Coordinate descent

Geoff Gordon & Ryan Tibshirani
Optimization 10-725 / 36-725

Adding to the toolbox, with stats and ML in mind

We've seen several general and useful minimization tools

- First-order methods
- Newton's method
- Dual methods
- Interior-point methods

These are some of the core methods in optimization, and they are the main objects of study in this field

In statistics and machine learning, there are a few other techniques that have received a lot of attention; these are not studied as much by those purely in optimization

They don't apply as broadly as above methods, but are interesting and useful when they do apply ... our focus for the next 2 lectures

Coordinate-wise minimization

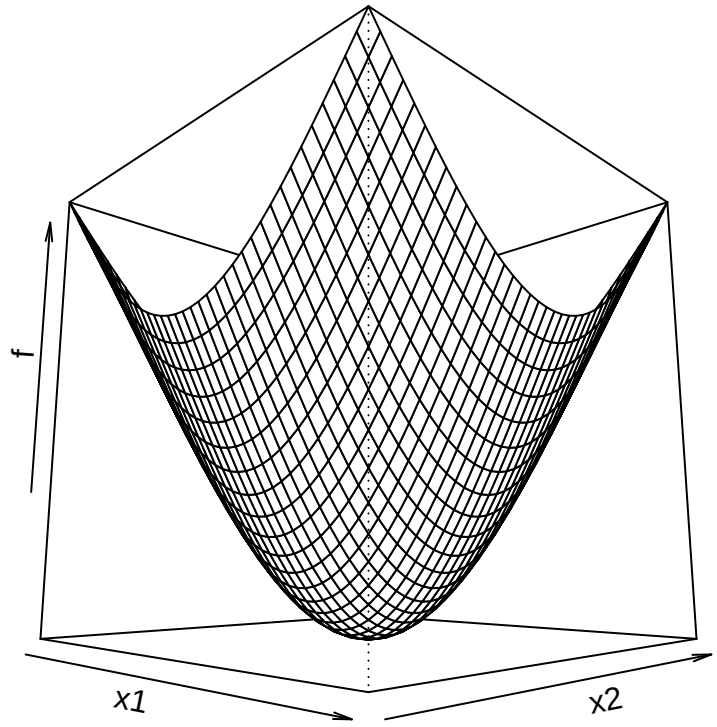
We've seen (and will continue to see) some pretty sophisticated methods. Today, we'll see an extremely **simple** technique that is surprisingly efficient and scalable

Focus is on **coordinate-wise minimization**

Q: Given convex, differentiable $f : \mathbb{R}^n \to \mathbb{R}$, if we are at a point x such that $f(x)$ is minimized along each coordinate axis, *have we found a global minimizer?*

I.e., does $f(x + d \cdot e_i) \geq f(x)$ for all $d, i \;\Rightarrow\; f(x) = \min_z f(z)$?

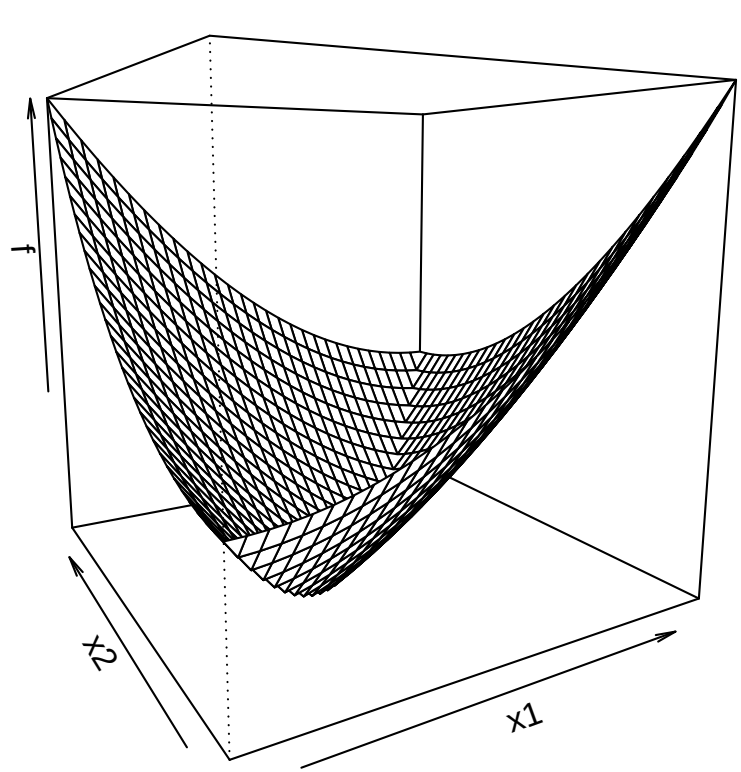
(Here $e_i = (0, \ldots, 1, \ldots 0) \in \mathbb{R}^n$, the ith standard basis vector)

