CS2910 - Optimization Summer 2023

Lecture 7 — Distributed Optimization and ADMM

Lecturer: Yunwei Ren *Scribed by Zhidan Li*

1 Overview

In this lecture, we focus on the distributed optimization problem. Under the distributed setting, we will show our gradient descent algorithm can be computed in parallel. This kind of parallel optimization algorithm will be of great significance when the dataset is extremely large.

2 Introduction

Recall ERM: given a dataset $\{(x_i, y_i)\}_{i=1}^n \subseteq \mathbb{R}^{d+1}$,

$$\min_{w \in \mathbb{R}^d} f(w) \triangleq \frac{1}{n} \sum_{i=1}^n \ell(h(x_i; w), y_i) + R(w). \tag{1}$$

A problem is, if the dataset $\{(x_i, y_i)\}_{i=1}^n$ is extremely large, we have to store the dataset in a distributed system (or multiple machines). This means we need to design a distributed optimization algorithm.

Theoretical model:

- There are m machines $M_1, \ldots, M_m$ and they can communicate with each other.
- Each machine contains a subset of the dataset $S_1, \ldots, S_m$ satisfying $\biguplus_{i=1}^m S_i = [n]$ and so that $\{(x_j, y_j)\}_{j \in S_j}$ stored in machine M_i.
- For the computation model of machines, there are two cases: infinite computation power (the local machine can compute arbitrary amount of things in a short time) in most of this lecture; finite computation power.
- Our goal is to minimize the communication cost while minimizing objects.

2.1 Attempt 1

For each machine M_j, we compute

$$w_j^* = \operatorname*{argmin}_w \left\{ \frac{1}{|S_j|} \sum_{i \in S_j} \ell(h(x_i; w), y_i) + R(w) \right\}.$$

Then we combine $\left\{ w_j^* \right\}_{j=1}^m$ in a sophistical way. However, this kind of attempt does not work, since $S_1, \ldots, S_m$ are not necessarily a list of i.i.d. variables.

2.2 Attempt 2

Now we try to communicate the gradients. Consider the gradient of the objective function

$$\frac{1}{n}\sum_{i=1}^{n}\nabla_w \ell(h(x_i;w),y_i)+\nabla_w R(w)=\frac{1}{n}\sum_{j=1}^{m}\sum_{i\in S_j}\nabla_w \ell(h(x_i;w),y_i)+\nabla_w R(w).$$

Then we can design the following parallel algorithm: At each iteration t,

- Each M_j computes the local gradient $\nabla_j=\sum_{i\in S_j}\nabla_w \ell(h(x_i;w),y_i)$.
- M_j sends ∇_j to anther machine M_0.
- M_0 computes the global gradient $\nabla=\frac{1}{n}\sum_{j=1}^{m}\nabla_j+\nabla R(w)$.
- M_0 broadcasts ∇ to every machine $M_1,\ldots,M_m$.
- Each M_j updates w (gradient descent) with ∇.

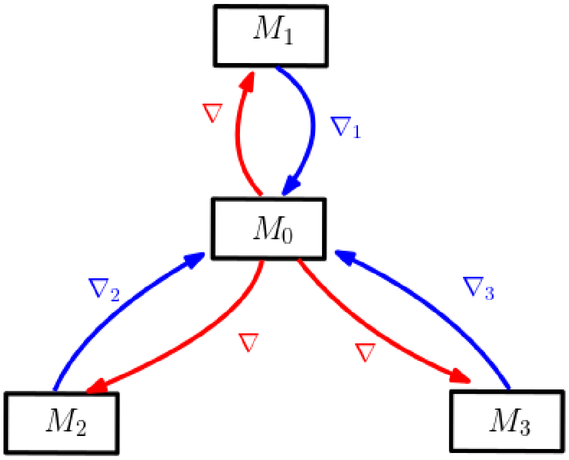

Figure 1: one iteration of the algorithm

In each iteration, the communication cost is $\Theta(md)$. Since the number of iterations depends on the convergence rate of gradient descent. However, the cost will be very large when the smoothness of the objective function is not good due to the convergence rate of gradient descent.

Key observation: the attempt does not utilize the infinite computation power.

