Hybrid Parallelization Strategies for Large-Scale Machine Learning in SystemML

Matthias Boehm, Shirish Tatikonda, Berthold Reinwald, Prithviraj Sen, Yuanyuan Tian, Douglas R. Burdick, Shivakumar Vaithyanathan

IBM Research – Almaden

Background

Motivation
- Problem
  - Analyzing big data
  - Advanced analytics / ML
- SystemML Approach
  - Declarative ML on top of MapReduce
  - Flexibility, optimization, data independence
  - Primarily data parallelism
- Challenges (related via memory constraints)
  - #1: MR vs in-memory computations
  - #2: No support for task parallelism
  - Major challenge: efficiency and scalability for variety of use case

SystemML Architecture

Runtime Strategies

Task Partitioning
- Fixed-size schemes: naïve, static, fixed
- Self-scheduling: factoring

Task Execution
- Local
- Remote
- Hybrid

Result Aggregation
- w/ and w/o compare
- Local in-memory/local file/remote MR

Runtime Optimizations
- Data partitioning
- Data locality

Experiments (as of 07/2013)

Bivariate Statistics

SystemML 07/2013
- R 2.15.1 64bit (doMC: 1x8, doSNOW: 5x8)
- Spark 0.8.0 (5x16 workers, 5x16GB memory)

Experimental Setting
- Physical Cluster: 5 nodes, each 2x4 (16HTW), 64GB RAM, 1.5TB storage, 1GbE, SLES 11 64bit
- Hadoop Cluster: IBM Hadoop 1.1.1, IBM JDK 1.6.0 64bit, map/reduce capacity: 80/40, HDFS block size 128MB, JVM size master/map/reduce: 1GB, ratio 0.7

Linear Regression – Feature Subsampling

Logistic Regression – Parameter Search