Labs
Optimization for Machine Learning
Spring 2024

Saarland University
CISPA Helmholtz Center for Information Security
Sebastian Stich
TAs: Yuan Gao & Xiaowen Jiang
https://cms.cispa.saarland/optml24/

Problem Set 8, June 11, 2024 (Local SGD - continue)

1 Scaffold

Consider a distributed optimization problem of the form $f(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x})$.

1.1 Unbiased Local Steps

Let $\mathbf{g}^i(\mathbf{x})$ denote a unbiased gradient estimator for the local objective function $f_i(\mathbf{x})$, i.e. $\mathbb{E}\mathbf{g}^i(\mathbf{x}) = \nabla f_i(\mathbf{x})$, $\forall \mathbf{x} \in \mathbb{R}^d$.

Let $\mathbf{y}_k^i$ denote a local iterate of Scaffold on node $i \in [n]$ at (local) iteration $k \in \{0, \dots, K-1\}$. Recall that the control variates are defined as $\mathbf{c}^i = \mathbf{g}^i(\mathbf{y}_0)$ and $\mathbf{c} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{c}^i$ ($\mathbf{y}_0^i = \mathbf{y}_0^j$ for all $i, j \in [n]$).

Prove that the update direction in Scaffold,

$$\mathbf{v}_k^i = \mathbf{g}^i(\mathbf{y}_k^i) - \mathbf{c}^i + \mathbf{c}\,,$$

is an unbiased gradient estimator, i.e. $\mathbb{E}\mathbf{v}_k^i = \nabla f_i(\mathbf{y}_k^i)$.

1.2 Hessian Similarity I

Let $f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x})$, with each f_i an L-smooth and twice differentiated function. δ-Hessian similarity was defined in the lecture as:

$$\left\|\nabla^2 f_i(\mathbf{x}) - \nabla^2 f(\mathbf{x})\right\| \leq \delta \qquad \forall i \in [n], \forall \mathbf{x} \in \mathbb{R}^d\,.$$

Prove that Hessian similarity always holds for $\delta \leq 2L$.

1.3 Hessian Similarity II

Prove that the following holds for any two (twice differentiable) functions $f_i(\mathbf{x})$, $f(\mathbf{x})$ satisfying δ-Hessian similarity:

$$\|\nabla f_i(\mathbf{y}) - \nabla f_i(\mathbf{x}) + \nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \leq \delta^2 \|\mathbf{y} - \mathbf{x}\|^2\,, \qquad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d\,.$$

2 Practical Implementation of FedAvg

Follow the Python notebook provided here:

https://colab.research.google.com/github/mlolab/optML-course/blob/main/labs/ex10/ex10.ipynb