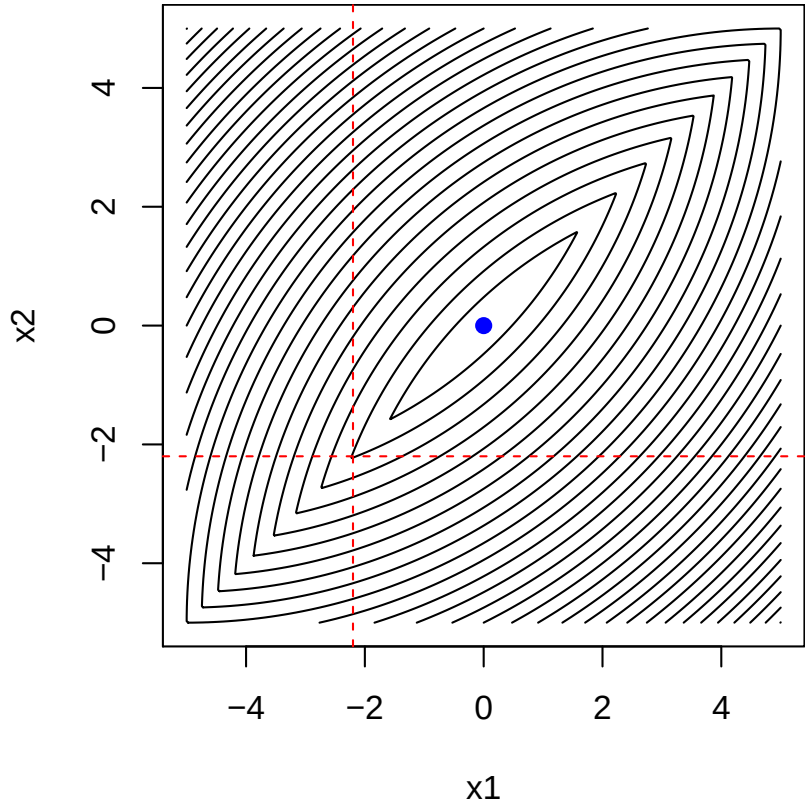


A: Yes! Proof:

$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}(x), \ldots \frac{\partial f}{\partial x_n}(x) \right) = 0$$

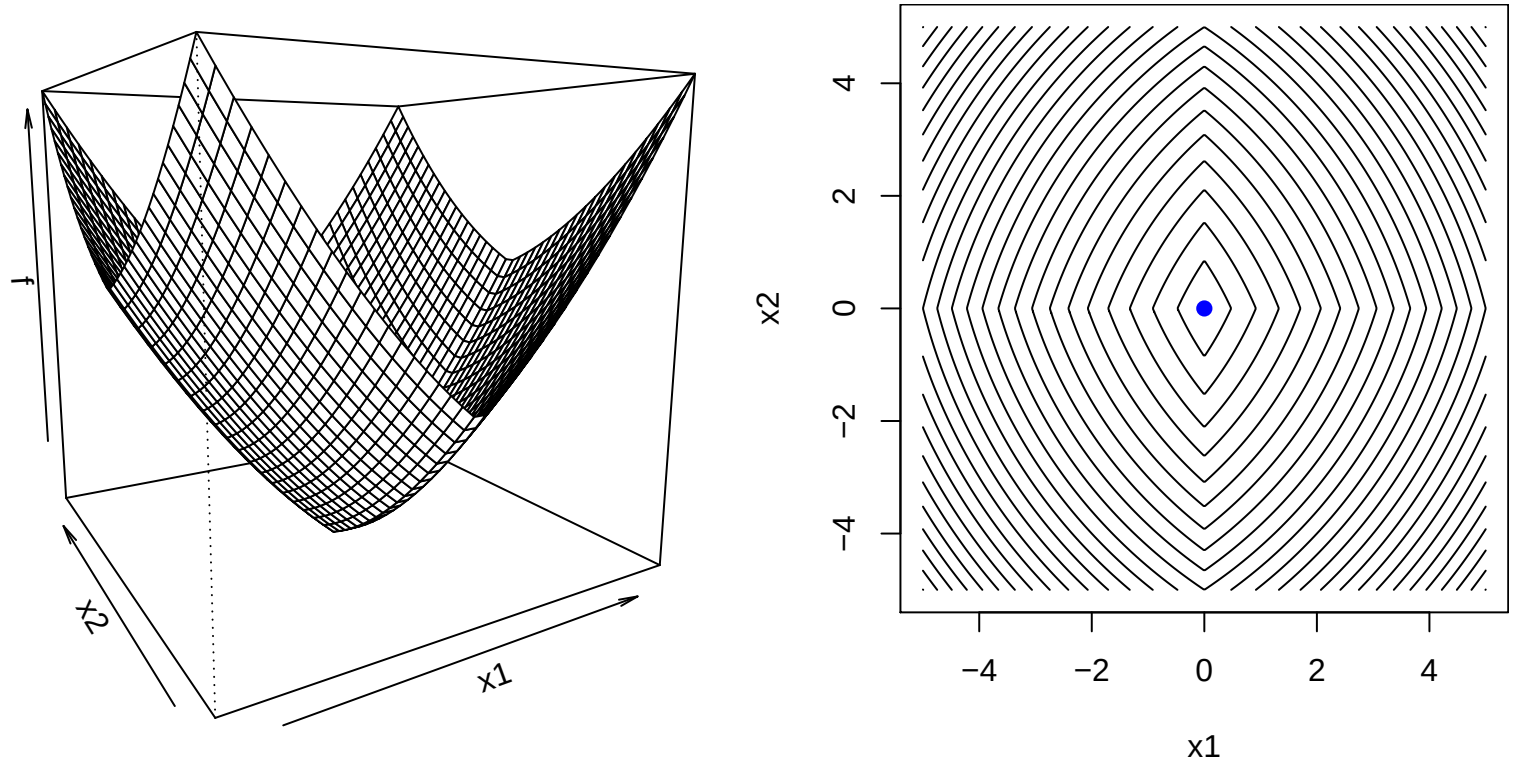
Q: Same question, but for f convex (not differentiable) ... ?





