Parallel Stochastic Successive Convex Approximation Method for Large-Scale Dictionary Learning

Alec Koppel, Aryan Mokhtari, and Alejandro Ribeiro

Contact E-mail: alec.koppel@gmail.com, aryan.mokhtari@mit.edu, arbero@seas.upenn.edu

ICASSP 2018: Sparse signal processing/dictionary learning Apr. 19, 2018

Large-Scale Parameter Estimation

- Supervised learning \( \xrightarrow{\text{vector}} \mathbf{x} \subset \mathbb{R}^n \) minimizing expected risk \( F(\theta) \)
- \( \mathbf{x}_0 = n \) - non-convex loss associated to \( n \)-th sample point.

Examples:
- Regression with non-linear transformation of feature space
- Objective defined by training neural network (deep learning)
- \( (\mathbf{Q}x) \) convex non-differentiable, block separable e.g., \( (\mathbf{Q}) \) sparsity penalty

Example problems:
- Support vector machines,
- Logistic regression,
- Deep learning
- Both the feature dimension \( p \) and sample size \( N \) are huge-scale

Feature and Sample Parallel Learning

- Sample size \( N \to \infty \) - stochastic gradient
- Workhorse for convex setting where \( n < N \), can't handle \( n \geq O(N) \)
- Key Novelties: operate on subsets of both features and samples
- Main idea: partition classifier into \( B \) distinct blocks of size \( p_0 \)
- \( \Rightarrow \) Associate to block \( \mathbf{b} \), \( \mathbf{x} \) with \( \mathbf{x} \) distinct sub-vector of features \( \mathbf{X}_j^b \)

- Rather than parallel stochastic gradient, SGD on only some blocks
- \( J < B \) processors work in parallel by randomly choosing blocks
- \( \Rightarrow \) Processor \( \mathbf{P} \) updates block \( \mathbf{x} = \mathbf{X}_j^b \), using stochastic subset of data.

- Merge advantages of stochastic coord. descent and stochastic approx.
- Prior work implemented this idea for convex stochastic programming
- This work: extension to non-convex composite problems

Successive Convex Approximation

- Broad idea: replace non-convex part of objective with convex approximation
- Agree on convex function at the gradient

- To develop doubly stochastic, successively approx.
- Many definitions required

- Define block-wise stochastic gradient

- Stochastic approx. of grad. of average cost w.r.t. block variables \( \mathbf{x} \)
- Evaluate at a distinct data mini-batch \( \mathbf{x}_0 \) of size \( L \)

- Define block-wise instantaneous convex surrogate function \( (\mathbf{x}_0, \mathbf{x}^b, \mathbf{x}^f) \)
- Differentiable and convex with respect to the block

- Example surrogate: linearization of instantaneous cost \( (\mathbf{x}_0, \mathbf{x}^b, \mathbf{x}^f) \)

- Similarly, define the mini-batch sample surrogate as

- Define mini-batch surrogate function w.r.t. data subset \( \mathbf{x}_0 \)

Convergence Result

- Lemma: The DSSC sequence \( \{x^t\} \) with alternating step-size parameters

- Denote the coding coefficients of \( \mathbf{x}^t \) as \( D^t \) as \( t \in \mathbb{N}^+ \)

- Parallel Stochastic Successive Convex Approximation

- Processor \( P_i \) chooses \( \mathbf{x}^t \) \( \in \{1, \ldots, T\} \) uniform randomly, data subset \( \mathbf{x}_0^t \)
- Executes “proximal” update using surrogate function \( f(\mathbf{x}_0^t, \mathbf{x}_0^b, \mathbf{x}_0^f) \)

- \( \mathbf{x}^t \) behaves similarly to inverse of step-size of standard SGD method
- \( \Rightarrow \) time-averaged stochastic gradient methods w.r.t.

