PARALLEL COORDINATE DESCENT FOR L1-REGULARIZED LOSS MINIMIZATION

Joseph K. Bradley, Aapo Kyrola, Danny Bickson, Carlos Guestrin (Select Lab, Carnegie Mellon University)

L1-REGULARIZED REGRESSION

Example application of regression

Stock volatility (label) ← Bigrams from financial reports (features)

(Kogan et al., 2009)

$5x10^6$ features
$3x10^4$ samples

- Produces **sparse** solutions
- Useful in **high-dimensional** settings (# features >> # examples)

- Lasso (Tibshirani, 1996)
- Sparse logistic regression (Ng, 2004)

Lasso (Tibshirani, 1996)

Goal: Regress $y \in \Re$ on $\mathbf{a} \in \Re^d$, given samples $\{(\mathbf{a}_i, y_i)\}_i$

Objective: $\min_{\mathbf{x}} F(\mathbf{x})$ where $F(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \lambda\|\mathbf{x}\|_1$

Squared error + L1 regularization

PARALLELIZING OPTIMIZATION

Many possible algorithms
- Gradient descent, stochastic gradient, interior point, hard/soft thresholding, ...
- Coordinate descent (a.k.a. **Shooting** (Fu, 1998))

⟹ One of the fastest algorithms (Yuan et al., 2010)

We use the **multicore** setting:
- shared memory
- low latency

We could parallelize:
- Matrix-vector ops (E.g., interior point) ⟹ Not great empirically.
- W.r.t. samples (E.g., stochastic gradient) ⟹ Best for many samples, not many features.
  (Zinkevich et al., 2010) ⟹ Analysis not for L1.
- W.r.t. features (E.g., shooting) ⟹ Inherently sequential? **Surprisingly, no!**

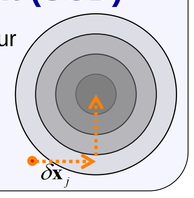
SHOOTING TO SHOTGUN

Shooting: Sequential Stochastic Coordinate Descent (SCD)

Shooting Algorithm (e.g., Shalev-Shwartz & Tewari, 2009)

While not converged,
- Choose random coordinate j,
- Update x_j (closed-form minimization): $\mathbf{x}_j \leftarrow \mathbf{x}_j + \delta\mathbf{x}_j$

$F(\mathbf{x})$ contour

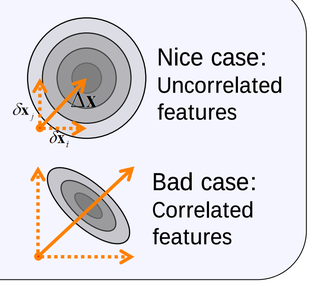


Shotgun: Parallel SCD

Shotgun Algorithm

While not converged,
- On each of P cores,
  - Choose random coordinate j,
  - Update x_j (same as for Shooting)

Collective update:
$\Delta\mathbf{x} = (\delta\mathbf{x}_i \ 0 \ 0 \ \delta\mathbf{x}_j \ 0)$



Nice case: Uncorrelated features

Bad case: Correlated features

Potential for Parallelism

Theorem: If A is normalized s.t. $\mathrm{diag}(A^TA)=1$,

Decrease of objective ≤ Sequential progress + Interference

$$F(\mathbf{x}+\Delta\mathbf{x}) - F(\mathbf{x}) \le -\frac{1}{2}\sum_{i_j \in \mathcal{P}} (\delta\mathbf{x}_{i_j})^2 + \frac{1}{2}\sum_{\substack{i_j, i_k \in \mathcal{P}, \\ j \neq k}} (\mathbf{A}^T\mathbf{A})_{i_j, i_k} \delta\mathbf{x}_{i_j}\delta\mathbf{x}_{i_k}$$

Sum over updated coordinates

$(A^TA)_{jk}=0$ for $j \neq k$. (if A^TA is centered)

Nice case: Uncorrelated features

$(A^TA)_{jk} \neq 0$

Bad case: Correlated features

SHOTGUN CONVERGENCE ANALYSIS

Main Theorem

Assume # parallel updates $P < \frac{1}{2}d/\rho + 1$

$\mathbf{x} \in \Re^d$ ρ = spectral radius of $\mathbf{A}^T\mathbf{A}$

Up to a threshold...

$$E\left[F(\mathbf{x}^{(T)})\right] - F(\mathbf{x}^*) \le \frac{d \cdot \left(\frac{1}{2}\|\mathbf{x}^*\|_2^2 + 2F(\mathbf{x}^{(0)})\right)}{T \cdot P}$$

Final objective − Optimum; iterations

linear speedups predicted.

Generalizes bounds for Shooting (from Shalev-Shwartz & Tewari, 2009)

Naive parallelization of coordinate descent works!

Nice case: Uncorrelated features

$\rho = 1 \Rightarrow P_{\max} = d$

We can update all features at once.

Bad case: Correlated features

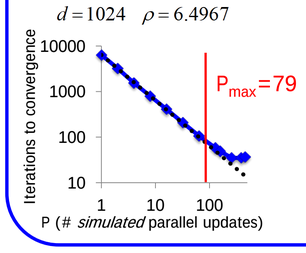
$\rho = d \Rightarrow P_{\max} = 1$ (at worst)

We can update only 1 feature at a time.