A: No! Look at the above counterexample

Q: Same question again, but now $f(x) = g(x) + \sum_{i=1}^{n} h_i(x_i)$, with g convex, differentiable and each h_i convex ... ? (Non-smooth part here called **separable**)



A: Yes! Proof: for any y,

$$f(y) - f(x) \geq \nabla g(x)^T (y - x) + \sum_{i=1}^{n} [h_i(y_i) - h_i(x_i)]$$

$$= \sum_{i=1}^{n} \underbrace{[\nabla_i g(x)(y_i - x_i) + h_i(y_i) - h_i(x_i)]}_{\geq 0} \geq 0$$

Coordinate descent

This suggests that for $f(x) = g(x) + \sum_{i=1}^n h_i(x_i)$ (with g convex, differentiable and each h_i convex) we can use **coordinate descent** to find a minimizer: start with some initial guess $x^{(0)}$, and repeat for $k = 1, 2, 3, \ldots$

$$
\begin{aligned}
x_1^{(k)} &\in \underset{x_1}{\operatorname{argmin}}\ f\big(x_1, x_2^{(k-1)}, x_3^{(k-1)}, \ldots x_n^{(k-1)}\big) \\
x_2^{(k)} &\in \underset{x_2}{\operatorname{argmin}}\ f\big(x_1^{(k)}, x_2, x_3^{(k-1)}, \ldots x_n^{(k-1)}\big) \\
x_3^{(k)} &\in \underset{x_2}{\operatorname{argmin}}\ f\big(x_1^{(k)}, x_2^{(k)}, x_3, \ldots x_n^{(k-1)}\big) \\
&\ldots \\
x_n^{(k)} &\in \underset{x_2}{\operatorname{argmin}}\ f\big(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \ldots x_n\big)
\end{aligned}
$$

Note: after we solve for $x_i^{(k)}$, we use its new value from then on

Seminal work of Tseng (2001) proves that for such f (provided f is continuous on compact set $\{x : f(x) \leq f(x^{(0)})\}$ and f attains its minimum), any limit point of $x^{(k)}$, $k = 1, 2, 3, \ldots$ is a minimizer of f. Now, citing real analysis facts:

- $x^{(k)}$ has subsequence converging to $x^\star$ (Bolzano-Weierstrass)
- $f(x^{(k)})$ converges to $f^\star$ (monotone convergence)

Notes:

- Order of cycle through coordinates is arbitrary, can use any permutation of $\{1, 2, \ldots n\}$
- Can everywhere replace individual coordinates with blocks of coordinates
- "One-at-a-time" update scheme is critical, and "all-at-once" scheme **does not** necessarily converge

Linear regression

Let $f(x) = \frac{1}{2}\|y - Ax\|^2$, where $y \in \mathbb{R}^n$, $A \in \mathbb{R}^{n \times p}$ with columns $A_1, \ldots A_p$

