EP3260: Machine Learning Over Networks Peer-review of CA3 of group 5

Stefanos Antaris[*1], Amaru Cuba Gyllensten[†1,2], Martin Isaksson[‡1,2,3], Sarit Khirirat[§1], and Klas Segeljakt[¶1,2]

[1]*KTH Royal Institute of Technology*
[2]*RISE AI*
[3]*Ericsson Research*

February, 2019

Contents

1 Computer assignment

1.1 General comments

- Good use of Python notebooks to make the solution and results easy to follow.

[*]antaris@kth.se
[†]amaru.cuba.gyllensten@ri.se
[‡]martisak@kth.se
[§]sarit@kth.se
[¶]klasseg@kth.se

- Good that you not only solved the problem but also explored different implementations.
- Good that you separated the problem of cleaning and training into two notebooks.

1.2 1) Try to solve this optimization task with proper choices of size of decision variables (matrix W_1, matrix W_2, and vector w_3) using GD, perturbed GD, SGD, SVRG, and block coordinate descent. For the SGD method, you may use the mini-batch version.

- GD, SGD, PGD, SVRG and BCD are all implemented correctly.
- Good reuse of code for SGD, PGD, and BCD.
- Is there a motivation behind the choice of hyperparameters? For example, why does SVRG have different layers than the rest?

1.3 2) Compare these solvers in terms complexity of hyper-parameter tunning, convergence time, convergence rate, and memory requirement

- The solvers are all compared w.r.t the properties.
- In general, there could be a bit more explanation, e.g., why does BCD have a low memory requirement?
- Why does PGD require as expensive memory management as GD? In the code, the memory requirement of PGD appears to be equivalent to that of SGD.