this part to support the theoretical results in the algorithm.

1.1 Related work

In literature, there are more discussion on Block Coordinate Descent Methods, which adjust groups of blocks of indices at each iteration, thus searching along a coordinate hyperplane rather than a single coordinate direction. Most derivation and analysis of single coordinate descent methods can be extended to the block-CD setting.

There are some discussion on the order of coordinates. Basically, three types of strategy are used in the end. Random choice, Greedy Strategy and Cyclic choice. Some other aspects, data dimension and regularization parameter, can be taken into account. As CD updates one or a block of coordinates in each iteration, the work per iteration and the associated data dimension play a role in the iteration complexity. When regularization is included, the regularization parameter changes the performance of algorithms. For instance, a number of recent papers suggest RCD performs better than CCD, while Gurbuzbalaban et al. 2017 shows that CCD is faster than RCD when the regularization parameter is large.

2 Methodology

Three versions of coordinate descent algorithms are discussed: randomized coordinate descent (CD), accelerated randomized CD, cyclic CD. Extensions to separable regularized CD are also included. We summarise the result in table 1

Algorithm	Assumption	Iteration Complexity	Work per iteration
Coordinate Descent(CD)	σ-SC and $L_{\max}$-CLip	$\mathcal{O}\left(\frac{nL_{\max}}{\sigma}\log\left(\frac{1}{\varepsilon}\right)\right)$	cheap
Coordinate Descent(CD)	C and $L_{\max}$-CLip	$\mathcal{O}\left(\frac{nL_{\max}}{\varepsilon}\right)$	cheap
Accelerated Randomized CD	σ-SC and $L_{\max}$-CLip	$\mathcal{O}\left(n\sqrt{\frac{L_{\max}}{\sigma}}\log\left(\frac{\sigma}{\varepsilon L_{\max}}\right)\right)$	expensive
Accelerated Randomized CD	C and $L_{\max}$-CLip	$\mathcal{O}\left(\frac{n\sqrt{L_{\max}}}{\sqrt{\epsilon}}\right)$	expensive
Cyclic version CD	σ-SC and $L_{\max}$-CLip	$\mathcal{O}\left(\frac{n^2L^2}{\sigma L_{\max}}\log(\frac{1}{\varepsilon})\right)$	cheap
Cyclic version CD	C and $L_{\max}$-CLip	$\mathcal{O}\left(\frac{n^2L^2}{\varepsilon L_{\max}}\right)$	cheap
Separable Regularized CD	σ-SC and $L_{\max}$-CLip	$\mathcal{O}\left(\frac{nL_{\max}}{\sigma}\log\left(\frac{1}{\epsilon}\right)\right)$	expensive
Separable Regularized CD	C and $L_{\max}$-CLip	$\mathcal{O}(\frac{nL_{\max}}{\varepsilon})\log(\frac{1}{\varepsilon})$	expensive

Table 1: Iteration complexity of Coordinate Descent Algorithm. SC stands for strongly-convex. C stands for just convex. $L_{\max}$-CLip stands for $L_{\max}$ component Lipschitz continuously differentiable. Iteration complexity refers to make how many steps do we need to run the algorithm to $E[f(x^k)]$ get ε close to the global minimum.

First, we present RCD algorithm in Algorithm 1 as the benchmark. The ARCD chooses index i_k in the same way as the RCD. The difference is that strong convexity $\sigma > 0$ and the component-wise Lipschitz constants L_i are assumed to be available. More details can be found in Algorithm 3 in the appendix. The CCD algorithm differs from RCD algorithm that i_k is chosen according to the cyclic ordering $i_{k+1} = [i_k \mod n] + 1$. We then extends RCD to the SpCD. The iteration step are modified with the additional regularization term shown in Algorithm 2

Algorithm 1 Randomized Coordinate Descent for (1)

```
repeat
    Choose index $i_k \in \{1, 2, ..., n\}$ with uniform probability, independently of choices at prior iterations;
    $x^{k+1} \leftarrow x^k - \alpha_k [\bigtriangledown f(x^k)]_{i_k} e_{ik}$ for some $\alpha_k \geq 0$
    $k \leftarrow k + 1$;
until termination test satisfied
```

Algorithm 2 Separable Coordinate Descent for (2)

```
Set $k \leftarrow 0$ and choose $x^0 \in R^n$;
repeat
    Choose index $i_k \in \{1, 2, ..., n\}$ with uniform probability, independently of choices at prior iterations;
    $z^k_{i_k} \leftarrow \arg\min_\chi \left(\chi - x^k_{i_k}\right)^T \left[\nabla f\left(x^k\right)\right]_{i_k} + \frac{1}{2\alpha_k} \left\|\chi - x^k_{i_k}\right\|_2^2 + \lambda\Omega_i(\chi)$ for some $\alpha_k > 0$;
    $x^{k+1} \leftarrow x^k + (z^k_{i_k} - x^k_{i_k}) e_{i_k}$;
    $k \leftarrow k + 1$;
until termination test satisfied
```

3 Theoretical Results

3.1 Basic Assumption and Notation

- Coordinate Lipschitz differentiable

$$|[\nabla f(x + te_i)]_i - [\nabla f(x)]_i| \leq L_i |t|$$

The corresponding coordinate Lipschitz constant $L_{\max}$ is defined as $L_{\max} = \max_{i=1,...,n} L_i$

- Standard Lipschitz differentiable

$$\|\nabla f(x + d) - \nabla f(x)\| \leq L\|d\|$$

Here we have the relationship that $1 \leq L/L_{\max} \leq n$ In order to gain more insight into this ratio, lower values of this ratio are attained on functions that are "more decoupled" and larger values attained when there is a greater dependence between the coordinates.