3 Alternating Direction Method of Multipliers (ADMM)

We consider a slightly general problem. Consider the following optimization problem

$$\min_{w\in\mathbb{R}^d}\frac{1}{m}\sum_{i=1}^{m}f_j(w) \tag{2}$$

where for each $j \in [m]$, f_j is a convex function and only depends on the dataset stored in M_j. Note that to solve (1), for each $j \in [m]$, we let

$$f_j(w) = \frac{m}{n} \sum_{i \in S_j} \ell(h(x_i; w), y_i) + R(w).$$

To solve (2), we consider the following optimization problem with constraints

$$\min_{w_1, \ldots, w_m \in \mathbb{R}^d} \frac{1}{m} \sum_{j=1}^{m} f_j(w_j) \quad s.t.\ w_1 = \ldots = w_m = w. \tag{3}$$

For some technical reason, we consider the following optimization problem

$$\min_{w_1, \ldots, w_m \in \mathbb{R}^d} \frac{1}{m} \sum_{j=1}^{m} \left(f_j(w_j) + \lambda \|w_j - w\|_2^2 \right) \quad s.t.\ w_1 = \ldots = w_m = w \tag{4}$$

where $\lambda > 0$ is a parameter we will choose later. Then it makes sense to consider its Lagrangian

$$L(\alpha_1, \ldots, \alpha_m, w_1, \ldots, w_m, w) = \frac{1}{m} \sum_{j=1}^{m} \left(f_j(w_j) + \lambda \|w_j - w\|_2^2 + \langle \alpha_j, w_j - w \rangle \right).$$

It is not hard to verify that the slater's condition holds. Then we consider its dual

$$\max_{\alpha \in \left(\mathbb{R}^d\right)^m} G(\alpha) \overset{\triangle}{=} \min_{w_1, \ldots, w_m \in \mathbb{R}^d} L(\alpha, w_1, \ldots, w_m, w) \tag{5}$$

Note that, $G(\alpha)$ is concave. That is to say, we can do 'gradient ascent' to solve (5). Now we show how to compute its gradient.

Lemma 1. *Let $h : \mathbb{R}^d \to \mathbb{R}$ be a strictly convex function and $A \in \mathbb{R}^{d' \times d}$ be a matrix. Put $F(\beta) = \min_y \{h(y) + \langle \beta, Ay \rangle\}$. Under some regularity conditions, $\nabla F(\beta) = Ay(\beta)$ where $y(\beta) = \operatorname{argmin}_y \{h(y) + \langle \beta, Ay \rangle\}$.*

Proof. Note that $F(\beta) = h(y(\beta)) + \langle \beta, Ay(\beta) \rangle$. Then,

$$\begin{aligned} \nabla F(\beta) &= \left(\nabla_\beta y(\beta)\right)^\top \nabla h(y(\beta)) + Ay(\beta) + \left(\nabla_\beta y(\beta)\right)^\top A^\top \beta \\ &= \left(\nabla_\beta y(\beta)\right)^\top \left(\nabla h(y(\beta)) + A^\top \beta \right) + Ay(\beta). \end{aligned}$$

Since $y(\beta) = \operatorname{argmin}_y \{h(y) + \langle \beta, Ay \rangle\}$, it holds that $\nabla h(y(\beta)) + A^\top \beta = 0$. Then $\nabla F(\beta) = Ay(\beta)$. □

As a corollary, we have the following result for G.

Corollary 2. *For each $j \in [m]$, it holds that*

$$\nabla_{\alpha_j} G(\alpha) = w_j^* - w^*,$$

where $(w_1^*, \ldots, w_m^*, w^*)$ is the minimizer of $(w_1, \ldots, w_m, w) \mapsto L(\alpha, w_1, \ldots, w_m, w)$.

Remark 1. The corollary means, as long as we can efficiently solver the inner minimizing problem (in a distributed way), we can solve the original optimization problem. However, the inner minimization is not usually easy since w is shared across machines.

Even it is not easy to do the inner minimization, we have the two following observations which are of great significance.

- **Observation 1:** For fixed w, the optimization problem

$$\min_{w_1,\ldots,w_m} \frac{1}{m} \sum_{j=1}^{m} \Big(f_j(w_j) + \lambda \|w_j - w\|^2 + \langle \alpha_j, w_j - w \rangle \Big)$$

can be solved in a distributed way.

