Big Data Optimization at SAS

Imre Pólik et al.
SAS Institute
Cary, NC, USA

Edinburgh, 2013
Outline

1. Optimization at SAS

2. Big Data Optimization at SAS
   - The SAS HPA architecture
   - Support vector machines
   - Quantile regression
   - Marketing Optimization
   - Local search optimization

3. Distributed/parallel optimization
   - Decomposition
   - Miscellaneous tools

4. Future plans
# About SAS

## The company

- Leader in business analytics software and services
- About $3 billion worldwide revenue
- Largest private software company in the world
- World’s Best Multinational Workplace in 2012
- More than 11,000 employees, 400 offices and 600 alliances
- SAS customers or their affiliates represent over 90% of the top 100 FORTUNE 500 companies

## The software

- Originally created for basic statistics by professors at NCSU
- Extended tremendously over the decades
- Covers all aspects of analytics and business intelligence
## SAS/OR Offerings

### Optimization modelling and solvers

Algebraic Modelling Language with all the usual solvers (LP, QP, MILP, NLP, CP, Scheduling, Decomposition, ...)

### Other tools

Graph and network algorithms
Discrete event simulation + the rest of SAS

### Solutions

Marketing Optimization, Service Parts Optimization, Revenue Optimization, Size Optimization, ...

### Services

Technical Support, Training, Professional Services, Consulting

### Platforms

Windows, Linux, Solaris x64/SPARC, HP-UX, AIX, z/OS
What is Big Data?

Nathan Brixius (in a recent blog post)

A big data analytics application is simply an analytics application where
- the required data does not fit on a single machine and
- needs to be considered in full to produce a result.

SAS

Big data is relative; it applies whenever an organization’s need to handle, store and analyze data exceeds its current capacity.

Related concepts/tools
- large-scale optimization
- distributed optimization
- parallel optimization
High Performance Analytics at SAS
## Parallel Implementations and Determinism

### Non-determinism

A *dirty word* in the commercial world

### Sources of non-determinism

- Adding columns in a different order
- Aggregating results in a different order
- Arbitrary random number seeds
- Different machines in the pool
- Time limits

### Workarounds

- Operations in a fixed order
- Deterministic criteria (nodes, iterations)
- *Deterministic ticks* (see Xpress/Cplex)

Determinism comes with a performance penalty.
Big Data Optimization at SAS

- Quadratic programming
- Support vector machine
- Linear programming
- Quantile regression
- Mixed-integer linear programming
- Marketing optimization
- Derivative-free optimization
- Local search optimization
Support Vector Machines

http://www.cac.science.ru.nl/people/ustun/SVM.JPG
### Primal formulation

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \|w\|^2 + \tau e^T z \\
\text{subject to} & \quad Yw - \beta d \geq e - z \\
& \quad z \geq 0.
\end{align*}
\]

### Dual formulation

\[
\begin{align*}
\text{minimize} & \quad -e^T v + \frac{1}{2} v^T Y Y^T v \\
\text{subject to} & \quad d^T v = 0, \\
& \quad 0 \leq v \leq \tau e.
\end{align*}
\]
Using Primal-Dual Interior-Point Approach

Dominant cost per iteration is forming/solving Newton system

Many more observations than columns/features

\[(I + Y^T \Omega_k^{-1} Y - v_k v_k^T) \Delta w = -r_w\]

\(Y^T \Omega_k^{-1} Y\) must be formed every iteration

Many more columns/features than observations

\[
\begin{pmatrix}
YY^T + \Omega_k & d \\
d^T & 0
\end{pmatrix}
\begin{pmatrix}
\Delta v \\
-\Delta \beta
\end{pmatrix}
= -
\begin{pmatrix}
\rho \beta \\
r \Omega
\end{pmatrix}
\]

\(YY^T\) constant for all iterations
SVM: Parallel Performance

Gv_100vars_1mows (DIRECT)

- 1 thread
- 2 threads
- 4 threads
- 8 threads

[Bar chart showing performance comparison across different thread counts]
SVM: Features and Plans

Features
- frequency/weight term
- iterative (PCG) or direct (Cholesky, threaded) method to solve the Newton system
- balance threads to avoid cache misses
- balance number of compute nodes to limit communication