3.2 Problem Setup

- Unconstrained minimzation problem:

$$\min_x f(x) \tag{1}$$

where $f : R^n \to R$ is continuous. Different variants of CD make further assumptions, which will be discussed in detail later. It is often assumed to be smooth and convex, sometimes smooth and possibly non-convex, and sometimes smooth but with a restricted domain.

- Regularized function

$$\min_x h(x) := f(x) + \lambda\Omega(x) \tag{2}$$

Usually, we assume f is smooth, Ω is assumed to be convex and separable. i.e. $\Omega(x) = \sum_{i=1}^n \Omega_i(x_i)$. $\lambda > 0$ is a regularization parameter.

3.3 Unconstrained Coordinate Descent

Theorem 1. *Suppose that Assumption 1 holds. Suppose that $\alpha_k \equiv 1/L_{\max}$ in Algorithm 1. Then for all $k > 0$ we have*

$$E\left(f\left(x^k\right)\right) - f^* \leq \frac{2nL_{\max}R_0^2}{k} \tag{3}$$

When $\sigma > 0$ we have in addition that

$$E\left(f\left(x^k\right)\right) - f^* \leq \left(1 - \frac{\sigma}{nL_{\max}}\right)^k \left(f\left(x^0\right) - f^*\right) \tag{4}$$

Compare it with full gradient descent with constant step length $\alpha_k = 1/L$, the iteration is

$$f\left(x^k\right) - f^* \leq \frac{2LR_0^2}{k}$$

In extreme case, the convergence rate for CD is equivalent to traditional gradient descent. i.e. when $L = nL_{\max}$. We pay a price in terms of slower convergence rate for using only one component of $\nabla f(x^k)$ at each iteration.

3.4 Accelerated Randomized Coordinate Descent

The Algorithm of ARCD is present in Algorithm 3 in the appendix. The convergence analysis is shown in Theorem 2

Theorem 2. *Suppose that Assumption 1 holds, and define*

$$S_0 = \sup_{x^* \in \mathcal{S}} L_{max} \|x^0 - x^*\|^2 + (f(x^0) - f^*)/n^2$$

Then for all $k \geq 0$, we have

$$\begin{aligned} E[f(x^k)] - f^* &\leq S_0 \frac{\sigma}{L_{max}} \left[\left(1 + \frac{\sqrt{\sigma/L_{max}}}{2n}\right)^{k+1} - \left(1 - \frac{\sqrt{\sigma/L_{max}}}{2n}\right)^{k+1}\right]^{-2} \\ &\leq S_0 (\frac{n}{k+1})^2 \end{aligned} \tag{5}$$

Sketch of Proof:
(1) Construct $r_{k+1}^2 = \|v_{k+1} - x^*\|^2$. Derive the inequality

$$b_{k+1}^2 E_{i_k}(r_{k+1}^2) \leq b_k^2 r_k^2 - 2a_{k+1}^2(E_{i_k}(f(x^{k+1})) - f^*) + 2a_k^2 \left(f(x^k) - f^*\right)$$

(2) Denote $\phi_{k+1} = E[f(x^{k+1})]$ and get

$$2a_{k+1}^2(\phi_{k+1} - f^*) + b_{k+1}^2 E(r_{k+1}^2) \leq 2a_0^2(f(x^0) - f^*) + b_0^2 \left\|x^0 - x^*\right\|^2$$

Note that $a_0 = 1/n$ and $b_0 = \sqrt{2L_{max}}$.
(3) Estimate the growth rate of a_k, b_k. Find $a_k \geq \frac{1}{\sqrt{\sigma/L_{max}}}[Q_1^{k+1} - Q_2^{k+1}]$ and $b_k \geq [Q_1^{k+1} + Q_2^{k+1}]$, where $Q_1 = 1 + \frac{\sqrt{\sigma/L_{max}}}{2n}$ and $Q_2 = 1 - \frac{\sqrt{\sigma/L_{max}}}{2n}$. Plug the growth rate into the inequality in (2). The proof is complete.

The term $\left(1 + \frac{\sqrt{\sigma/L_{max}}}{2n}\right)^{k+1}$ eventually dominates the second term in brackets in equation 5. The linear convergence rate is significantly faster than the corresponding rate in Algorithm 1. Essentially, the measure σ/L_{max} of conditioning in equation 4 is replaced by its square root in equation 5, suggesting a decrease by a factor of $\sqrt{L_{max}/\sigma}$ in the number of iterations required to meet a specified error tolerance.