- **Observation 2:** For fixed $w_1, \ldots, w_m$, minimizing over w is easy:

$$\nabla_w L = -2\lambda \sum_{j=1}^{m} (w_j - w) - \sum_{j=1}^{m} \alpha_j = 0$$
$$\implies w = \frac{1}{2m\lambda} \sum_{j=1}^{m} \alpha_j + \frac{1}{m} \sum_{j=1}^{m} w_j.$$

Based on the two observations, we would minimize $w_1, \ldots, w_m$ and w in an alternative way.

Definition 3 (ADMM)**.** *Consider the problem*

$$\min_{w \in \mathbb{R}^d} \frac{1}{m} \sum_{j=1}^{m} f_j(w)$$

where $f_j : \mathbb{R}^d \to \mathbb{R}$ are convex functions. Suppose that f_j only depends on the dataset stored in M_j. Without loss of generality assume that all f_j are non-negative. ADMM updates the parameters using the following rules

$$w_j^{(t+1)} = \operatorname*{argmin}_{w_j} \left\{ f_j(w_j) + \lambda \left\| w_j - w^{(t)} \right\|^2 + \left\langle \alpha_j^{(t)}, w_j - w^{(t)} \right\rangle \right\} \tag{6}$$

$$w^{(t+1)} = \frac{1}{2m\lambda} \sum_{j=1}^{m} \alpha_j^{(t)} + \frac{1}{m} \sum_{j=1}^{m} w_j^{(t+1)} \tag{7}$$

$$\alpha_j^{(t+1)} = \alpha_j^{(t)} - \eta \left(w^{(t+1)} - w_j^{(t+1)} \right) \tag{8}$$

where $\eta > 0$, $\lambda > 0$ are two positive parameters which can be chosen, and $\alpha_1^{(0)}, \ldots, \alpha_m^{(0)}, w_1^{(0)}, \ldots, w_m^{(0)}, w^{(0)}$ is the starting point. Precisely, ADMM works as: suppose that at the beginning of each iteration t, each machine M_j locally stores $w_j^{(t)}$ and $\alpha_j^{(t)}$ and a copy of $w^{(t)}$.

- *Each machine locally computes $w_j^{(t+1)}$ by* (6)*.*
- *All machines send $w_j^{(t+1)}$ and $\alpha_j^{(t)}$ to a center server.*

- *The center server computes $w^{(t+1)}$ by (6).*
- *The center server broadcasts $w^{(t+1)}$ to all machines.*
- *Each machine locally computes $\alpha_j^{(t+1)}$ by (8).*

Remark 2. Note that, the cost of each iteration of ADMM is $\Theta(md)$. Unfortunately each iteration of ADMM does not necessarily achieve the 'local minimum'. And the convergence rate of ADMM is not trivial.

3.1 Convergence rate of ADMM

Now we show the convergence rate of ADMM.

Proposition 4. *Under the definitions and settings in Definition 3, for any given $T > 0$, choose $\eta = 2\sqrt{T}$ and $\lambda = \eta/2$. Initialize all $\alpha_j^{(0)} = 0$. Then there exists $t^* \le T$ such that for all w,*

$$\frac{1}{m}\sum_{j=1}^m f_j(w_j^*) \le \frac{1}{m}\sum_{j=1}^m f_j(w) + \frac{1}{\sqrt{T}}\left\|w - w^{(0)}\right\|^2,$$

$$\frac{1}{m}\sum_{j=1}^m \left\|w^{(t^*)} - w_j^{(t^*)}\right\|^2 \le \frac{1}{m\sqrt{T}}\sum_{j=1}^m f_j(w) + \frac{1}{T}\left\|w - w^{(0)}\right\|^2.$$

To prove Proposition 4, firstly we prove the following lemma.

