Model Calibration for Terascale Analytics

Florin Rusu and Chengjie Qin
University of California, Merced

May 19, 2015

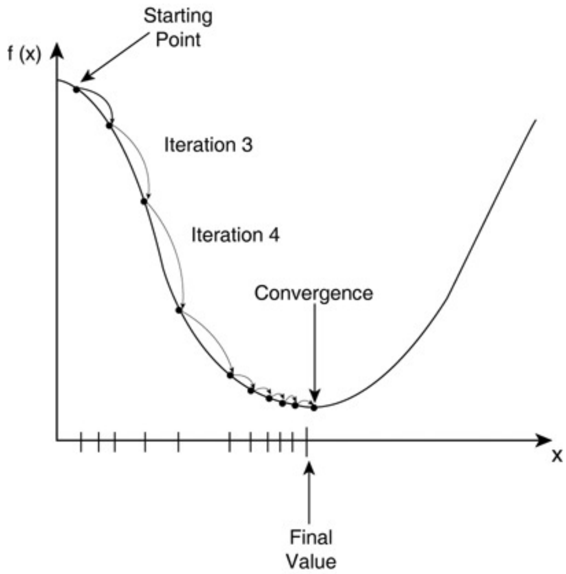
Terascale Analytics

- Massive amounts of example data, e.g., 10 billion
- Highly-dimensional models, e.g., 50 to 600 million

Analytics task	Objective function
Logistic Regression (LR)	$\sum_{(x_i,y_i)\in\text{data}} \log\left(1+e^{-y_i w^T x_i}\right)+\mu\|\overrightarrow{w}\|_1$
Classification (Support Vector Machines - SVM)	$\sum_{(x_i,y_i)\in\text{data}} \left(1-y_i w^T x_i\right)+\mu\|\overrightarrow{w}\|_1$
Recommendation (Low-Rank Matrix Factorization - LMF)	$\sum_{(i,j)\in\Omega} \left(L_i^T R_j - M_{ij}\right)^2+\mu\|L,R\|_F^2$
Labeling (Conditional Random Fields - CRF)	$\sum_{(x_i,y_i)\in\text{data}} \left[\sum_j w_j F_j\left(y_i,x_i\right)-\log Z\left(x_i\right)\right]$

Table 1: [Feng, Kumar, Recht, and Re: *Towards a Unified Architecture for in-RDBMS Analytics*, SIGMOD 2012]

Gradient Descent Optimization



http://www.yaldex.com/game-development/1592730043_ch18lev1sec4.html

$$\min_{w \in \mathbb{R}^d} \sum_{(x_i, y_i) \in \text{data}} f(w, x_i, y_i)$$

$$w^{(k+1)} = w^{(k)} - \alpha_k \nabla f\left(w^{(k)}\right)$$

∇f is the gradient

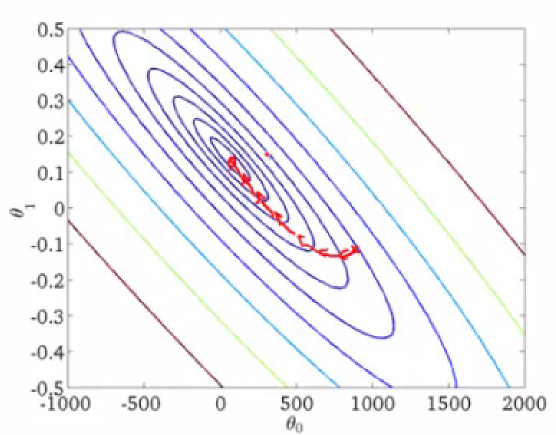
α_k is step size or learning rate

$w^{(0)}$ is the starting point (random)

- Convergence to minimum guaranteed for convex objective function

Batch and Stochastic Gradient Descent

Batch Gradient Descent (BGD)

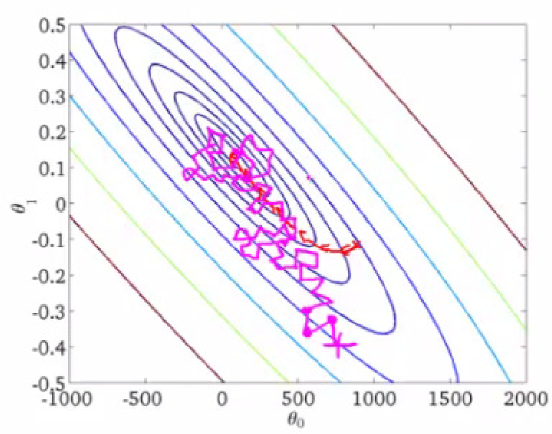


http://www.holehouse.org/mlclass/17_Large_Scale_Machine_Learning.html

$$w^{(k+1)} = w^{(k)} - \alpha_k \nabla f\left(w^{(k)}\right)$$

- Exact gradient computation
- Single step for one iteration
- Faster convergence close to minimum

Stochastic Gradient Descent (SGD)