3.5 Cyclic Version of Coordinate Descent

Theorem 3. *Suppose that Assumption 1 holds. Suppose that $\alpha_k = 1/L_{max}$, with the iteration i_k, chosen according to the cyclic ordering (with $i_0 = 1$). Then for $k = n, 2n, 3n, ...,$ we have*

$$f(x^k) - f^* \leq \frac{4nL_{max}(1 + nL^2/L_{max}^2)R_0^2}{k+8} \tag{6}$$

When $\sigma > 0$ in the strong convexity condition, we have in addition for $k = n, 2n, 3n, ...$

$$f(x^k) - f^* \leq \Big(1 - \frac{\sigma}{2L_{max}(1 + nL^2/L_{max}^2)}\Big)^{k/n}(f(x^0) - f^*) \tag{7}$$

Comparing the complexity bounds for the cyclic variant with the corresponding bounds proved in Theorem 1 for the randomized variant, we see that since $L \geq L_{max}$ in general, the numerator in equation 5 is $O(n^2)$, in contrast to $O(n)$ term in equation 3. A similar factor of n in seen in comparing equation 3 to equation 7, when we note that $(1-\epsilon)^{1/n} \approx 1 - \epsilon/n$ for small values of ϵ. The bounds in Theorem 3 are deterministic, however, rather than being bounds on expected nonoptimality, as in Theorem 1.

3.6 Separable Regularized Coordinate Descent

Theorem 4. *Suppose that Assumption 2 holds, $\alpha \equiv 1/L_{\max}$. Then for all $k \geq 0$, we have*

$$E\left(h\left(x^k\right)\right) - h^* \leq \left(1 - \frac{\sigma}{nL_{\max}}\right)^k \left(h\left(x^0\right) - h^*\right)$$

The result here is almost the same as that in unconstrained coordinate descent. For the subproblem in the algorithm 2, the operation of solving this subproblem is refereed to as "shrink operation". We can think about this subproblem as minimizing the taylor expansion to the order two together with the penalized term.

4 Numerical Experiments Detail

4.1 Setup and Implementation

We consider solving synthetic instances of the linear regression model with least-squares objective function:

$$f^* := \min_{\beta \in R^p} f(\beta) = \|y - X\beta\|_2^2$$

using a baseline method Gradient Descent(GD), Randomized Coordinate Descent (RCD 1) Method (both were implemented in the submitted .R code named "RCD.R"), and Accelerated Randomized Coordinate Descent (ARCD 3) Method (implemented in the submitted .R code named "Accelerated_RCD.R"). The mechanism for generating the data(y, X) are described in the supplementary materials; functions for generating the simulation data could be found in the submitted .R code named "Experiment.R".

We also consider solving the linear regression model with L_1 penalty:

$$f^* := \min_{\beta \in R^p} f(\beta) = \|y - X\beta\|_2^2 + \lambda\|\beta\|_1$$

using the Separable Coordinate Descent (SpCD 2) Method (implemented in the submitted .R code named "Separable_RCD.R").

Algorithm	RCD	ARCD	GD
$\kappa = 10$	0.246 / 1310	0.204 / 1365	0.008 / 48
$\kappa = 100$	1.284 / 5699	1.057 / 5622	0.036 / 207
$\kappa = 1000$	4.468 /13398	3.736 / 12734	0.094 / 471

Table 2: The true run time (s) / numbers of iterations for each algorithm to converge with the criterion $f(x^k) - f^* \leq 0.001$. Here we repeat the experiment for 50 times and take the average value in the present result

4.2 Results and Analysis

Figure 4 shows the optimality gap versus numbers of iteration for solving different instances of linear regression with different conditinoal numbers of the matrix $X^T X$ using RCD, ARCD and GD algorithms. In each plot, the vertical axis is the objective value optimality gap $f(\beta^k) - f^*$ in log scale. The horizontal axis is the numbers of iteration. Each column corresponds to an instance with the prescribed condition number κ of $X^T X$.

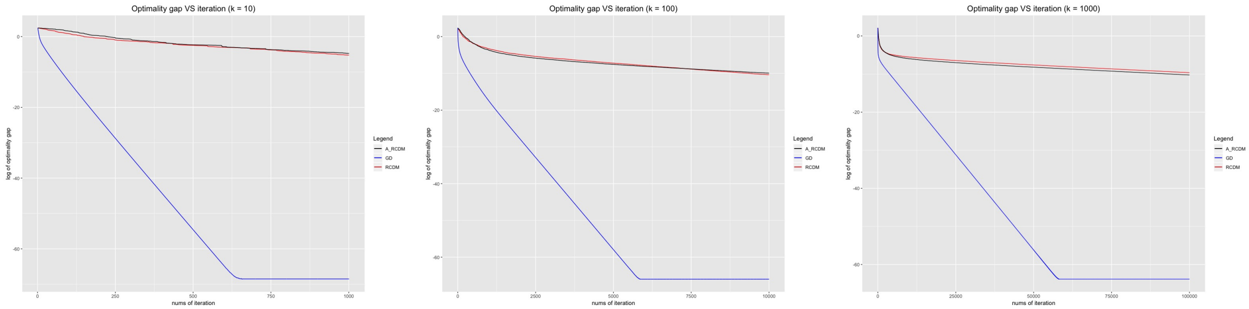

Figure 1: Gap vs.iter (k=10)

Figure 2: Gap vs.iter (k=100)

Figure 3: Gap vs.iter (k=1000)

Figure 4: For different κ, we show the convergence result with vertical axis is the objective value optimality gap $f(\beta^k) - f^*$ in log scale. The horizontal axis is the numbers of iteration.