Lemma 5. *Under the settings of Proposition 4, when all $\alpha_j^{(0)} = 0$, then for all $t \ge 0$, $\sum_{j=1}^m \alpha_j^{(t)} = 0$. As a result, (6) can be written as*

$$w^{(t+1)} = \frac{1}{m}\sum_{j=1}^m w_j^{(t+1)}.$$

Proof. We prove it by induction. By (8) and (7), it holds that

$$\begin{aligned}\frac{1}{m}\sum_{j=1}^m \alpha_j^{(t+1)} &= \frac{1}{m}\sum_{j=1}^m \alpha_j^{(t)} - \eta\left(w^{(t+1)} - \frac{1}{m}\sum_{j=1}^m w_j^{(t+1)}\right)\\ &= \frac{1}{m}\sum_{j=1}^m \alpha_j^{(t)} - \eta\left(\frac{1}{2m\lambda}\sum_{j=1}^m \alpha_j^{(t)} + \frac{1}{m}\sum_{j=1}^m w_j^{(t+1)} - \frac{1}{m}\sum_{j=1}^m w_j^{(t+1)}\right)\\ &= \frac{1}{m}\sum_{j=1}^m \alpha_j^{(t)} - \frac{\eta}{2m\lambda}\sum_{j=1}^m \alpha_j^{(t)}.\end{aligned}$$

As long as $\sum_{j=1}^m \alpha_j^{(t)} = 0$, we have $\sum_{j=1}^m \alpha_j^{(t+1)} = 0$. □

Proof of Proposition 4. Since $w_j^{(t+1)}$ is the minimizer, we have

$$0 \in \partial f_j(w_j^{(t+1)}) + 2\lambda(w_j^{(t+1)} - w^{(t)}) + \alpha_j^{(t)}.$$

It is equivalent to

$$\partial f_j(w_j^{(t+1)}) \ni -2\lambda(w_j^{(t+1)} - w^{(t)}) - \alpha_j^{(t)}$$

Since f_j is convex, for any $w \in \mathbb{R}^d$,

$$f_j(w) \geq f_j(w_j^{(t+1)}) + \left\langle -2\lambda(w_j^{(t+1)} - w^{(t)}) - \alpha_j^{(t)}, w - w_j^{(t+1)} \right\rangle$$

Then

$$f_j(w_j^{(t+1)}) \leq f_j(w) + 2\lambda \left\langle w_j^{(t+1)} - w^{(t)}, w - w_j^{(t+1)} \right\rangle + \left\langle \alpha_j^{(t)}, w - w_j^{(t+1)} \right\rangle.$$

By (8), it holds that

$$w_j^{(t+1)} = w^{(t+1)} + \frac{\alpha_j^{(t+1)} - \alpha_j^{(t)}}{\eta}.$$

Recall that $\lambda = \frac{\eta}{2}$. Plugging into above, we have

$$\begin{aligned}
f_j(w_j^{(t+1)}) &\leq f_j(w) + \eta \left\langle w^{(t+1)} - w^{(t)} + \frac{\alpha_j^{(t+1)} - \alpha_j^{(t)}}{\eta}, w - w_j^{(t+1)} \right\rangle + \left\langle \alpha_j^{(t)}, w - w_j^{(t+1)} \right\rangle \\
&= f_j(w) + \eta \left\langle w^{(t+1)} - w^{(t)}, w - w_j^{(t+1)} \right\rangle + \left\langle \alpha_j^{(t+1)}, w - w_j^{(t+1)} \right\rangle \\
&= f_j(w) + \eta \left\langle w^{(t+1)} - w^{(t)}, w - w_j^{(t+1)} \right\rangle + \left\langle \alpha_j^{(t+1)}, w - w^{(t+1)} \right\rangle - \frac{1}{\eta} \left\langle \alpha_j^{(t+1)}, \alpha_j^{(t+1)} - \alpha_j^{(t)} \right\rangle.
\end{aligned}$$

Summing over j, it holds that

$$\begin{aligned}
\frac{1}{m}\sum_{j=1}^m f_j(w_j^{(t+1)}) \leq \frac{1}{m}\sum_{j=1}^m f_j(w) &+ \eta \left\langle w^{(t+1)} - w^{(t)}, w - \frac{1}{m}\sum_{j=1}^m w_j^{(t+1)} \right\rangle \\
&+ \left\langle \frac{1}{m}\sum_{j=1}^m \alpha_j^{(t+1)}, w - w^{(t+1)} \right\rangle - \frac{1}{m\eta}\sum_{j=1}^m \left\langle \alpha_j^{(t+1)}, \alpha_j^{(t+1)} - \alpha_j^{(t)} \right\rangle.
\end{aligned}$$

