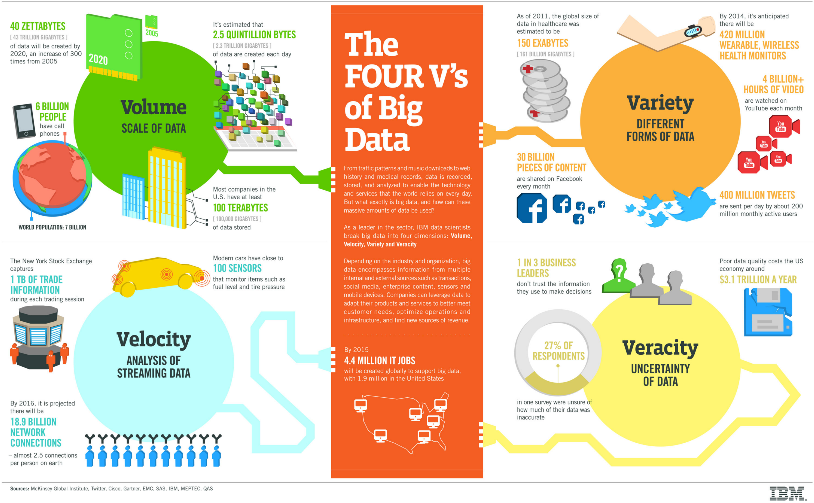
Online MM Algorithms for Machine Learning

Josh Day

NC State University
jtday2@ncsu.edu

December 21, 2016

Motivation


The FOUR V's of Big Data
From traffic patterns and music downloads to web history and medical records, data is recorded, stored, and analyzed to enable the technology and services that the world relies on every day. But what exactly is big data, and how can these massive amounts of data be used?
As a leader in the sector, IBM data scientists break big data into four dimensions: Volume, Velocity, Variety and Veracity
Depending on the industry and organization, big data encompasses information from multiple internal and external sources such as transactions, social media, enterprise content, sensors and mobile devices. Companies can leverage data to adapt their products and services to better meet customer needs, optimize operations and infrastructure, and find new sources of revenue.
By 2015
4.4 MILLION IT JOBS
will be created globally to support big data, with 1.9 million in the United States
Volume
SCALE OF DATA
40 ZETTABYTES
[43 TRILLION GIGABYTES]
of data will be created by 2020, an increase of 300 times from 2005
2005
2020
It's estimated that
2.5 QUINTILLION BYTES
[2.3 TRILLION GIGABYTES]
of data are created each day
6 BILLION PEOPLE
have cell phones
WORLD POPULATION: 7 BILLION
Most companies in the U.S. have at least
100 TERABYTES
[100,000 GIGABYTES]
of data stored
Velocity
ANALYSIS OF STREAMING DATA
The New York Stock Exchange captures
1 TB OF TRADE INFORMATION
during each trading session
Modern cars have close to
100 SENSORS
that monitor items such as fuel level and tire pressure
By 2016, it is projected there will be
18.9 BILLION NETWORK CONNECTIONS
– almost 2.5 connections per person on earth
Variety
DIFFERENT FORMS OF DATA
As of 2011, the global size of data in healthcare was estimated to be
150 EXABYTES
[161 BILLION GIGABYTES]
By 2014, it's anticipated there will be
420 MILLION WEARABLE, WIRELESS HEALTH MONITORS
4 BILLION+ HOURS OF VIDEO
are watched on YouTube each month
30 BILLION PIECES OF CONTENT
are shared on Facebook every month
400 MILLION TWEETS
are sent per day by about 200 million monthly active users
Veracity
UNCERTAINTY OF DATA
1 IN 3 BUSINESS LEADERS
don't trust the information they use to make decisions
Poor data quality costs the US economy around
$3.1 TRILLION A YEAR
27% OF RESPONDENTS
in one survey were unsure of how much of their data was inaccurate
Sources: McKinsey Global Institute, Twitter, Cisco, Gartner, EMC, SAS, IBM, MEPTEC, QAS

- Statisticians don't have the tools to handle all of this
- Adapting our favorite algorithms is often nontrivial
- Note: If we can handle Velocity, we can handle Volume

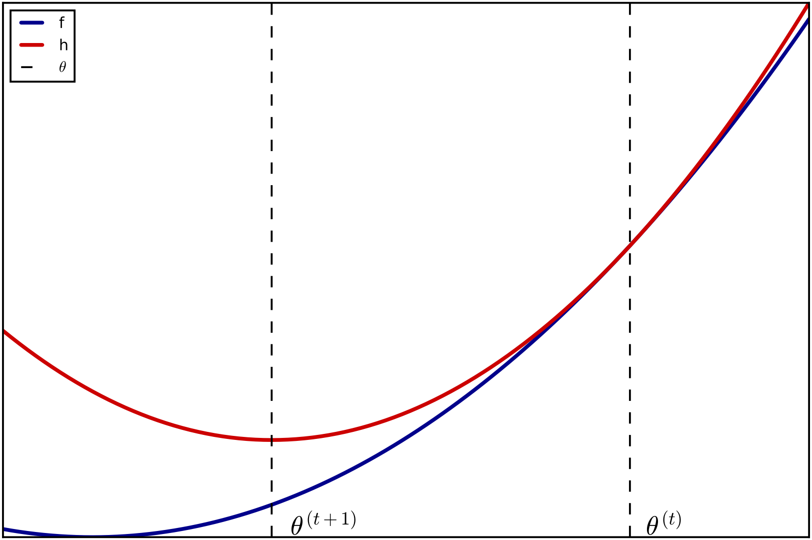
What is a Majorization Minimization Algorithm?

