GPU-Accelerated Primal Learning for Extremely Fast Large-Scale Classification

John Halloran and David Rocke
UC Davis

jthalloran@ucdavis.edu
@convexDad
jthalloran.bitbucket.io

Motivation

GPUs have become indispensable compute tools for fast deep learning. However, GPU speedups for many of the fastest ML algorithms are nonexistent. As stated in the scikit-learn documentation:

> "Outside of neural networks, GPUs don't play a large role in machine learning today, and much larger gains in speed can often be achieved by a careful choice of algorithms."

Contrary to this common conception, we show that GPUs effectively speed up extremely intricate, fast machine learning algorithms.

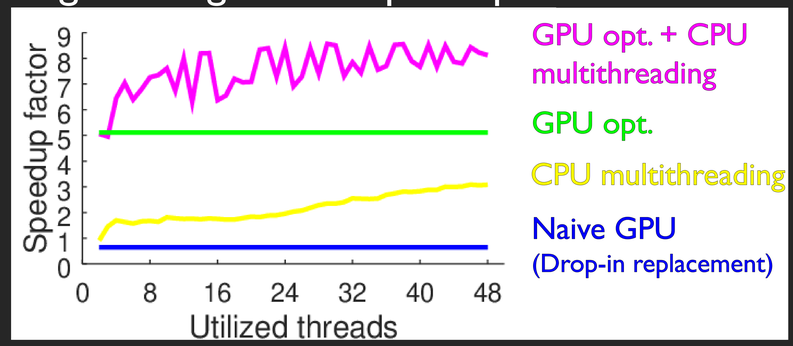
GPU-Optimization Principles

Fast (intricate) ML algorithms contain many sequential dependencies between CPU and GPU variables, causing latency. Steps to alleviate this problem:

1. Offload as much dependent compute in a sequence to the GPU.
2. Calculate dependent compute early and (async.) transfer ASAP.
3. Conceal transfer latency using independent CPU compute.
4. Sync variable transfers as late as possible.

Using careful GPU-optimization principles, even CPU-centric ML algorithms (e.g., those in scikit-learn/LIBLINEAR) can enjoy huge speedups.

Logistic Regression Speedups in LIBLINEAR

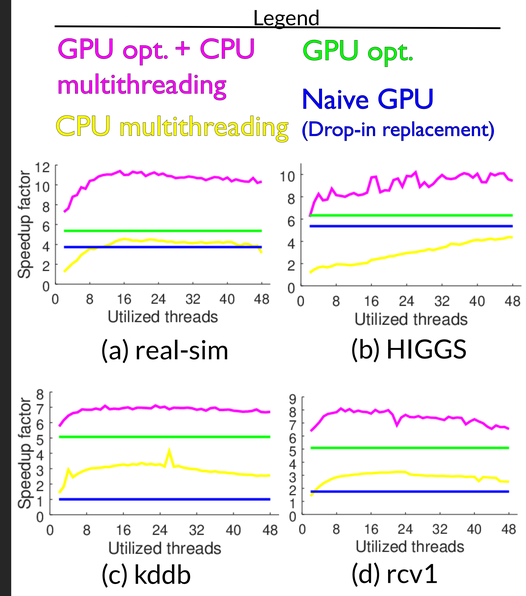


SUSY dataset (5M instances)

Full paper

Faster Logistic Regression in LIBLINEAR

Mix GPU and CPU for speed

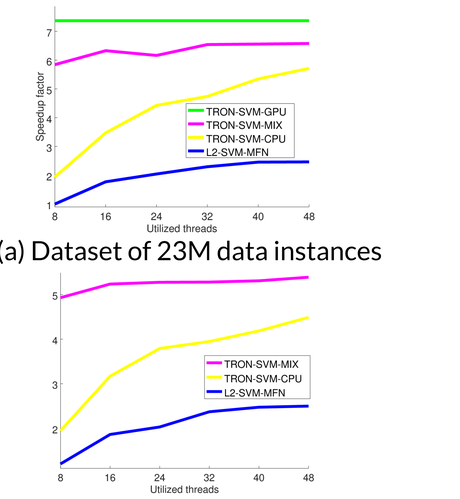


(a) real-sim (b) HIGGS

(c) kddb (d) rcv1

Faster SVM Learning for Massive-Scale Proteomics

Mix GPU and CPU to reduce GPU memory-use.



(a) Dataset of 23M data instances

(b) Massive dataset of 215M data instances, too large for GPU opt. solver