We also present the average run time for each algorithm to converge to $\epsilon = 0.001$ close optimality are in table 2. We found that the convergence of traditional GD show much advantage over the other two methods. That is because in quadratic form, we are very easily to calculate closed form solution of the gradient in each step. The power of Coordinate descent method may lies in the problem where it is too expensive to compute the gradient compared to the individual derivatives. As for ARCD, it shows a little advantage over RCD in the convergence comparision.

We also perform time complexity analysis for RCD, ARCD, and SpCD algorithms. We present our analysis as an ensemble plot Figure 4 - 12 (provides in Appendix: Additional Experiments, section D.2). Each row of the figure matrix corresponds to three instances of the algorithms; And the columns represent complexity of ϵ under σ-strong convexity assumption (S-C), complexity of σ under σ-strong convexity assumption (S-C), and complexity of ϵ under convexity assumption (C), respectively.

Please also refer our Github page for this project: Ensemble Methods of Coordinate Descent Algorithms for more details.

A Algorithm

Algorithm 3 Accelerated Randomized Coordinate Descent for (1)

```
Choose $x_0 \in \mathcal{R}^n$;
Set $k \leftarrow 0$, $v^0 \leftarrow x^0$, $\gamma_{-1} \leftarrow x^0$;
repeat
    Choose $\gamma_k$ to be the large root of
        $\gamma_k^2 - \frac{\gamma_k}{n} = (1 - \frac{\gamma_k \sigma}{n})\gamma_{k-1}^2$
    Set $y^k \leftarrow \alpha_k v^k + (1 - \alpha_k)x^k$;
    Choose index $i_k \in \{1, 2, ..., n\}$ with uniform probability and set $d^k = [\nabla f(y^k)]_{i_k} e_{i_k}$;
    Set $x^{k+1} \leftarrow y^k - (1/L_{i_k})d^k$;
    Set $v^{k+1} \leftarrow \beta_k v^k + (1 - \beta^k)y^k - (\gamma_k / L_{i_k})d^k$;
    $k \leftarrow k + 1$;
until termination test satisfied
```

B Assumption

Assumption 1. *The function f in (1)is convex and uniformly LipSchitz continuously differentiable, and attains its minimum value f^* on a set $\mathcal{S}$. There is a finite R_0 such that the level set for f defined by x^0 is bounded, that is,*

$$\max_{x^* \in \mathcal{S}} \max_{x} \left\{ \|x - x^*\| : f(x) \leq f\left(x^0\right) \right\} \leq R_0$$

Assumption 2. *The function f in (2) is uniformly Lipschitz continuously differentiable and strongly convex with modulus $\sigma > 0$. The functions $\Omega_o, i = 1, ..., n$ are convex. The function h in (2) attains its minimum value h^* at a unique point x^**

C Additional Proof Details

The detail proof of the theorems can be found in literature. Here we just highlight some key points and ideas under the proof.

Proof of Theorem 1.
Sketch of Proof:
Detail proof is shown in Wright 2015. The proof just use Taylor expansion and basic calculus. By choosing $\alpha_k = 1/L_{\max}$ and

$$\phi_k := E\left(f\left(x^k\right)\right) - f^*$$

we can use the taylor expansion to derive that

$$\phi_{k+1} \leq \phi_k - \frac{1}{2nL_{\max}} E\left(\left\|\nabla f\left(x^k\right)\right\|^2\right) \leq \phi_k - \frac{1}{2nL_{\max}}\left[E\left(\left\|\nabla f\left(x^k\right)\right\|\right)\right]^2$$

Some for each function setting, we just use different tricks to find the lower bound for $\left[E\left(\left\|\nabla f\left(x^k\right)\right\|\right)\right]$

- convex setting

  $$f\left(x^k\right) - f^* \leq \nabla f\left(x^k\right)^T\left(x^k - x^*\right) \leq \left\|\nabla f\left(x^k\right)\right\|\left\|x^k - x^*\right\| \leq R_0\left\|\nabla f\left(x^k\right)\right\|$$

  we can then derive

  $$E\left(\left\|\nabla f\left(x^k\right)\right\|\right) \geq \frac{1}{R_0}\phi_k$$

  it follows

  $$\frac{1}{\phi_k} \geq \frac{1}{\phi_0} + \frac{k}{2nL_{\max}R_0^2} \geq \frac{k}{2nL_{\max}R_0^2}$$

- $\sigma-$ strong convex
  From the definition of σ strong convex

  $$f^* \geq f\left(x^k\right) - \frac{1}{2\sigma}\left\|\nabla f\left(x^k\right)\right\|^2$$

  Then it follows

  $$\phi_{k+1} \leq \phi_k - \frac{\sigma}{nL_{\max}}\phi_k = \left(1 - \frac{\sigma}{nL_{\max}}\right)\phi_k$$

□

Proof of Theorem 2.
Sketch of Proof:
(1) Construct $r_{k+1}^2 = \|v_{k+1} - x^*\|^2$. Derive the inequality

$$b_{k+1}^2 E_{i_k}(r_{k+1}^2) \leq b_k^2 r_k^2 - 2a_{k+1}^2(E_{i_k}(f(x^{k+1})) - f^*) + 2a_k^2\left(f(x^k) - f^*\right)$$