- h is said to *majorize* f at $\theta^{(t)}$ if:

$$h(\theta|\theta^{(t)}) \geq f(\theta|\theta^{(t)})$$
$$h(\theta^{(t)}|\theta^{(t)}) = f(\theta^{(t)}|\theta^{(t)})$$

- MM update: $\theta^{(t+1)} = \arg\min_\theta \; h(\theta|\theta^{(t)})$

MM Algorithms



What is an Online Algorithm?

Velocity can be handled by *online algorithms*

- Input enters piece by piece
- Entire dataset is not available at once
- A well known example: Stochastic Gradient Descent

$$\boldsymbol{\theta}^{(t)} = \boldsymbol{\theta}^{(t-1)} - \gamma_t \nabla f_t(\boldsymbol{\theta}^{(t-1)})$$

- The MM principle is common in offline optimization
  - Coordinate Descent, Proximal Gradient Method, ADMM, ...
- Where does it occur in online optimization?

Revisit SGD

- SGD minimizes quadratic approximations of noisy functions

$$h_t(\boldsymbol{\theta}) = f_t(\boldsymbol{\theta}^{(t-1)}) + \nabla f_t(\boldsymbol{\theta}^{(t-1)})^T(\boldsymbol{\theta} - \boldsymbol{\theta}^{(t-1)}) + \frac{1}{2\gamma_t}\|\boldsymbol{\theta} - \boldsymbol{\theta}^{(t-1)}\|_2^2$$

$$\begin{aligned}\boldsymbol{\theta}^{(t)} &= \boldsymbol{\theta}^{(t-1)} - \gamma_t \nabla f_t(\boldsymbol{\theta}^{(t-1)}) \\ &= \arg\min_{\boldsymbol{\theta}} h_t(\boldsymbol{\theta})\end{aligned}$$

- L-Lipschitz continuous gradient

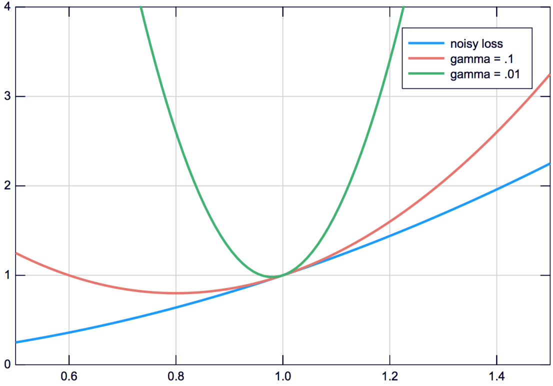
$$f(\boldsymbol{\theta}) \leq f(\boldsymbol{u}) + \nabla f(\boldsymbol{u})^T(\boldsymbol{\theta} - \boldsymbol{u}) + \frac{L}{2}\|\boldsymbol{\theta} - \boldsymbol{u}\|_2^2$$

- i.e., there exists a quadratic upper bound
- RHS is a majorizing function

Suppose $f_t(\theta)$ has L_t-Lipschitz continuous gradient

- SGD updates are MM updates anytime $\gamma_t^{-1} \leq L_t$

- Majorizations get worse over time ($\gamma_t \to 0$), forcing the parameter to remain near its current state



Online MM

- Define an Online MM Algorithm as one which follows these steps for each observation/minibatch of data:

1. observe $f_t(\boldsymbol{\theta})$
2. create majorization $h_t(\boldsymbol{\theta})$
3. Update surrogate objective function $Q_t(\boldsymbol{\theta})$ with $h_t(\boldsymbol{\theta})$
4. $\boldsymbol{\theta}^{(t+1)} = argmin_{\boldsymbol{\theta}} Q_t(\boldsymbol{\theta})$

- Algorithms differ at steps 2 (the majorization) and 3 (how Q_t is updated)
- Ex: SGD
  - h_t = quadratic upper bound (m_t-strongly convex, $m_t \to \infty$)
  - $Q_t(\boldsymbol{\theta}) = h_t(\boldsymbol{\theta})$

Online MM Type 1

- Rare case when there exists "sufficient statistics" for majorizing $\frac{1}{t}\sum_{\tau=1}^{t} f_\tau(\boldsymbol{\theta})$ analytically
- Online majorization = offline majorization
- However, to get same estimate as offline algorithm, we need multiple iterations per update

Online MM Type 2

- $Q_t(\theta) = (1 - \gamma_t) Q_{t-1}(\theta) + \gamma_t h_t(\theta)$
- Weighted average of surrogate objective and noisy majorization

Online MM Type 3

- $Q_t(\theta) = h_t(\theta)$, h_t is m_t strongly convex, $m_t \to \infty$
- SGD-like algorithms (ADAGRAD, ADAM)
- Are there non-SGD algorithms?