Consider minimizing over x_i, with all x_j, $j \neq i$ fixed:

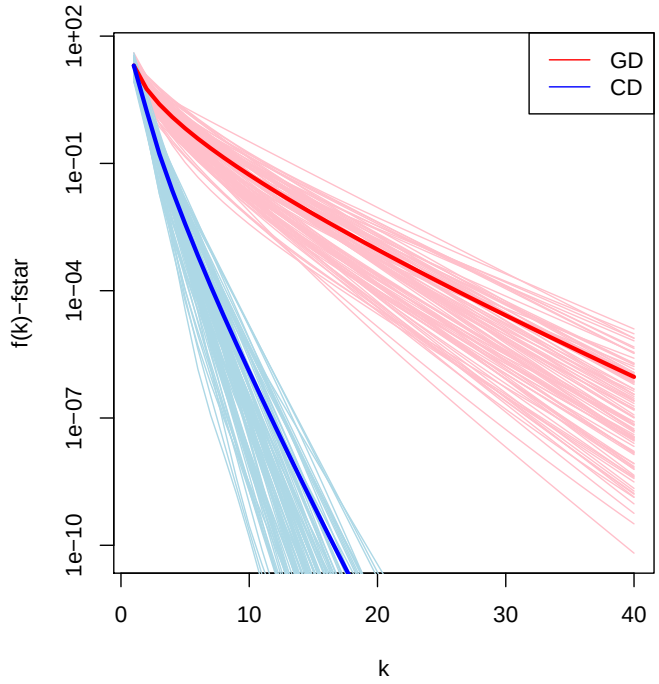
$$0 = \nabla_i f(x) = A_i^T(Ax - y) = A_i^T(A_i x_i + A_{-i} x_{-i} - y)$$

i.e., we take

$$x_i = \frac{A_i^T(y - A_{-i} x_{-i})}{A_i^T A_i}$$

Coordinate descent repeats this update for $i = 1, 2, \ldots, p, 1, 2, \ldots$

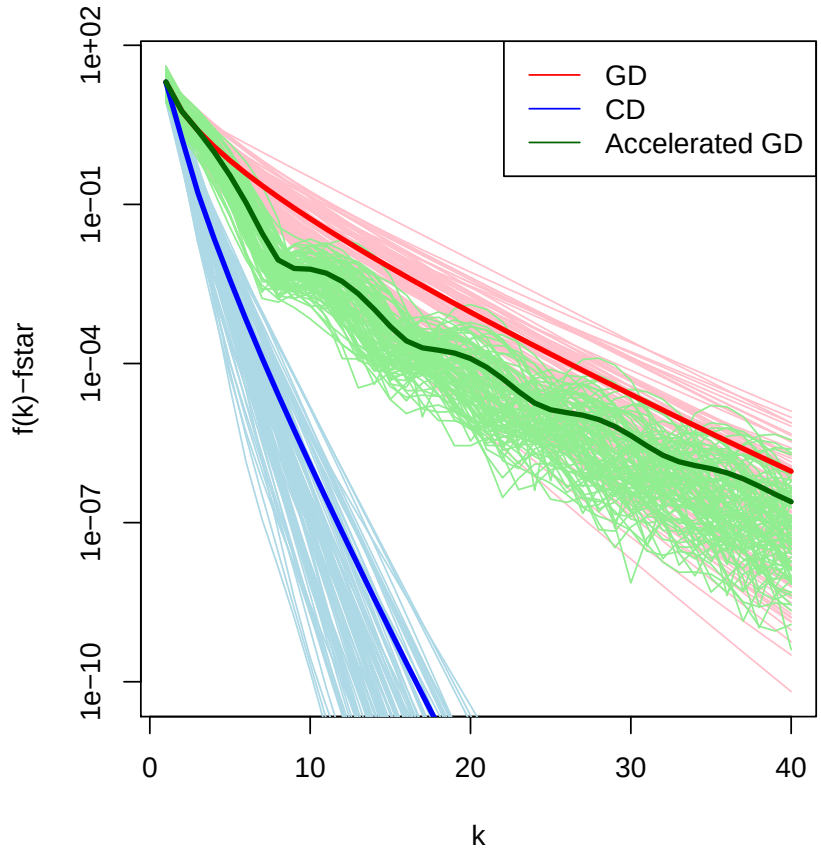
Coordinate descent vs gradient descent for linear regression: 100 instances ($n = 100$, $p = 20$)



Is it fair to compare 1 cycle of coordinate descent to 1 iteration of gradient descent? Yes, if we're clever:

$$x_i = \frac{A_i^T(y - A_{-i}x_{-i})}{A_i^T A_i} = \frac{A_i^T r}{\|A_i\|^2} + x_i^{\text{old}}$$

where $r = y - Ax$. Therefore each coordinate update takes $O(n)$ operations — $O(n)$ to update r, and $O(n)$ to compute $A_i^T r$ — and one cycle requires $O(np)$ operations, just like gradient descent



Same example, but now with accelerated gradient descent for comparison

Is this contradicting the optimality of accelerated gradient descent? I.e., is coordinate descent a first-order method?

