Towards Geo-Distributed Machine Learning
Ignacio Cano, Markus Weimer, Dhruv Mahajan, Carlo Curino, Giovanni Matteo Fumarola
icano@cs.washington.edu, {mweimer,dhromium,ccurino,giuima}@microsoft.com

Motivation
- Data is generated and stored all around the world.
- ML applications require a global view of such data to achieve the best results.

Setup

Current Solutions (Centralized)
1. Copy the RAW data into a single data center.
2. Run the ML algorithm “locally”.
   - Intuitive as ML is:
     - Iterative.
     - Communication intensive.
   - Problems:
     - X-DC transfers are costly.
     - Data sovereignty issues.
     - Security threats.
     - X-DC high latency.

Our Approach
1. Leave the data in place.
2. Train in a geo-distributed fashion.

Key Challenges:
- Algorithm: reduces X-DC communication of centralized and achieves same accuracy.
- System: realizes benefits of algorithm, and make it robust to network failure.

1. Algorithm: Mahajan et al., 2015
   Objective Function
   \[ f(w) = \frac{\lambda}{2}||w||^2 + \sum_{p=1}^{P} L_p(w) \]
   Local approximations
   \[ f_p(w) = \frac{\lambda}{2}||w||^2 + \hat{L}_p(w) \]
   Approximation of the losses of the other DCs
   \[ \sum_{q \neq p} L_q(w) \]

2. System: Apache REEF application on top of a federated YARN cluster.

Preliminary Results
- Splice dataset for Human Splice Site Recognition.
  - 50M examples, 47K features, 200GB on disk.
- Simulation of 2, 4, 8 and 16 DCs in large centralized cluster.
- L2 regularized Logistic Regression with TRON.

Conclusions & Future Work
- Introduced a new kind of learning problems that need to deal with geo-distributed datasets (GDML).
- Implemented an initial system for X-DC training.
- Empirical results show orders of magnitude improvements in terms of X-DC transfers while achieving same accuracy.
- Next: Fault-Tolerance, Latency, Privacy, Scheduling...

1 http://reef.apache.org/