- \( \Rightarrow \mathbf{J} \) is gradient averaging momentum parameter

- Next block-variable \( \Rightarrow \) convex comb. of \( \mathbf{x}^t \) and “proximal” \( \mathbf{x}^t \)

- \( \Rightarrow \gamma \) is additional momentum step-size parameter

- Update is done for each processor \( \mathbf{P} \) in parallel

Parallel Architecture Implementation

- Iteration \( t \), processor \( \mathbf{P} \) picks

- Block \( D \) \( \in \mathcal{P} \) at random
- Sample subset \( \mathbf{x}_0^t \)

- Computes proximal step \( \mathbf{x}^t \)

- with surrogate \( (\mathbf{x}_0^t, \mathbf{x}_0^b, \mathbf{x}_0^f) \), recursively avg. grad. \( \mathbf{d}^t \)

- Stabilized proximal update \( \mathbf{x}^t \)

- Indepedent of processors \( \mathbf{P} \)

- Allows for processors to operate on distinct data subset at same time
- Theoretical speedup w.r.t. comparison to number of processors \( J \)
- Allow for passing through training set \( J \) times faster
- Amenable to implementation on GPU for HPC system

Technical Conditions

- The block-wise feasible sets \( f \), are convex and compact
- Common condition to guarantee optimization sequence is bounded

- Block stochastic gradients have bounded stochastic approx. error

- Standard stochastic approximation condition

- Typically bounded by constant plus a term proportional to \( \|x^t\|^2 \)

- The sequences \( \{\mathbf{x}^t\} \) \( \neq \emptyset \) are chosen such that

- \( \mathbf{x}^t \) converges to \( \mathbf{x} = \mathbf{argmin}_{\mathbf{x}} f(\mathbf{x}) \)

- \( \Rightarrow \) gradient averaging scheme converges

- Attenuating step-size rules for both

- Convergence rate \( \mathbf{x} \)

- \( \Rightarrow \gamma \) is a constant in limit \( \mathbf{d}^t \) true block gradient

Results for Linear Estimation

- Fix \( \mathbf{W} \) \( n \times p \) and \( n \gg p \), i.i.d.

- Additive noise \( \mathbf{y} = \mathbf{Wx} + \mathbf{n} \)

- Solve \( \mathbf{x} = \mathbf{W}^{-1}\mathbf{y} \)

- The noise variance \( \sigma^2 \)

- Common condition to guarantee optimization sequence is bounded

- Block stochastic gradients have bounded stochastic approx. error

- Standard stochastic approximation condition

- Typically bounded by constant plus a term proportional to \( \|x^t\|^2 \)

- The sequences \( \{\mathbf{x}^t\} \) \( \neq \emptyset \) are chosen such that

- \( \mathbf{x}^t \) converges to \( \mathbf{x} = \mathbf{argmin}_{\mathbf{x}} f(\mathbf{x}) \)

- \( \Rightarrow \) gradient averaging scheme converges

- Attenuating step-size rules for both

- Convergence rate \( \mathbf{x} \)

- \( \Rightarrow \gamma \) is a constant in limit \( \mathbf{d}^t \) true block gradient

Conclusions

- Classical stochastic approximation methods

- Can't handle \( p \geq O(N) \)

- DSSC breaks bottlenecks in \( p \). First time for non-convex problems

- Operates on random subsets of samples and features

- Can be implemented on a parallel computing architecture

- No coordination among distinct computing nodes required

- Established convergence of DSSC

- Via use of regularized attenuation of convex surrogate gradients

- Under standard technical conditions

- Benefit of both stochastic coordinate descent and stochastic approx.

- Proposing results only on numerical experiments

- Demonstrated on unsupervised learning problem: dictionary learning

Dictionary Learning

- Collection of signals \( \{x_k\} \subset \mathbb{R}^p \) for \( k = 1, \ldots, N \)

- Represent signal \( x_k \) as a combination of \( k \) basis elements \( \{d_k\} \)

- Basis elements unknown, learn from data

- Group basis elements into dictionary matrix \( D = [d_1, \ldots, d_k] \subset \mathbb{R}^{P \times k} \)

- Dictionary learning problem

- Find coding \( \mathbf{x} \) and dictionary \( D \)

- Use dictionary representations \( \mathbf{x} \) for all \( m \) at the same time

- Eliminate the scale ambiguity of the bilinear term \( \mathbf{x}^T \mathbf{D} \)

- Reduce a large subspace from which sparse codes are learned

- Bilinear term makes this formulation non-convex.

Results on Robot Image Feed

- We want to learn dictionary online

- Via image stream

- Consisting of image patches

- Via concrete, grass paths

- Dict. atoms \( \rightarrow \) terrain features

- Learned dict. from robot images

- Images collected on Husky robot

- Geographical landmark

- Various types of objects

- Faster learning with less features

- Validates trend in linear reg. prob.

- Problem \( \mathbf{W} \) empirical data

https://koppel.bitbucket.io/