No. It uses much more than first-order information

Lasso regression

Consider the lasso problem

$$f(x) = \frac{1}{2}\|y - Ax\|^2 + \lambda\|x\|_1$$

Note that the non-smooth part is separable: $\|x\|_1 = \sum_{i=1}^{p} |x_i|$

Minimizing over x_i, with x_j, $j \neq i$ fixed:

$$0 = A_i^T A_i x_i + A_i^T (A_{-i} x_{-i} - y) + \lambda s_i$$

where $s_i \in \partial |x_i|$. Solution is given by soft-thresholding

$$x_i = S_{\lambda/\|A_i\|^2}\left(\frac{A_i^T(y - A_{-i}x_{-i})}{A_i^T A_i}\right)$$

Repeat this for $i = 1, 2, \ldots p, 1, 2, \ldots$

Box-constrained regression

Consider box-constrainted linear regression

$$\min_{x \in \mathbb{R}^n} \ \frac{1}{2}\|y - Ax\|^2 \ \text{ subject to } \ \|x\|_\infty \leq s$$

Note this fits our framework, as $1\{\|x\|_\infty \leq s\} = \sum_{i=1}^n 1\{|x_i| \leq s\}$

Minimizing over x_i with all x_j, $j \neq i$ fixed: with same basic steps, we get

$$x_i = T_s\left(\frac{A_i^T(y - A_{-i}x_{-i})}{A_i^T A_i}\right)$$

where T_s is the truncating operator:

$$T_s(u) = \begin{cases} s & \text{if } \ u > s \\ u & \text{if } \ -s \leq u \leq s \\ -s & \text{if } \ u < -s \end{cases}$$

Support vector machines

A coordinate descent strategy can be applied to the SVM dual:

$$\min_{\alpha \in \mathbb{R}^n} \frac{1}{2}\alpha^T K \alpha - 1^T \alpha \text{ subject to } y^T \alpha = 0,\ 0 \leq \alpha \leq C1$$

Sequential minimal optimization or SMO (Platt, 1998) is basically blockwise coordinate descent in blocks of 2. Instead of cycling, it chooses the next block greedily

Recall the complementary slackness conditions

$$\alpha_i \cdot \left[(Av)_i - y_i d - (1 - s_i)\right] = 0, \quad i = 1, \ldots n \tag{1}$$

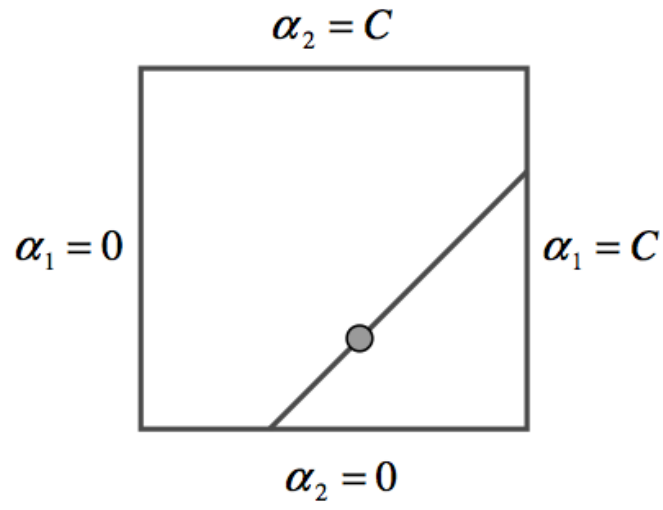
$$(C - \alpha_i) \cdot s_i = 0, \quad i = 1, \ldots n \tag{2}$$

where v, d, s are the primal coefficients, intercept, and slacks, with $v = A^T \alpha$, d computed from (1) using any i such that $0 < \alpha_i < C$, and s computed from (1), (2)

SMO repeats the following two steps:

- Choose α_i, α_j that do not satisfy complementary slackness
- Minimize over α_i, α_j exactly, keeping all other variables fixed

Second step uses equality constraint, reduces to minimizing univariate quadratic over an interval (From Platt, 1998)



First step uses heuristics to choose α_i, α_j greedily

Note this does not meet separability assumptions for convergence from Tseng (2001), and a different treatment is required

Coordinate descent in statistics and ML

History in statistics:

- Idea appeared in Fu (1998), and again in Daubechies et al. (2004), but was inexplicably ignored
- Three papers around 2007, and Friedman et al. (2007) really sparked interest in statistics and ML community

Why is it used?

