Communication Limits of Distributed Algorithms for Statistical Learning

Yuxin Su

Department of Computer Science and Engineering
The Chinese University of Hong Kong

January 28, 2015
Outline

1 Motivation
   - Distributed Machine Learning
   - Another Perspective

2 General Information-theoretic Framework
   - General Framework
   - Result
Outline

1 Motivation
   - Distributed Machine Learning
   - Another Perspective

2 General Information-theoretic Framework
   - General Framework
   - Result
The volume of data is quite large

Model is big enough like huge kernel or big latent matrix

How to achieve fast response?
How to process big data

- The volume of data is quite large
- Model is big enough like huge kernel or big latent matrix
- How to achieve fast response?
How to process big data

- The volume of data is quite large
- Model is big enough like huge kernel or big latent matrix
- How to achieve fast response?
Current solution

Machine 2

Data or Model Parameter

Communication is slow

Machine 1

Data or Model Parameter

Machine 3

Data or Model Parameter

Machine 4

Data or Model Parameter
How to handle the bottleneck of network

Wait for communication
- MapReduce
- Bulk Synchronous Parallel
- GraphLab

Trade-off between communication and performance
- Petuum
- Many global approximation methods from local sub-solution
  - Local computation -> reduce to global result
How to handle the bottleneck of network

Wait for communication
- MapReduce
- Bulk Synchronous Parallel
- GraphLab

Trade-off between communication and performance
- Petuum
- Many global approximation methods from local sub-solution
  - Local computation -> reduce to global result
Outline

1. Motivation
   - Distributed Machine Learning
   - Another Perspective

2. General Information-theoretic Framework
   - General Framework
   - Result
Information constrains in learning

Memory constrain
- kernel methods

Sequential access constrain
- Online learning

Communication Constrain
- Distributed machine learning

Partial access to the underlying data
- Matrix completion
- Multi-armed bandit problem
Communication constrain vs Partial access

Machine 2
Data or Model Parameter

Machine 3
Data or Model Parameter

Machine 4
Data or Model Parameter

Machine 1
Data or Model Parameter

Ready?

Ready?

Ready?
Outline

1 Motivation
   - Distributed Machine Learning
   - Another Perspective

2 General Information-theoretic Framework
   - General Framework
   - Result
Motivation

- How the learning algorithms interact with the training data
- How these constrains impact the performance
Information-constrained protocols

\((b, n, m)\) protocol

Given access to a sequence of \(m \times n\) i.i.d instance in \(\mathbb{R}^d\), an algorithm is a \((b, n, m)\) protocol if it has the following form:

- For \(t = 1, \ldots, m\)
  - Let \(X^t\) be a batch of \(n\) i.i.d instances
  - Compute message \(W^t = f_t(X^t, W^1, \ldots, W^{t-1})\)
- Return \(W = f(W^1, \ldots, W^m)\)

\(W^t\) are constrained to be only \(b\) bits.

In distributed setting

There are \(m\) machines, each machine will received a set of messages in serial order.
Hide-and-seek Problem

It is similar to “exploration and exploitation” strategy in multi-armed bandit problem.

**Definition**

Consider the set of product distributions $\{\Pr_j(\cdot)\}_{j=1}^d$ over $\{-1,1\}^d$ defined via $\mathbb{E}_{x \sim \Pr_j(\cdot)}[x_i] = 2\rho \mathbf{1}_{i=j}$ for all coordinates $i = 1, \ldots, d$. Given an i.i.d sample of $m \times n$ instances generated from $\Pr_j(\cdot)$, where $j$ is unknown, detect $j$. 

Yuxin Su (CSE@CUHK)
Theorem: without information constrain

Consider the hide-and-seek problem. Given $m \times n$ samples, if $\tilde{J}$ is the coordinate with the highest empirical average, then:

$$\Pr_j(\tilde{J} = j) \geq 1 - 2d \exp(-\frac{1}{2}mnp^2)$$
Consider the hide-and-seek problem on $d > 1$ coordinates, with some bias $\rho \leq 1/4$ and sample size $m$. The for any estimate $\tilde{J}$ of the biased coordinate returned by an $(b,1,m)$ protocol, there exists some coordinate $j$ such that:

$$\Pr_j(\tilde{J} = j) \leq \frac{3}{d} + 21 \sqrt{m \frac{\rho^2 b}{d}}$$

For any algorithm based on $(b,1,m)$ protocol, it requires sample size $m$ to reliably detect some $j$.

$$m \geq \Omega\left(\frac{d}{b\rho^2}\right)$$
Theorem: \((b, n, m)\) protocol

**Theorem**

Consider the hide-and-seek problem on \(d > 1\) coordinates, with some bias \(\rho \leq 1/4n\) and sample size \(m \times n\). Then for any estimate \(\tilde{J}\) of the biased coordinate returned by any \((b, n, m)\) protocol, there exists some coordinate \(j\) such that:

\[
Pr_j(\tilde{J} = j) \leq \frac{3}{d} + 5 \sqrt{mn \min\left\{\frac{10\rho b}{d}, \rho^2\right\}}
\]

**Implication**

For any algorithm based on \((b, n, m)\) protocol, it requires sample size at least \(\Omega(\max\left\{\frac{(d/b)}{\rho}, \frac{1}{\rho^2}\right\})\) to reliably detect some \(j\).
Outline

1 Motivation
   - Distributed Machine Learning
   - Another Perspective

2 General Information-theoretic Framework
   - General Framework
   - Result
Lower bound

Generic regret lower bound for partial access

\[ \Omega(\sqrt{\frac{d}{b}}T) \]

- \( d \) is the dimension of loss or reward vector.
- \( b \) is the dimension of extracted vector from received message.
- \( T \) is the number of round.

Trade-off between communication and sample complexity

For serial protocol on i.i.d data, the lower bound of communication is \( \tilde{\Omega}(d^2) \) per machine.

- \( d \) is the dimension of problem.
Open Question

Whether the results for distributed algorithms can be extended to more interactive protocols, where the different machines can communicate over several rounds.