Experiments match the theory.

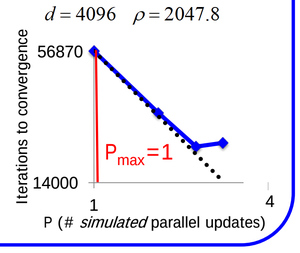
Mug32_singlepixcam

$d = 1024 \quad \rho = 6.4967$



Ball64_singlepixcam

$d = 4096 \quad \rho = 2047.8$



EXPERIMENTS: LASSO

7 Algorithms
- **Shotgun P=8 (multicore)**
- Shooting (Fu, 1998)
- Interior point (Parallel L1_LS) (Kim et al., 2007)
- Shrinkage (FPC_AS, SpaRSA) (Wen et al., 2010; Wright et al., 2009)
- Projected gradient (GPSR_BB) (Figueiredo et al., 2008)
- Iterative hard thresholding (Hard_l0) (Blumensath & Davies, 2009)
- Also ran: GLMNET, LARS, SMIDAS

35 Datasets
- # samples n: [128, 209432]
- # features d: [128, 5845762]

λ=.5, 10 (dense, sparse results)

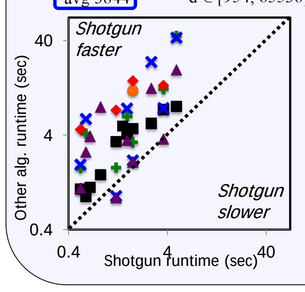
Hardware
- 8 core AMD Opteron 8384 (2.69 GHz)

Shotgun & Parallel L1_LS used 8 cores. Other algorithms are sequential.

Optimization Details
- Pathwise optimization (continuation)
- Asynchronous Shotgun with atomic operations

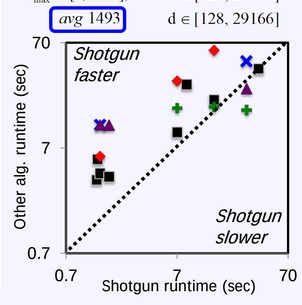
Sparse Compressed Imaging

$P_{\max} \in [1432, 5889]$, avg 3844 — $n \in [477, 32768]$, $d \in [954, 65536]$



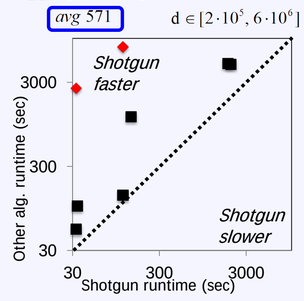
Sparco (van den Berg et al., 2009)

$P_{\max} \in [1, 8683]$, avg 1493 — $n \in [128, 29166]$, $d \in [128, 29166]$



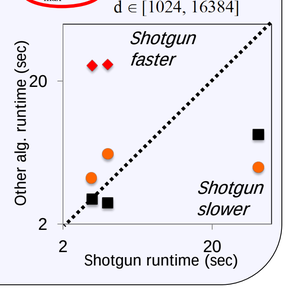
Large, Sparse Datasets

$P_{\max} \in [107, 1036]$, avg 571 — $n \in [3 \cdot 10^4, 2 \cdot 10^5]$, $d \in [2 \cdot 10^5, 6 \cdot 10^6]$



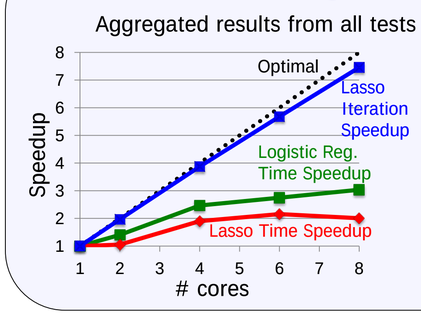
Single-Pixel Camera (Duarte et al., 2008)

$P_{\max} = 1$ — $n \in [410, 4770]$, $d \in [1024, 16384]$



Shotgun Self-Speedup

Aggregated results from all tests



Mediocre time speedups. ☹
But fewer iterations! ☺

Explanation:
Memory wall (Wulf & McKee, 1995)
Memory bus gets flooded.

Logistic regression uses more FLOPS/datum.
➔ Extra computation hides memory latency.
➔ Better speedups!

EXPERIMENTS: LOGISTIC REGRESSION

Algorithms
- Shooting (CDN)
- Shotgun CDN
- Stochastic Gradient Descent (SGD)
- Parallel SGD (Zinkevich et al., 2010)
  - Averages results of 8 instances run in parallel

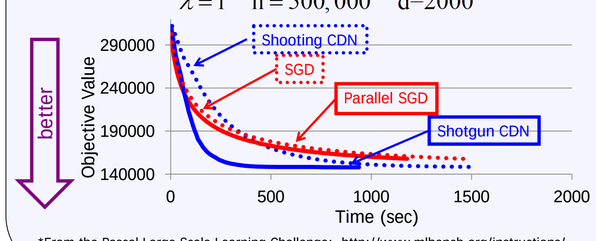
Shotgun & Parallel SGD used 8 cores.