(2) Denote $\phi_{k+1} = E[f(x^{k+1})]$ and get

$$2a_{k+1}^2(\phi_{k+1} - f^*) + b_{k+1}^2 E(r_{k+1}^2) \leq 2a_0^2(f(x^0) - f^*) + b_0^2\left\|x^0 - x^*\right\|^2$$

Note that $a_0 = 1/n$ and $b_0 = \sqrt{2L_{max}}$.
(3) Estimate the growth rate of a_k, b_k. Find $a_k \geq \frac{1}{\sqrt{\sigma/L_{max}}}[Q_1^{k+1} - Q_2^{k+1}]$ and $b_k \geq [Q_1^{k+1} + Q_2^{k+1}]$, where $Q_1 = 1 + \frac{\sqrt{\sigma/L_{max}}}{2n}$ and $Q_2 = 1 - \frac{\sqrt{\sigma/L_{max}}}{2n}$. Plug the growth rate into the inequality in (2). The proof is complete.
Detail of Proof:
Detail Proof is shown in Nesterov 2012. We note that (i) a_k, b_k are not mentioned in Theorem 2 and not needed by Algorithm 2, but in the proof of convergence rate, using a_k, b_k as intermediate tools is super helpful; (ii) Theorem 2 set $b_0 = \sqrt{2L_{max}}$, while Nesterov 2012 set $b_0 = 2$, thus in Nesterov 2012, $S_0 = \sup_{x^* \in \mathcal{S}} 2\|x^0 - x^*\|^2 + (f(x^0) - f^*)/n^2$.

Let x^k and v^k be the implementations of corresponding random variables generated by **??** after k iterations. Denote $r_k^2 = \left\|v^k - x^*\right\|^2$. We have

$$\begin{aligned}
r_{k+1}^2 &= \left\|\beta_k v^k + (1-\beta_k)y^k - (\frac{\gamma_k}{L_{i_k}}d^k) - x^*\right\|^2 \\
&= \left\|\beta_k v^k + (1-\beta_k)y^k - x^*\right\|^2 + \left\|\frac{\gamma_k}{L_{i_k}}d^k\right\|^2 + 2\gamma_k\left\langle\frac{d^k}{L_{i_k}}, (x^* - \beta_k v^k - (1-\beta_k)y^k)\right\rangle \\
&= \left\|\beta_k v^k + (1-\beta_k)y^k - x^*\right\|^2 + \frac{\gamma_k^2}{L_{i_k}}[\nabla f(y^k)]_{i_k}^2 + 2\gamma_k\left\langle[\nabla f(y^k)]_{i_k}, (x^* - \beta_k v^k - (1-\beta_k)y^k)\right\rangle
\end{aligned}$$

Then using representation

$$v_k = y_k + \frac{1-\alpha_k}{\alpha_k}(y_k - x_k),$$

and

$$f(y^k) - f(y^k - \frac{1}{L_{i_k}}d^k) \geq \frac{1}{2L_{i_k}}[\nabla f(y^k)]_{i_k}^2$$

we obtain

$$r_{k+1}^2 \leq \left\|\beta_k v^k + (1-\beta_k)y^k - x^*\right\|^2 + 2\gamma_k^2(f(y^k) - f(y^k - \frac{1}{L_{i_k}}d^k))$$
$$+ 2\gamma_k \left\langle [\nabla f(y^k)]_{i_k}, (x^* - y^k + \frac{\beta_k(1-\alpha_k)}{\alpha_k}(x^k - y^k))\right\rangle$$

Taking the expectation of both sides in i_k, we obtain

$$\begin{aligned}
E_{i_k}(r_{k+1}^2) &\leq \beta_k r_k^2 + (1-\beta_k)\left\|y^k - x^*\right\|^2 \\
&\quad + 2\gamma_k^2[f(y^k) - E_{i_k}(f(x_{k+1}))] + 2\frac{\gamma_k}{n}\left\langle \nabla f(y^k), (x^* - y^k + \frac{\beta_k(1-\alpha_k)}{\alpha_k}(x^k - y^k))\right\rangle \\
&\leq \beta_k r_k^2 + (1-\beta_k)\left\|y^k - x^*\right\|^2 \\
&\quad + 2\gamma_k^2[f(y^k) - E_{i_k}(f(x_{k+1}))] + 2\frac{\gamma_k}{n}\left[\nabla f(y^k)(x^* - y^k) + \frac{\beta_k(1-\alpha_k)}{\alpha_k}\nabla f(y^k)(x^k - y^k)\right] \\
&\overset{convexity}{\leq} \beta_k r_k^2 + (1-\beta_k)\left\|y^k - x^*\right\|^2 \\
&\quad + 2\gamma_k^2[f(y^k) - E_{i_k}(f(x_{k+1}))] + 2\frac{\gamma_k}{n}\left[f^* - f(y^k) - \frac{1}{2}\sigma\left\|y^k - x^*\right\|^2 + (x^* - y^k + \frac{\beta_k(1-\alpha_k)}{\alpha_k}(x^k - y^k))\right]
\end{aligned}$$