Which types can you use?

- Type 1? Only if majorization depends on O(1) sufficient statistics
- Type 2? Always
- Type 3? Only if majorization can get worse as $t \to \infty$

Can you Make a Majorization "Worse"?

✓ Quadratic upper bound

$$h_t(\boldsymbol{\theta}) = f_t(\boldsymbol{u}) + \nabla f_t(\boldsymbol{u})^T(\boldsymbol{\theta} - \boldsymbol{u}) + \frac{L}{2\gamma_t}\|\boldsymbol{\theta} - \boldsymbol{u}\|_2^2$$

✗ Definition of convexity with linear predictor

$$h_t(\boldsymbol{x}_t^T\boldsymbol{\beta}) = \sum_j \alpha_{tj} f_t\left[\frac{x_{tj}}{\alpha_{tj}}(\beta_j - \beta_j^{(t-1)}) + \boldsymbol{x}_t^T\boldsymbol{\beta}^{(t-1)}\right]$$

Online **MM** with **Q**uadratic upper bound using a **D**iagonal Hessian approximation

- If $\boldsymbol{H} - d^2 f(\boldsymbol{\theta})$ is nonnegative definite, then

$$h(\boldsymbol{\theta}) = f(\boldsymbol{u}) + \nabla f(\boldsymbol{u})^T(\boldsymbol{\theta} - \boldsymbol{u}) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{u})\boldsymbol{H}(\boldsymbol{\theta} - \boldsymbol{u})$$

  is a majorizing function
- However, parameters only split if $\boldsymbol{H}$ is diagonal

- Type 2 Online MM with Quadratic Upper Bound is

$$\boldsymbol{\theta}^{(t)} = \boldsymbol{A}_t^{-1}\boldsymbol{b}_t,$$

where

$$\begin{aligned}
\boldsymbol{A}_t &= (1-\gamma_t)\boldsymbol{A}_{t-1} + \gamma_t \boldsymbol{H}_t \\
\boldsymbol{b}_t &= (1-\gamma_t)\boldsymbol{b}_{t-1} + \gamma_t[\boldsymbol{H}_t\boldsymbol{\theta}^{(t-1)} - \nabla f_t(\boldsymbol{\theta}^{(t-1)})]
\end{aligned}$$

Applying regularization

For machine learning problems

$$\frac{1}{n}\sum f_i(\boldsymbol{\theta}) + g(\boldsymbol{\theta}),$$

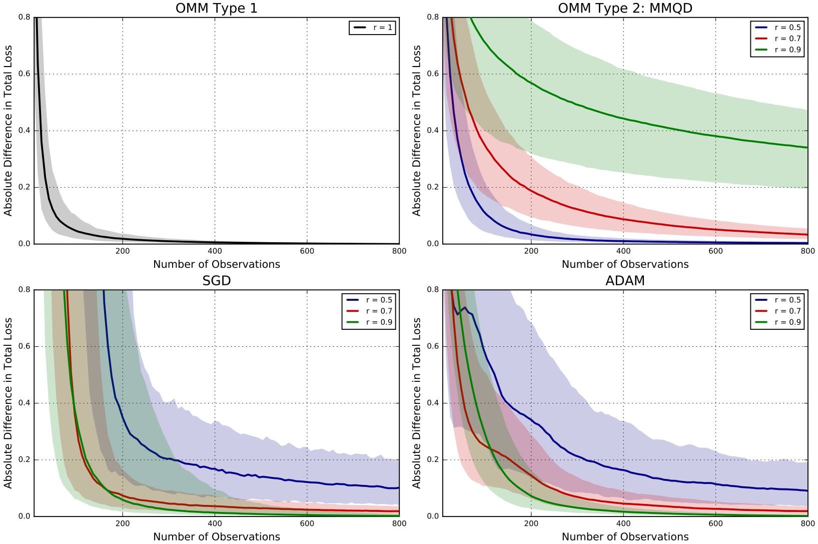
the MMQD element-wise updates simplify to

$$\theta_j^{(t)} = \mathbf{prox}_{h_{jj}^{-1} g}\left(\frac{b_{tj}}{h_{jj}}\right)$$

- The following plots display the average loss of various online algorithms (using several different learning rates) relative to the offline solution for 1000 replications of simulated data.
- Solid line: median relative loss
- Ribbons: 0.05 and 0.95 quantiles

Linear Regression

- Learning Rate: $\gamma_t = t^{-r}, r \in [.5, .7, .9]$



Thank You

- Software available:
  - OnlineStats.jl
  - OnlineStatsModels.jl
- Questions?