Learning in a Distributed and Heterogeneous Environment

Martin Jaggi
EPFL Machine Learning and Optimization Laboratory
mlo.epfl.ch
Machine Learning Methods to Analyze Large-Scale Data
Machine Learning Systems
Machine Learning Systems

machine 1

GPU 1a

FPGA 1b

machine 2

GPU 2a

FPGA 2b

machine 3
The Cost of Communication

- Reading $v$ from memory (RAM)
  $100 \text{ ns}$

- Sending $v$ to another machine
  $500’000 \text{ ns}$

- Typical Map-Reduce iteration
  $10’000’000’000 \text{ ns}$

$\mathbf{v} \in \mathbb{R}^{100}$

Challenge
Challenge

The Cost of Communication

![Graph showing suboptimality over time for different implementations.](image-url)
Usability
Good distributed and parallel code is hard

- no reusability of good single machine algorithms & code
- no portability: model-specific and system-specific code
Distributed

What if the data does not fit onto one device anymore?
Problem class

\[
\min_{\alpha \in \mathbb{R}^n} \quad f(A\alpha) + g(\alpha)
\]
One-Shot Averaging Does Not Work

\[ \begin{align*}
\mathbf{w}^{(1)} &= \mathbf{w}^{(1)*} \\
\mathbf{w}^{(5)} &= \mathbf{w}^{(5)*} \\
\mathbf{w} &= \frac{1}{K} \sum_k \mathbf{w}^{(k)}
\end{align*} \]
Optimization: Primal-Dual Context

\[
\min_{\alpha \in \mathbb{R}^n} \left[ O_A(\alpha) := f(A\alpha) + g(\alpha) \right] + A_{loc} \Delta \alpha[k] + w
\]

primal Lasso
dual L2-reg SVM/Log-Regr
primal L1-reg SVM/Log-Reg
repeat

\[ w := w + \frac{1}{K} \sum_k \Delta w^{(k)} \]
Distributed Experiments

L1-Regularized Linear Regression

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Training</th>
<th>Features</th>
<th>Sparsity</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>2,396,130</td>
<td>3,231,961</td>
<td>3.5e-3%</td>
</tr>
<tr>
<td>epsilon</td>
<td>400,000</td>
<td>2,000</td>
<td>100%</td>
</tr>
<tr>
<td>kddb</td>
<td>19,264,097</td>
<td>29,890,095</td>
<td>9.8e-5%</td>
</tr>
<tr>
<td>webspam</td>
<td>350,000</td>
<td>16,609,143</td>
<td>0.02%</td>
</tr>
</tbody>
</table>

NIPS 2014, ICML 2015, arxiv.org/abs/1611.02189

- part of TensorFlow core (L2)
- custom code (L1), TF, spark, C
Summary

- **adaptivity** to the communication cost
- **re-usability** of good existing solvers
- **accuracy** certificates

Next Steps

- **second-order** and **trust-region** version (local Hessian)
- **adaptivity** to the degree of separability (coming soon)
- generalization to **deep learning, SGD**
- **benchmarking & code**
Leveraging Heterogenous Systems

Compute & Memory Hierarchy: Which data to put in which device?
Leveraging Heterogenous Systems

duality gap as selection criterion

Unit A

30GB

Unit B

8GB

adaptive importance sampling

AISTATS 2017, 2018
NIPS 2017a,b
Experiments

RAM ⇄ GPU, 30GB dataset

Lasso

SVM
Open Research

- **limited precision operations** for efficiency of communication and computation
- **asynchronous** and **fault tolerant** algorithms
- **multi-level approach** on heterogeneous systems
- more **re-usable** algorithmic building blocks
  - for more systems and problems
Thanks!

mlo.epfl.ch

Celestine Dünner, Virginia Smith, Simone Forte, Thomas Parnell, Chenxin Ma, Martin Takac, Dmytro Perekrestenko, Volkan Cevher, Michael I. Jordan, Thomas Hofmann, Sebastian Stich, Anant Raj