According to the choice of γ_k and β_k

$$\gamma_k^2 - \frac{\gamma_k}{n} = \frac{\beta_k\gamma_k}{n}\frac{1-\alpha_k}{\alpha_k}$$

$$\beta_k = 1 - \frac{\gamma_k\sigma}{n}$$

$$\begin{aligned}
E_{i_k}(r_{k+1}^2) &\leq \beta_k r_k^2 + (1-\beta_k - \frac{\gamma_k\sigma}{n})\left\|y^k - x^*\right\|^2 \\
&\quad + (2\gamma_k^2 - 2\frac{\gamma_k}{n} - \frac{2\beta_k\gamma_k}{n}\frac{(1-\alpha_k)}{\alpha_k})f(y^k) - 2\gamma_k^2 E_{i_k}(f(x_{k+1})) + 2\frac{\gamma_k}{n}\left[f^* + \frac{\beta_k(1-\alpha_k)}{\alpha_k}f(x^k))\right] \\
&= \beta_k r_k^2 - 2\gamma_k^2 E_{i_k}(f(x_{k+1})) + \frac{\gamma_k}{n}\left[f^* + \frac{\beta_k(1-\alpha_k)}{\alpha_k}f(x^k))\right]
\end{aligned}$$

We denote $a_0 = \frac{1}{n}$, $b_0 = 2$, $a_{k+1} = \gamma_k b_{k+1}$ and $b_{k+1} = \frac{b_k}{\sqrt{\beta_k}}$. Note that

$$b_{k+1}^2 = \frac{1}{\beta_k}b_k^2, \quad a_{k+1}^2 = \gamma_k^2 b_{k+1}^2, \quad \gamma_k\frac{\beta_k(1-\alpha_k)}{n\alpha_k} = \frac{a_k^2}{b_{k+1}^2}$$

Therefore, multiply the last inequality by b_{k+1}^2, we obtain

$$b_{k+1}^2 E_{i_k}(r_{k+1}^2) \leq b_k^2 r_k^2 - 2a_{k+1}^2(E_{i_k}(f(x_{k+1})) - f^*) + 2a_k^2\left(f(x^k) - f^*\right)$$

Taking the expectation of both sides of the inequality conditional on the observed implementation of random variables $\{i_0, ..., i_{k-1}\}$ before step k, we get

$$2a_{k+1}^2(\phi_{k+1} - f^*) + b_{k+1}^2 E(r_{k+1}^2) \leq 2a_k^2(\phi_k - f^*) + b_k^2 r_k^2 \leq 2a_0^2(f(x_0) - f^*) + b_0^2 \left\|x_0 - x^*\right\|^2$$

It remains to estimate the growth of coefficients a_k and b_k. We have

$$b_k^2 = \beta_k b_{k+1}^2 = (1 - \frac{\sigma}{n}\gamma_k) b_{k+1}^2 = \Big(1 - \frac{\sigma}{n}\frac{a_{k+1}}{b_{k+1}}\Big) b_{k+1}^2$$

Thus $\frac{\sigma}{n} a_{k+1} b_{k+1} \leq b_{k+1}^2 - b_k^2 \leq 2b_{k+1}(b_{k+1} - b_k)$, and we conclude that

$$b_{k+1} \geq b_k + \frac{\sigma}{2n} a_k$$

On the other hand, $\frac{a_{k+1}^2}{b_{k+1}^2} - \frac{a_{k+1}}{nb_{k+1}} = \frac{\beta_k a_k^2}{b_k^2} = \frac{a_k^2}{b_{k+1}^2}$. Therefore,

$$\frac{1}{n} a_{k+1} b_{k+1} \leq a_{k+1}^2 - a_k^2 \leq 2a_{k+1}(a_{k+1} - a_k)$$

and we obtain

$$a_{k+1} \geq a_k + \frac{1}{2n} b_k$$

Further, denote $Q_1 = 1 + \frac{\sqrt{\sigma}}{2n}$ and $Q_2 = 1 - \frac{\sqrt{\sigma}}{2n}$ and using the inequalities $b_{k+1} \geq b_k + \frac{\sigma}{2n} a_k$, $a_{k+1} \geq a_k + \frac{1}{2n} b_k$, it is easy to prove by induction that

$$a_k \geq \frac{1}{\sqrt{\sigma}}[Q_1^{k+1} - Q_2^{k+1}], \quad b_k \geq [Q_1^{k+1} + Q_2^{k+1}]$$

Finally, using the inequality $(1 + t)^k - (1 - t)^k \geq 2kt$, $t \geq 0$, we obtain

$$Q_1^{k+1} - Q_2^{k+1} \geq \frac{k+1}{n}\sqrt{\sigma}$$

Therefore, for any $k \geq 0$, we have

$$\begin{aligned}
\phi_k - f^* &\leq \frac{1}{a_k^2}\Big[\frac{1}{n^2}(f(x_0) - f^*) + 2\left\|x_0 - x^*\right\|^2\Big] \\
&\leq \sigma\Big[\Big(1 + \frac{\sqrt{\sigma}}{2n}\Big)^{k+1} - \Big(1 - \frac{\sqrt{\sigma}}{2n}\Big)^{k+1}\Big]^{-2}\Big[\frac{1}{n^2}(f(x_0) - f^*) + 2\left\|x_0 - x^*\right\|^2\Big] \\
&\leq (\frac{n}{k+1})^2\Big[\frac{1}{n^2}(f(x_0) - f^*) + 2\left\|x_0 - x^*\right\|^2\Big]
\end{aligned}$$

□

Proof of Theorem 4.