By Lemma 5, we have

$$\frac{1}{m}\sum_{j=1}^m f_j(w_j^{(t+1)}) \leq \frac{1}{m}\sum_{j=1}^m f_j(w) + \eta \left\langle w^{(t+1)} - w^{(t)}, w - w^{(t+1)} \right\rangle - \frac{1}{m\eta}\sum_{j=1}^m \left\langle \alpha_j^{(t+1)}, \alpha_j^{(t+1)} - \alpha_j^{(t)} \right\rangle.$$

With the law of cosines applied, we obtain

$$\begin{aligned}
\frac{1}{m}\sum_{j=1}^m f_j(w_j^{(t+1)}) \leq \frac{1}{m}\sum_{j=1}^m f_j(w) &- \frac{\eta}{2}\left(\left\| w^{(t)} - w^{(t+1)} \right\|^2 + \left\| w - w^{(t+1)} \right\|^2 - \left\| w - w^{(t)} \right\|^2 \right) \\
&- \frac{1}{2m\eta}\left(\left\| \alpha_j^{(t+1)} \right\|^2 + \left\| \alpha_j^{(t+1)} - \alpha_j^{(t)} \right\|^2 - \left\| \alpha_j^{(t)} \right\|^2 \right).
\end{aligned}$$

Summing over $t = 0, \ldots, T-1$, it holds that

$$\frac{1}{mT}\sum_{t=0}^{T-1}\sum_{j=1}^m f_j(w_j^{(t+1)}) \leq \frac{1}{m}\sum_{j=1}^m f_j(w) + \frac{\eta}{2T}\left\| w - w^{(0)} \right\| - \frac{1}{2m\eta T}\sum_{j=1}^m \left(\sum_{t=0}^{T-1} \left\| \alpha_j^{(t+1)} - \alpha_j^{(t)} \right\|^2 - \left\| \alpha_j^{(0)} \right\|^2 \right).$$

By (8), it holds that

$$\frac{1}{2m\eta T}\sum_{j=1}^{m}\sum_{t=0}^{T-1}\left\|\alpha_j^{(t+1)}-\alpha_j^{(t)}\right\|^2=\frac{\eta}{2mT}\sum_{j=1}^{m}\sum_{t=0}^{T-1}\left\|w^{(t+1)}-w_j^{(t+1)}\right\|^2.$$

Therefore, we have

$$\frac{1}{T}\sum_{t=0}^{T-1}\left(\frac{1}{m}\sum_{j=1}^{m}f_j(w_j^{(t+1)})+\frac{\eta}{2m}\sum_{j=1}^{m}\sum_{j=1}^{m}\left\|w^{(t+1)}-w_j^{(t+1)}\right\|^2\right)\leq\frac{1}{m}\sum_{j=1}^{m}f_j(w)+\frac{\eta}{2T}\left\|w-w^{(0)}\right\|^2.$$

As a result, there exists some $t_*\leq T$ satisfying that

$$\frac{1}{m}\sum_{j=1}^{m}f_j(w_j^{(t^*)})+\frac{\eta}{2m}\sum_{j=1}^{m}\sum_{j=1}^{m}\left\|w^{(t^*)}-w_j^{(t^*)}\right\|^2\leq\frac{1}{m}\sum_{j=1}^{m}f_j(w)+\frac{\eta}{2T}\left\|w-w^{(0)}\right\|^2.$$

Choose $\eta=2\sqrt{T}$, and we get

$$\frac{1}{m}\sum_{j=1}^{m}f_j(w_j^{(t^*)})+\frac{\sqrt{T}}{m}\sum_{j=1}^{m}\sum_{j=1}^{m}\left\|w^{(t^*)}-w_j^{(t^*)}\right\|^2\leq\frac{1}{m}\sum_{j=1}^{m}f_j(w)+\frac{1}{\sqrt{T}}\left\|w-w^{(0)}\right\|^2.$$

Thus we get

$$\frac{1}{m}\sum_{j=1}^{m}f_j(w_j^{(t^*)})\leq\frac{1}{m}\sum_{j=1}^{m}f_j(w)+\frac{1}{\sqrt{T}}\left\|w-w^{(0)}\right\|^2$$

$$\frac{1}{m}\sum_{j=1}^{m}\sum_{j=1}^{m}\left\|w^{(t^*)}-w_j^{(t^*)}\right\|^2\leq\frac{1}{m\sqrt{T}}\sum_{j=1}^{m}f_j(w)+\frac{1}{T}\left\|w-w^{(0)}\right\|^2.$$

□