In progress
- nonlinear SVM
- build a distributed QP solver

Available soon in SAS/OR
Quantile Regression

Goal
Approximate the median or some other quantile of the response variable of a number of observations

\[
\min \tau u^+ + (1 - \tau) u^- \\
A(\beta^+ - \beta^-) + u^+ + u^- = b \\
\beta^+, \beta^-, u^+, u^- \geq 0
\]

- \(A\): observations in rows, fully dense
- \(b\): response variable
- \(\tau\): quantile level

Problem size
Up to \(10^8\) observations each of dimension \(10^4\)
Quantile Regression

**Features**
- distributed IPM, similar to the SVM case
- Newton system solved directly or iteratively
- different preconditioner

**Plans**
- categorical variables – sparse observations
- nonlinear quantile regression
- build a distributed dense LP/IPM solver

Available soon in SAS/OR
### Marketing Optimization

**Problem**
Assign ads/offers to customers based on budget, policy, user preferences, history and other kinds of constraints.

**Formulation**
Typical sizes: millions of customers, hundreds of offers  
Formulated as a MILP with millions of binary variables

**Solution**
Special decomposition  
Subproblems are solved distributed on the grid  
Subgradient algorithm for the master  
Available as a SAS solution.
Marketing Optimization

Typical data (telecommunications)

- 15 million customers
- 910 communications
- 14 aggregate constraints
- 19 rolling contact policies (per day, per week, per month)
- 90 million offers in the contact history

Performance

- Used to take 10 hours on a single machine with regular MO
- Solved in 2 minutes on an EMC Greenplum appliance (32 nodes, 24 threads, 48GB ram)
- Allows for scenario analysis
Local Search Optimization

Algorithm
- GA-guided pattern search
- Continuous, discrete and categorical variables
- Up to about 100 variables

Implementation
- Classical: each worker evaluates the function at a given point
- Big data: each worker computes part of the function value from its own data, then these are aggregated
- Function value cache
- Leading to simulation-based optimization

Available as part of SAS/OR
Parallel Optimization

Not Big Data, but uses the same infrastructure

- Decomposition
- Multistart NLP
- Option tuner for MILP
Decomposition — Outline

Algorithm
- Dantzig-Wolfe Decomposition embedded in B&B
  - a specific variant of column generation
- Relax $A$ and subproblem $B$ becomes tractable (even separable)
- Find convex combinations of extreme points of subproblems that satisfy the continuous relaxation of the master constraints
- Iterate between master $A$ (reformulated space) and $B$

Blocks
From user, network or auto

Available in SAS/OR

\[
\begin{pmatrix}
A_1 & A_2 & \cdots & A_{|K|} \\
B_1 \\
B_2 \\
\vdots \\
B_{|K|}
\end{pmatrix}
\]
### Decomposition — Parallel Implementation

#### Implementation
- **Shared** (Threaded) and/or **Distributed** Memory (Gridded)
- Subproblems use a standard queue

#### Areas of parallelism
- Branch & Bound
  - ✓ Heuristics (non-blocking price-and-branch)
  - ✓ Subproblem solves (across subproblems)
  - ✓ Master solve (IPM or concurrent)
  - ✓ Subproblem solves (for each subproblem)

#### Factors affecting parallel performance
- percentage of time in subprob vs master (modeling)
- load balance — aggregate subproblems
- enforce balance with time limits? (non-deterministic)
- MPI overhead — jobs must be *significant*
Other parallel optimization applications

<table>
<thead>
<tr>
<th>Network tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Graph centrality, community detection</td>
</tr>
<tr>
<td>- Social network analysis for fraud detection</td>
</tr>
<tr>
<td>- Marketing analysis for telecommunications</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Multistart NLP (Global optimization)</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Standard NLP solvers started from different points</td>
</tr>
<tr>
<td>- Function evaluations and solvers are distributed</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Option tuner for MILP</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Find the best option setting for a set of MILP problems</td>
</tr>
<tr>
<td>- Continuous, discrete and categorical options are all included</td>
</tr>
</tbody>
</table>
Extend the list of HP enabled procedures

- driven by customers’ needs
- distributed LP/QP
- distributed graph algorithms
- parallel MILP
- parallel solves in OPTMODEL
- simulation-based optimization
Thank you for your attention.

imre@polik.net