Coordinate Descent Newton (CDN) (Yuan et al., 2010)
- Uses line search
- Extensive tests show CDN is very fast.

SGD
- Lazy shrinkage updates (Langford et al., 2009)
- Used best of 14 learning rates

Zeta* dataset: low-dimensional setting

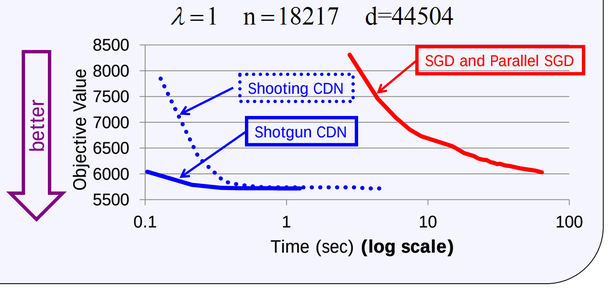
$\lambda = 1 \quad n = 500,000 \quad d = 2000$



*From the Pascal Large Scale Learning Challenge: http://www.mlbench.org/instructions/

rcv1 dataset (Lewis et al, 2004): **high-dimensional setting**

$\lambda = 1 \quad n = 18217 \quad d = 44504$



FUTURE WORK

- Distributed setting
- Hybrid Shotgun + parallel SGD
- More FLOPS/datum, e.g., Group Lasso (Yuan and Lin, 2006)
- Alternate hardware, e.g., graphics processors

Code and Data
http://www.select.cs.cmu.edu/projects

References

- Blumensath, T. and Davies, M.E. Iterative hard thresholding for compressed sensing. Applied and Computational Harmonic Analysis, 27(3):265–274, 2009.
- Duarte, M.F., Davenport, M.A., Takhar, D., Laska, J.N., Sun, T., Kelly, K.F., and Baraniuk, R.G. Single-pixel imaging via compressive sampling. Signal Processing Magazine, IEEE, 25(2):83–91, 2008.
- Figueiredo, M.A.T, Nowak, R.D., and Wright, S.J. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. IEEE J. of Sel. Top. in Signal Processing, 1(4):586–597, 2008.
- Friedman, J., Hastie, T., and Tibshirani, R. Regularization paths for generalized linear models via coordinate descent. J. of Statistical Software, 33(1):1–22, 2010.
- Fu, W.J. Penalized regressions: The bridge versus the lasso. J. of Comp. and Graphical Statistics, 7(3):397– 416, 1998.
- Kim, S. J., Koh, K., Lustig, M., Boyd, S., and Gorinevsky, D. An interior-point method for large-scale l1-regularized least squares. IEEE Journal of Sel. Top. in Signal Processing, 1(4):606–617, 2007.
- Kogan, S., Levin, D., Routledge, B.R., Sagi, J.S., and Smith, N.A. Predicting risk from financial reports with regression. In Human Language Tech.-NAACL, 2009.
- Langford, J., Li, L., and Zhang, T. Sparse online learning via truncated gradient. In NIPS, 2009a.
- Lewis, D.D., Yang, Y., Rose, T.G., and Li, F. RCV1: A new benchmark collection for text categorization research. JMLR, 5:361–397, 2004.
- Ng, A.Y. Feature selection, l1 vs. l2 regularization and rotational invariance. In ICML, 2004.
- Shalev-Shwartz, S. and Tewari, A. Stochastic methods for l1 regularized loss minimization. In ICML, 2009.
- Tibshirani, R. Regression shrinkage and selection via the lasso. J. Royal Statistical Society, 58(1):267–288, 1996.
- van den Berg, E., Friedlander, M.P., Hennenfent, G., Herrmann, F., Saab, R., and Yilmaz, O. Sparco: A testing framework for sparse reconstruction. ACM Transactions on Mathematical Software, 35(4):1–16, 2009.
- Wen, Z., Yin, W. Goldfarb, D., and Zhang, Y. A fast algorithm for sparse reconstruction based on shrinkage, subspace optimization and continuation. SIAM Journal on Scientific Computing, 32(4):1832–1857, 2010.
- Wright, S.J., Nowak, D.R., and Figueiredo, M.A.T. Sparse reconstruction by separable approximation. IEEE Trans. on Signal Processing, 57(7):2479–2493, 2009.
- Wulf, W.A. and McKee, S.A. Hitting the memory wall: Implications of the obvious. ACM SIGARCH Computer Architecture News, 23(1):20–24, 1995.
- Yuan, G. X., Chang, K. W., Hsieh, C. J., and Lin, C. J. A comparison of optimization methods and software for large-scale l1-reg. linear classification. JMLR, 11:3183– 3234, 2010.
- Zinkevich, M., Weimer, M., Smola, A.J., and Li, L. Parallelized stochastic gradient descent. In NIPS, 2010.

Intel Science & Technology Center for Cloud Computing

Carnegie Mellon University · Georgia Tech · intel · Princeton University · UC Berkeley