Sketch of Proof:

Detail proof is shown in Wright 2015 and Richtárik and Takáč 2014. we define the function

$$H\left(x^k, z\right) := f\left(x^k\right) + \nabla f\left(x^k\right)^T\left(z - x^k\right) + \frac{1}{2}L_{\max}\left\|z - x^k\right\|^2 + \lambda\Omega(z)$$

The motivation to bound this function follows from

$$E_{i_k} h\left(x^{k+1}\right) = \frac{1}{n}\sum_{i=1}^{n}\left[f\left(x^k + \left(z_i^k - x_i^k\right)e_i\right) + \lambda\Omega_i\left(z_i^k\right) + \lambda\sum_{j\neq i}\Omega_j\left(x_j^k\right)\right] = \frac{n-1}{n}h\left(x^k\right) + \frac{1}{n}H\left(x^k, z^k\right)$$

- With the relation that $H(x^k, z^k) = \min_z H(x^k, z)$. So it is fundamental to estimate $H(x^k, z^k)$ from above in terms of $h(x^k)$. Here in strongly convex case, we have

$$\begin{aligned} H\left(x^k, z^k\right) &= \min_z H\left(x^k, z\right) \\ &\leq \min_z h(z) + \frac{1}{2}\left(L_{\max} - \sigma\right)\left\|z - x^k\right\|^2 \\ &\leq \min_{\alpha \in [0,1]} h\left(\alpha x^* + (1-\alpha)x^k\right) + \frac{1}{2}\left(L_{\max} - \sigma\right)\alpha^2\left\|x^k - x^*\right\|^2 \\ &\leq \min_{\alpha \in [0,1]} \alpha h^* + (1-\alpha)h\left(x^k\right) + \frac{1}{2}\left[\left(L_{\max} - \sigma\right)\alpha^2 - \sigma\alpha(1-\alpha)\right]\left\|x^k - x^*\right\|^2 \\ &\leq \frac{\sigma}{L_{\max}} h^* + \left(1 - \frac{\sigma}{L_{\max}}\right) h\left(x^k\right) \end{aligned}$$

  The cleverness of the proof lies in it represents z as $\alpha x^* + (1-\alpha)x^k$, and then use the result of zero-order representation of convex function

$$h(\alpha x + (1-\alpha)y) \leq \alpha h(x) + (1-\alpha)h(y) - \frac{1}{2}\sigma\alpha(1-\alpha)\|x-y\|^2$$

  to connect z with h^* and x^k. It then follows that

$$E\left(h\left(x^{k+1}\right)\right) - h^* \leq \left(1 - \frac{\sigma}{nL_{\max}}\right)^k\left(E\left(h\left(x^k\right)\right) - h^*\right)$$

- From lemma 3 in Richtárik and Takáč 2014. If we assume that $\varepsilon < \min\left\{\mathcal{R}_L^2\left(x_0\right), F\left(x_0\right) - F^*\right\}$, $c = \frac{2nL_{\max}\mathcal{R}^2(x_0)}{\epsilon} > 1$ we have

$$\mathbf{E}\left[\xi_{k+1} | \xi_k\right] \leq \max\left\{1 - \frac{\epsilon}{2n\mathcal{R}_L^2\left(x_0\right)}, 1 - \frac{1}{2n}\right\}\xi_k = \left(1 - \frac{1}{c}\right)\xi_k$$

  then it follows

$$E\left(h\left(x^{k+1}\right)\right) - h^* \leq \left(1 - \frac{1}{c}\right)^k\left(E\left(h\left(x^k\right)\right) - h^*\right)$$

□

D Additional Experiments

D.1 Implementation Detail

The data (X, y) for the linear regression problems is generated as follows. For a given number of samples n and problem dimension p(in the experiments, we used $n = 100, p = 50$, we generate a random design matrix $\bar{X} \in R^{p\times n}$ with fixed condition number κ for X^TX in following ways. We first generate random orthogonal matrix $U \in R^{p\times p}$ and $V \in R^{n\times n}$. Then we set the diagonal matrix D of singular values linearly such that the smallest singular value is $\frac{1}{\sqrt{\kappa}}$ and the largest singular value of D is 1. We then compute the final design matrix $X = UDV^T$ and there for the condition number of X^TX becomes κ. We generate the response vector y using the linear model $y \sim \mathcal{N}(X\beta^*, \sigma^2)$, with true model β^* chosen randomly by a Gaussian distribution as well.

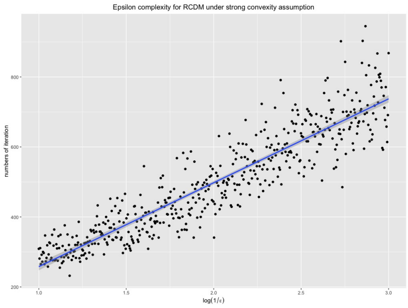

Figure 5: RCD ϵ complexity under (S-C)

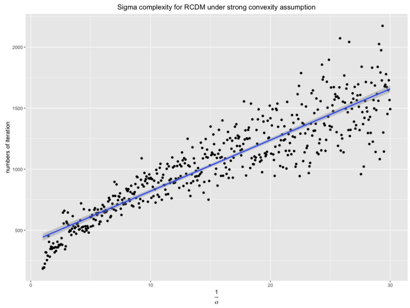

Figure 6: RCD σ complexity under (S-C)