- Very simple and easy to implement
- Careful implementations can attain state-of-the-art
- Scalable, e.g., don't need to keep data in memory

Some examples: lasso regression, SVMs, lasso GLMs, group lasso, fused lasso (total variation denoising) trend filtering, graphical lasso, regression with nonconvex penalties

Pathwise coordinate descent for lasso

Here is the basic outline for pathwise coordinate descent for lasso, from Friedman et al. (2007), Friedman et al. (2009)

Outer loop (**pathwise** strategy):

- Compute the solution at sequence $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_r$ of tuning parameter values
- For tuning parameter value λ_k, initialize coordinate descent algorithm at the computed solution for λ_{k+1}

Inner loop (**active set** strategy):

- Perform one coordinate cycle (or small number of cycles), and record active set S of coefficients that are nonzero
- Cycle over coefficients in S until convergence
- Check KKT conditions over all coefficients; if not all satisfied, add offending coefficients to S, go back one step

Even if solution is only desired at one value of λ, pathwise strategy ($\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_r = \lambda$) is much faster than directly performing coordinate descent at λ

Active set strategy takes algorithmic advantage of sparsity; e.g., for large problems, coordinate descent for lasso is much faster than it is for ridge regression

With these strategies in place (and a few more tricks), coordinate descent is competitve with fastest algorithms for 1-norm penalized minimization problems

Freely available via **glmnet** package in MATLAB or R (Friedman et al., 2009)

Convergence rates?

Global convergence rates for coordinate descent have not yet been established as they have for first-order methods

Recently Saha et al. (2010) consider minimizing

$$f(x) = g(x) + \lambda \|x\|_1$$

and assume that

- g convex, ∇g Lipschitz with constant $L > 0$, and $I - \nabla g/L$ monotone increasing in each component
- there is z such that $z \geq S_\lambda(z - \nabla g(z))$ or $z \leq S_\lambda(z - \nabla g(z))$ (component-wise)

They show that for coordinate descent starting at $x^{(0)} = z$, and generalized gradient descent starting at $y^{(0)} = z$ (step size $1/L$),

$$f(x^{(k)}) - f(x^\star) \leq f(y^{(k)}) - f(x^\star) \leq \frac{L\|x^{(0)} - x^\star\|^2}{2k}$$

Graphical lasso

Consider a data matrix $A \in \mathbb{R}^{n \times p}$, whose rows $a_1, \ldots a_n \in \mathbb{R}^p$ are independent observations from $N(0, \Sigma)$, with unknown covariance matrix Σ

Want to estimate Σ; normality theory tells us that

$$\Sigma^{-1}_{ij} = 0 \quad \Leftrightarrow \quad A_i, A_j \text{ conditionally independent given } A_\ell,\ \ell \neq i, j$$

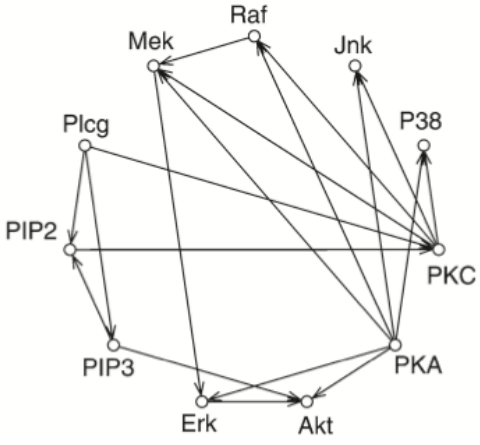
If p is large, we believe above to be true for many i, j, so we want a sparse estimate of Σ^{-1}. We get this by solving **graphical lasso** (Banerjee et al., 2007, Friedman et al., 2007) problem:

$$\min_{\Theta \in \mathbb{R}^{p \times p}} \; -\log\det\Theta + \mathrm{tr}(S\Theta) + \lambda\|\Theta\|_1$$

