EP3260: Fundamentals of Machine Learning over Networks
Computer Assignment 3, Due Date: March 27, 2020
Group 4: H. Hellström, M. Tavana, F. Vannella, and M. Zeng

Computer Assignment 3. *(Training a neural network)*
Consider optimization problem

$$\underset{W_1,W_2,\boldsymbol{w}_3}{\text{minimize}} \ \frac{1}{N}\sum_{i\in[N]} \left\|\boldsymbol{w}_3 s\left(W_2 s\left(W_1 \boldsymbol{x}_i\right)\right) - \boldsymbol{y}_i\right\|_2^2,$$

where $s(x) = 1/(1+\exp(-x))$. You may add your choice of regularizer. Using the "Individual household electric power consumption" and "Greenhouse Gas Observing Network" datasets, address the following questions:

(a) Try to solve this optimization task with proper choices of size of decision variables (matrix W_1, matrix W_2, and vector $\boldsymbol{w}_3$) using GD, perturbed GD, SGD, SVRG, and block coordinate descent. For the SGD method, you may use the mini-batch version.

(b) Compare these solvers in terms complexity of hyper-parameter tunning, convergence time, convergence rate (in terms of # outer-loop iterations), and memory requirement

Proof. We focus on a balanced binary classification task for both "Individual household electric power consumption" and "Greenhouse Gas Observing Network" datasets. We transform these regression datasets in a balanced binary classification task by dividing the label variable $y \in \mathcal{Y}$ in two classes, i.e. $\mathcal{Y} = \{-1, 1\}$. We select the median value γ of the continuous label variable and we execute hard thresholding value as

$$\begin{cases} y_i \leftarrow 1 \text{ if } y_i > \gamma \\ y_i \leftarrow -1 \text{ if } y_i < \gamma \end{cases} \tag{1}$$

Subsequently we implement the 5 optimization methods: GD, SGD, PGD, SVGR, and BCD and we compare them. The decision variables are chosen such that $|W_1| = d \times w_{size}, |W_2| = w_{size} \times w_{size}$, $w_3 = w_{size} \times 1$, where d is the number of input features and w_{size} is an hyper-parameter determined empirically.

Individual household electric power consumption results

We focus first on the Individual household electric power consumption dataaset. For this dataset we execute experiments by sweeping over the hyper-parameter values $w_{size} = [3, 5, 7, 9]$ and Learning rate $\gamma = 1 : 0.5 : 15$ for all optimization methods, by empirically checking that these values are effective over all optimization methods, whose optimal choice is reported in Table . The Loss values at convergence and execution Time are also shown in the Table.

Mehtod	Loss	Time	w_{size}	Learning Rate
GD	0.524	75.371	15	3
SGD	0.525	2.601	15	3
PGD	0.536	73.718	7.5	7
BCD	0.557	65.753	6.5	5
SVGR	0.544	74.956	1	9

Table 1: Artificial Neural Network Time and Loss performance for the household electric power consumption Dataset.

In Figure 2 the Loss and Time performance are reported varying training epoch.