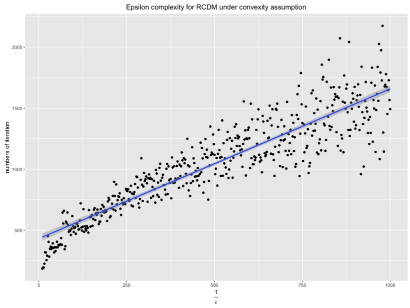

Figure 7: RCD ϵ complexity under (C)

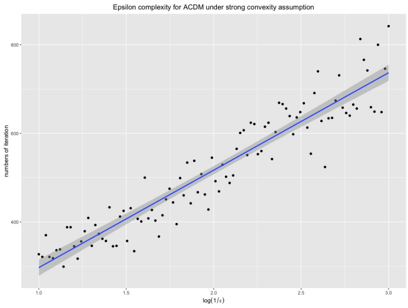

Figure 8: ARCD ϵ complexity under (S-C)

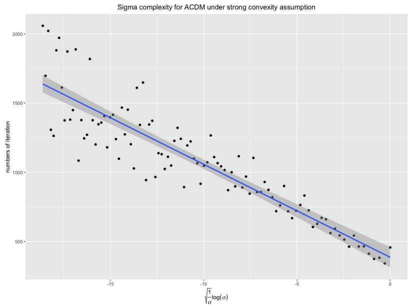

Figure 9: ARCD σ complexity under (S-C)

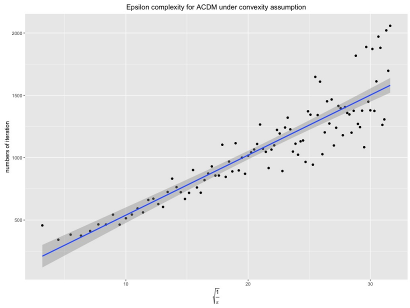

Figure 10: ARCD ϵ complexity under (C)

D.2 Verification of time complexity

- The complexity analysis plots for RCD algorithm are provided as Figure 5 - 7:

- The complexity analysis plots for ARCD algorithm are provided as Figure 8 - 10:

- The complexity analysis plots for SpCD algorithm are provided as Figure 11 - 13:

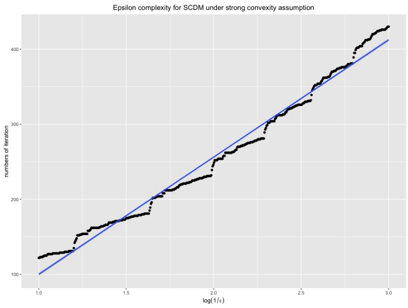

Figure 11: SpCD ϵ complexity under (S-C)

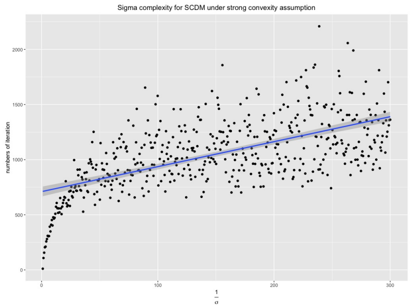

Figure 12: SpCD σ complexity under (S-C)

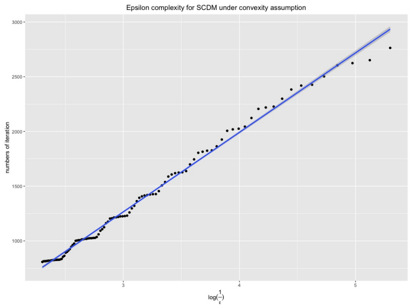

Figure 13: SpCD ϵ complexity under (C)

E Discussion

E.1 Order of coordinates

- greedy strategy. It means that the block with the largest descent or guaranteed descent is chosen. But it is very time consuming.
- Cyclic choice. the complexity analysis of a cyclic CD method in satisfying generality has not yet been done.
- Random choice. Recent efforts suggest that complexity results are perhaps more readily obtained for randomized methods and that randomization can actually improve the convergence rate.

References

[Aus76] Alfred Auslender. "Optimisation". In: *Méthodes numériques* (1976).

[Gur+17] Mert Gurbuzbalaban et al. "When cyclic coordinate descent outperforms randomized coordinate descent". In: *Advances in Neural Information Processing Systems*. 2017, pp. 6999–7007.

[Nes12] Yu Nesterov. "Efficiency of coordinate descent methods on huge-scale optimization problems". In: *SIAM Journal on Optimization* 22.2 (2012), pp. 341–362.

[OR00] James M Ortega and Werner C Rheinboldt. *Iterative solution of nonlinear equations in several variables*. SIAM, 2000.

[Pow73] Michael JD Powell. "On search directions for minimization algorithms". In: *Mathematical programming* 4.1 (1973), pp. 193–201.

[RT14] Peter Richtárik and Martin Takáč. "Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function". In: *Mathematical Programming* 144.1-2 (2014), pp. 1–38.

[Wri15] Stephen J Wright. "Coordinate descent algorithms". In: *Mathematical Programming* 151.1 (2015), pp. 3–34.

[Zad70] Norman Zadeh. "Note—A Note on the Cyclic Coordinate Ascent Method". In: *Management Science* 16.9 (1970), pp. 642–644.