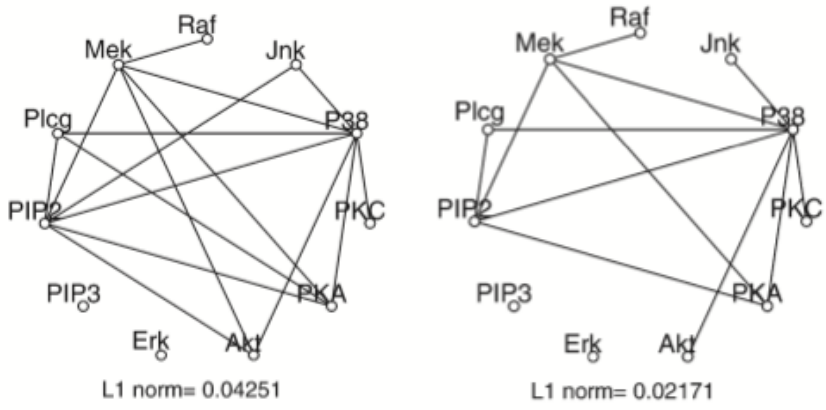
Minimizer $\Theta^\star$ is an estimate for Σ^{-1}. (Note here $S = A^T A/n$ is the empirical covariance matrix, and $\|\Theta\|_1 = \sum_{i,j=1}^p |\Theta_{ij}|$)

Example from Friedman et al. (2007), cell-signaling network:

Believed network

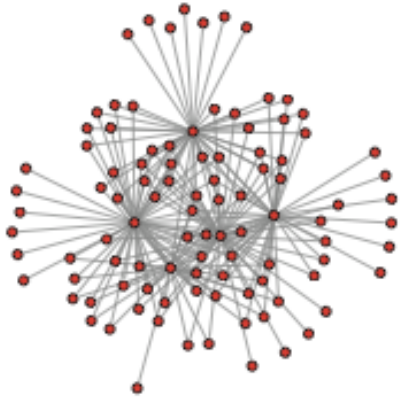


Graphical lasso estimates

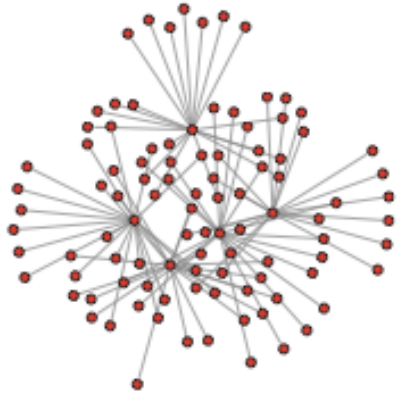


Example from Liu et al. (2010), hub graph simulation:

True graph



Graphical lasso estimate



Graphical lasso KKT conditions (stationarity):

$$-\Theta^{-1} + S + \lambda\Gamma = 0$$

where $\Gamma_{ij} \in \partial|\Theta_{ij}|$. Let $W = \Theta^{-1}$; we will solve in terms of W. Note $W_{ii} = S_{ii} + \lambda$, because $\Theta_{ii} > 0$ at solution. Now partition:

$$W = \begin{bmatrix} W_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \quad \Theta = \begin{bmatrix} \Theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{bmatrix} \quad S = \begin{bmatrix} S_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \quad \Gamma = \begin{bmatrix} \Gamma_{11} & \gamma_{12} \\ \gamma_{21} & \gamma_{22} \end{bmatrix}$$

where $W_{11} \in \mathbb{R}^{(p-1)\times(p-1)}$, $w_{12} \in \mathbb{R}^{(p-1)\times 1}$, and $w_{21} \in \mathbb{R}^{1\times(p-1)}$, $w_{22} \in \mathbb{R}$; same with others

Coordinate descent strategy: solve for w_{12}, the last column of W (note w_{22} is known), with all other columns fixed; then solve for second-to-last column, etc., and cycle around until convergence. (Solve for Θ along the way, so we don't have to invert W to get Θ)

Now consider 12-block of KKT conditions:

$$-w_{12} + s_{12} + \lambda\gamma_{12} = 0$$

Because $\begin{bmatrix} W_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \begin{bmatrix} \Theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix}$, we know that $w_{12} = -W_{11}\theta_{12}/\theta_{22}$. Substituting this into the above,

$$W_{11}\frac{\theta_{12}}{\theta_{22}} + s_{12} + \lambda\gamma_{12} = 0$$

Letting $x = \theta_{12}/\theta_{22}$ and noting that $\theta_{22} > 0$ at solution, this is

$$W_{12}x + s_{12} + \lambda\rho = 0$$

where $\rho \in \partial\|x\|_1$. What does this condition look like?