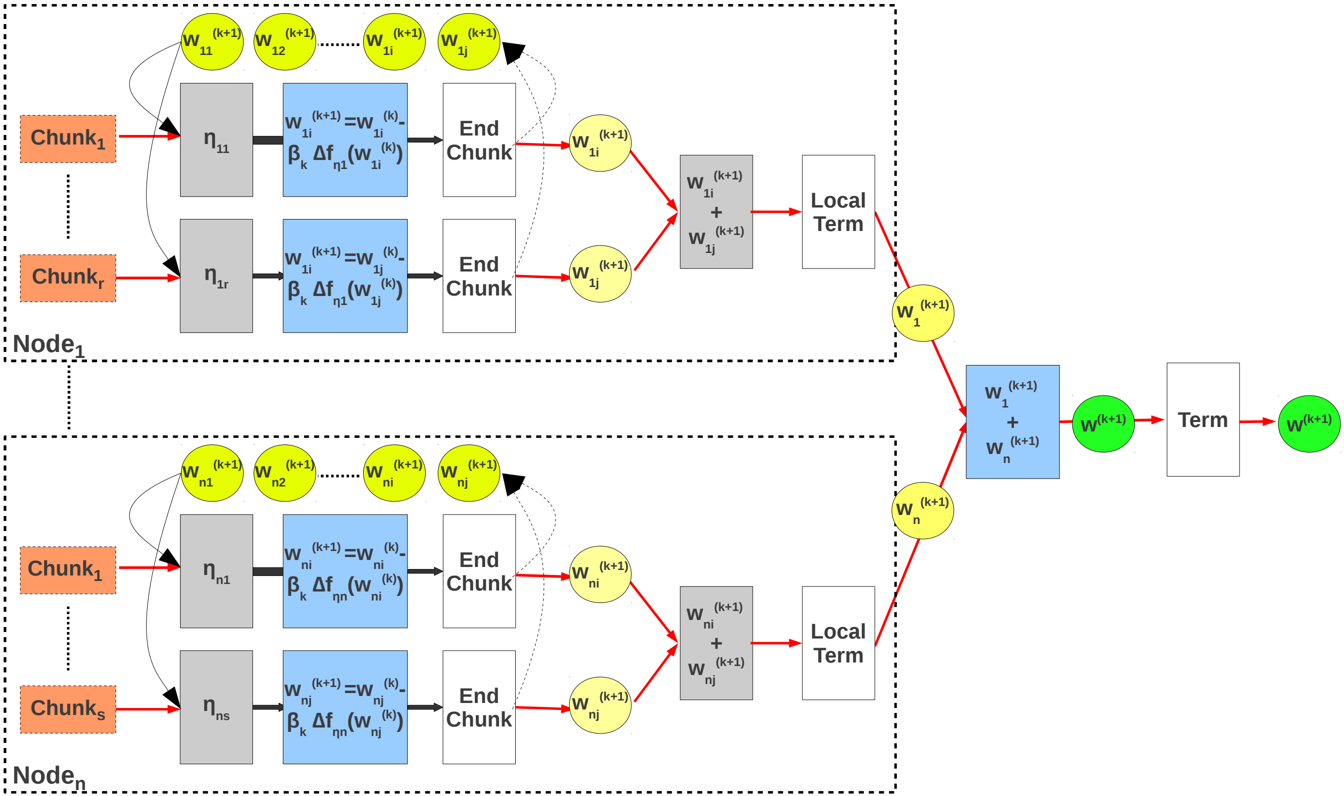


http://www.holehouse.org/mlclass/17_Large_Scale_Machine_Learning.html

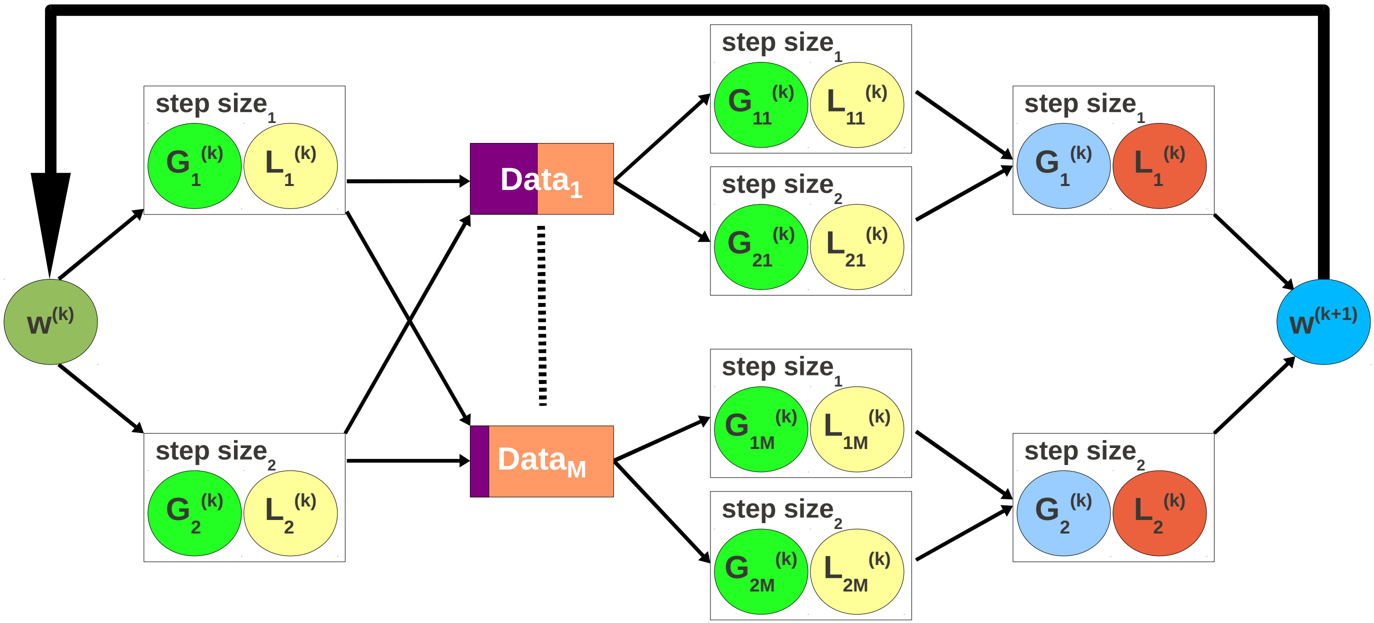
$$w^{(k+1)} = w^{(k)} - \beta_k \nabla f_{\eta(k)}\left(w^{(k)}\right)$$

- Approximate gradient at data point
- One step for each data point
- Faster convergence far from minimum

Gradient Descent as Generalized Linear Aggregate in GLADE

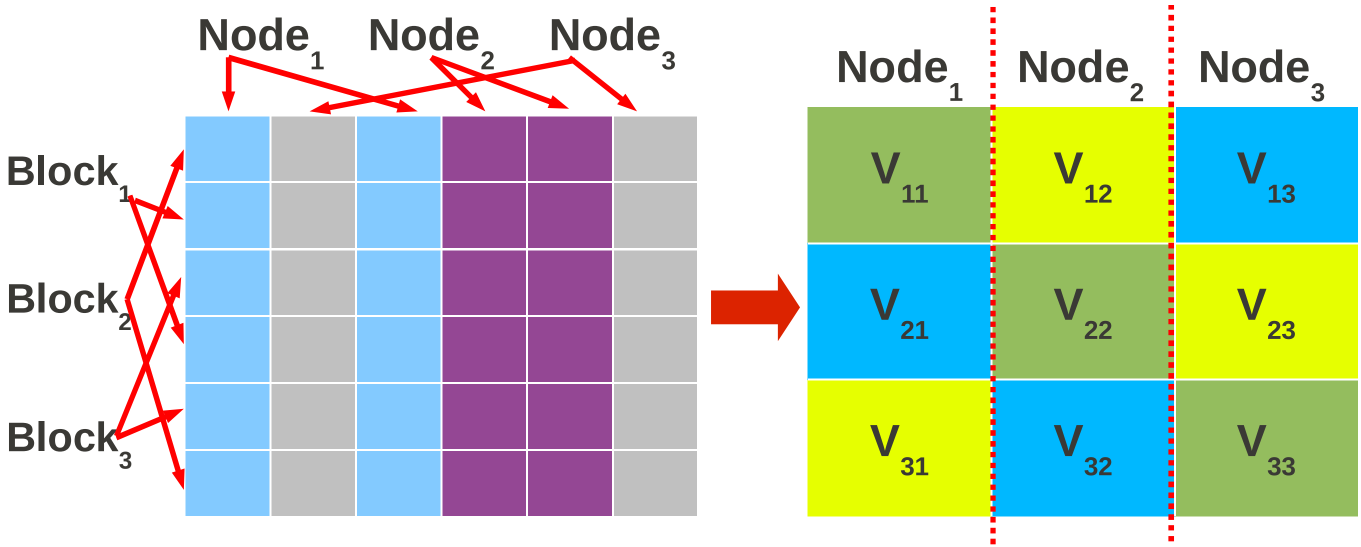


Online Hyper-Parameter Concurrent Testing



- Concurrent evaluation of multiple hyper-parameter configurations
- Online identification of sub-optimal configurations

Data- and Model-Partitioning Parallelism



- Consider model structure in partitioning
- Improve convergence by eliminating unnecessary model merges

Complete Description & Experimental Results

- C. Qin and F. Rusu: *"Scalable I/O-Bound Parallel Incremental Gradient Descent for Big Data Analytics in GLADE"* (2013)
- C. Qin and F. Rusu: *"Speeding-Up Distributed Low-Rank Matrix Factorization"* (2013)
- C. Qin and F. Rusu: *"Speculative Approximations for Terascale Analytics"* (2014)
- C. Qin and F. Rusu: *"Speculative Approximations for Terascale Distributed Gradient Descent Optimization"* (2015)