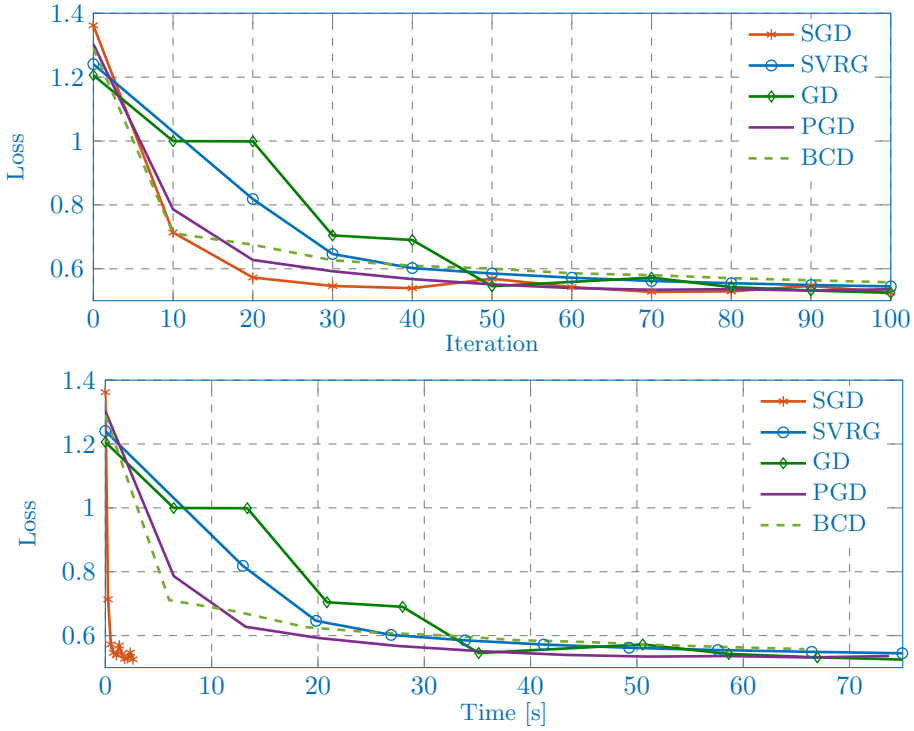

Figure 1: Artificial Neural Network Time and Loss performance for the power consumption Dataset.

Greenhouse Gas Observing Network

In the second batch of experiments we focus on the IGreenhouse Gas Observing Network dataaset. As beforem, Fo we execute experiments by sweeping over the hyper-parameter values $w_{size} = [3, 5, 7, 9]$ and Learning rate $\gamma = 0.1 : 0.5 : 5$ for all optimization methods, by empirically checking that these values are effective over all optimization methods, whose optimal choice is reported in Table 2. The Loss values at convergence and execution Time are also shown in the Table.

Mehtod	Loss	Time	w_{size}	Learning Rate
GD	0.957	76.866	7	0.9
SGD	0.913	1.799	9	1.9
PGD	0.895	73.387	9	0.5
BCD	0.951	73.735	6.5	5
SVGR	0.930	76.858	1	9

Table 2: Artificial Neural Network Time and Loss performance for the Greenhouse Gas Observing Network Dataset.

In Figure 2 the Loss and Time performance are reported varying training epoch.

Discussion

Based on the results of these two experiments we can draw some conclusion. It surely emerges that the faster solver is the SGD. This solver also appears to achieve the best results in terms of Loss (i.e. lowest loss among the other solvers). The SGD also allow to fine-tune the parameter faster since it requires less time for each experiment. All the other methods have comparable performance in terms of convergence speed and convergence rate and overall performs worse than SGD especially for what concerns convergence speed wrt SGD. □

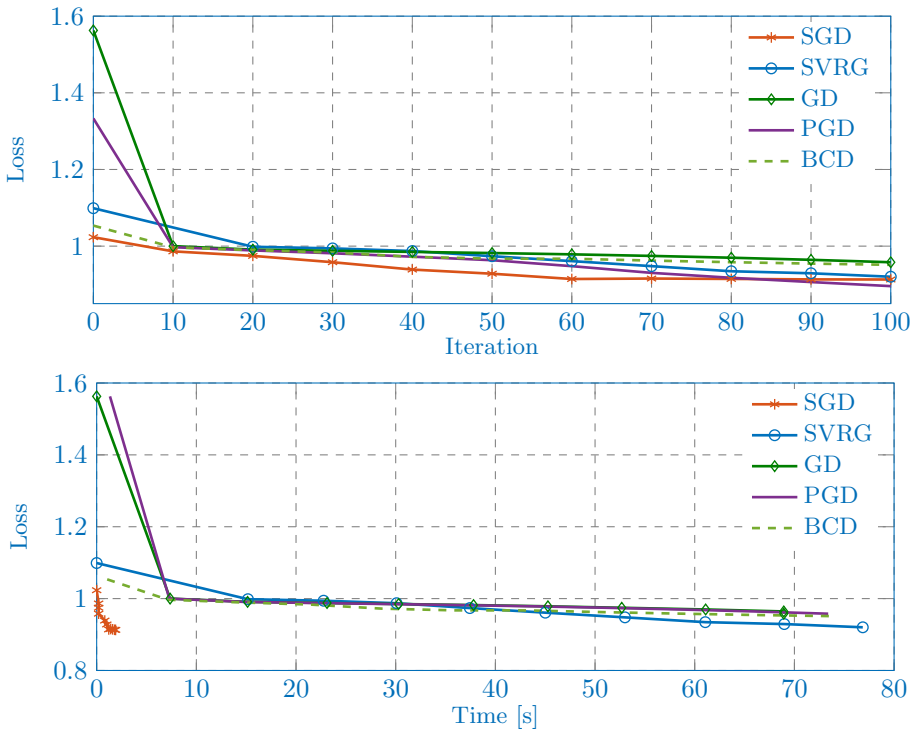

Figure 2: Artificial Neural Network Time and Loss performance for the power consumption Dataset.