These are exactly the KKT conditions for

$$\min_{x \in \mathbb{R}^{p-1}} x^T W_{11} x + s_{12}^T x + \lambda \|x\|_1$$

which is (basically) a lasso problem and can be solved quickly via coordinate descent

From x we get $w_{12} = -W_{11}x$, and θ_{12}, θ_{22} are obtained from the identity $\begin{bmatrix} W_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \begin{bmatrix} \Theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix}$

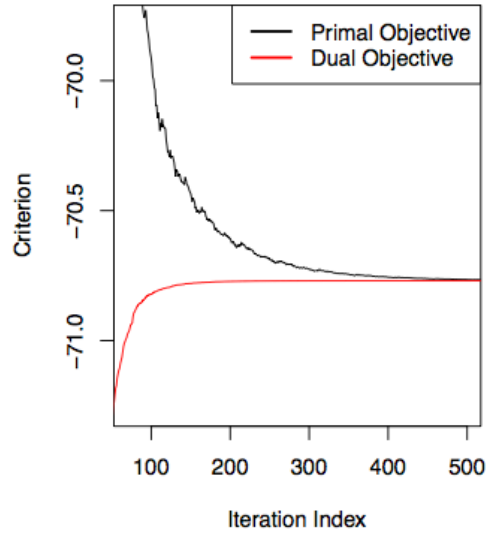
We set $w_{21} = w_{12}^T$, $\theta_{21} = \theta_{12}^T$, and move on to a different column; hence we have reduced the graphical lasso problem to a bunch of sequential lasso problems

This coordinate descent approach for the graphical lasso, usually called **glasso** algorithm (Friedman et al., 2007) is very efficient and scales well

Meanwhile, people have noticed that using glasso algorithm, it can happen that the objective function doesn't decrease monotonically across iterations — is this a bug?

No! The glasso algorithm makes a variable transformation and solves in terms of coordinate blocks of W; note that these are **not coordinate blocks** of original variable Θ, so strictly speaking it is not a coordinate descent algorithm

However, it can be shown that glasso is doing coordinate ascent on the dual problem (Mazumder et al., 2011)



Screening rules for graphical lasso

Graphical lasso computations can be significantly accelerated by using a clever screening rule (this is analogous to the SAFE rules for the lasso)

Mazumder et al. (2011), Witten et al. (2011) examine the KKT conditions:

$$-\Theta^{-1} + S + \lambda\Gamma = 0$$

and conclude that Θ is block diagonal over variables C_1, C_2 if and only if $|S_{ij}| \leq \lambda$ for all $i \in C_1$, $j \in C_2$. Why?

- If Θ is block diagonal, then so is Θ^{-1}, and thus $|S_{ij}| \leq \lambda$ for $i \in C_1$, $j \in C_2$
- If $|S_{ij}| \leq \lambda$ for $i \in C_1$, $j \in C_2$, then the KKT conditions are satisfied with Θ^{-1} block diagonal, so Θ is block diagonal

Exact same idea extends to multiple blocks. Hence group structure in graphical lasso solution is just given by **covariance thresholding**

References

Early coordinate descent references in statistics and ML:

- I. Daubechies and M. Defrise and C. De Mol (2004), *An iterative thresholding algorithm for linear inverse problems with a sparsity constraint*
- J. Friedman and T. Hastie and H. Hoefling and R. Tibshirani (2007), *Pathwise coordinate optimization*
- W. Fu (1998), *Penalized regressions: the bridge versus the lasso*
- T. Wu and K. Lange (2008), *Coordinate descent algorithms for lasso penalized regression*
- A. van der Kooij (2007), *Prediction accuracy and stability of regresssion with optimal scaling transformations*

Applications of coordinate descent:

- O. Banerjee and L. Ghaoui and A d'Aspremont (2007), *Model selection through sparse maximum likelihood estimation*
- J. Friedman and T. Hastie and R. Tibshirani (2007), *Sparse inverse covariance estimation with the graphical lasso*
- J. Friedman and T. Hastie and R. Tibshirani (2009), *Regularization paths for generalized linear models via coordinate descent*
- J. Platt (1998), *Sequential minimal optimization: a fast algorithm for training support vector machines*

Theory for coordinate descent:

- R. Mazumder and J. Friedman and T. Hastie (2011), *SparseNet: coordinate descent with non-convex penalties*
- A. Saka and A. Tewari (2010), *On the finite time convergence of cyclic coordinate descent methods*
- P. Tseng (2001), *Convergence of a block coordinate descent method for nondifferentiable minimization*

More graphical lasso references:

- H. Liu and K. Roeder and L. Wasserman (2010), *Stability approach to regularization selection (StARS) for high dimensional graphical models*
- R. Mazumder and T. Hastie (2011), *The graphical lasso: new insights and alternatives*
- R. Mazumder and T. Hastie (2011), *Exact covariance thresholding into connected components for large-scale graphical Lasso*
- D. Witten and J. Friedman and N. Simon (2011), *New insights and faster computations for the graphical lasso*