Computation & optimization for Lasso - part 2

Luyang Han & Janosch Ott

ETH Zürich

22 October 2018

Computation & optimization for Lasso - part 2

Luyang Han & Janosch Ott

ETH Zürich

22 October 2018

Overview

Overview
1. Coordinate Descent
2. A Simulation Study
3. Least Angle Regression
4. Digression: Duality
5. ADMM
6. Minor-Max Algorithms
7. Alternating Minimizations
8. Screening Rules

Digression: Duality in optimization

In various section, I came across terms like "dual" and "dual problem"

Primal	
Optimize	$\min f(x)$
Constraints	$g_i(x) \leq 0, h_j(x) = 0, x \in X$
Function	$L(x, \lambda, \mu) := f(x) + \sum_i \lambda_i g_i(x) + \sum_j \mu_j h_j(x)$
Dual	
Function	$q(\lambda, \mu) = \inf_{x \in X} L(x, \lambda, \mu)$
Constraints	$\lambda \geq 0$
Optimize	$\max q(\lambda, \mu)$

Why though? - **Dual problem is always convex!**

Primal	
Optimize	$\min f(x)$
Constraints	$g_i(x) \leq 0, h_j(x) = 0, x \in X$
Function	$L(x, \lambda, \mu) := f(x) + \sum_i \lambda_i g_i(x) + \sum_j \mu_j h_j(x)$
Dual	
Function	$q(\lambda, \mu) = \inf_{x \in X} L(x, \lambda, \mu)$
Constraints	$\lambda \geq 0$
Optimize	$\max q(\lambda, \mu)$

Why though? - **Dual problem is always convex!**

$x \in X$ for e.g. solutions in a cone or integer solutions
Terms: Primal problem, Lagrange function with dual variables/Lagrange-multipliers, dual function, dual problem
Dual problem is always convex! - I don't know much about optimization yet, but they really like convexity.
The second advantage is that all local optima are global optima. That allows local search algorithms to guarantee optimal solutions. And local search is often faster.
(Convexity confers two advantages. The first is that, in a constrained problem, a convex feasible region makes it easier to ensure that you do not generate infeasible solutions while searching for an optimum.)

Alternating Direction Method of Multipliers (ADMM)

Problem

$$\underset{\beta\in\mathbb{R}^m,\theta\in\mathbb{R}^n}{\text{minimize}}\, f(\beta)+g(\theta) \quad \text{subject to } \mathbf{A}\beta+\mathbf{B}\theta=c$$

Lagrangian

$$f(\beta)+g(\theta)+\rho||\mathbf{A}\beta+\mathbf{B}\theta-c||_2^2$$

Augmented Lagrangian

$$L_\rho(\beta,\theta,\mu):=f(\beta)+g(\theta)+\langle\mu,\mathbf{A}\beta+\mathbf{B}\theta-c\rangle+\frac{\rho}{2}||\mathbf{A}\beta+\mathbf{B}\theta-c||_2^2$$

Alternating Direction Method of Multipliers (ADMM)
Problem
$\underset{\beta\in\mathbb{R}^m,\theta\in\mathbb{R}^n}{\text{minimize}}\, f(\beta)+g(\theta)$ subject to $\mathbf{A}\beta+\mathbf{B}\theta=c$
Lagrangian
$f(\beta)+g(\theta)+\rho||\mathbf{A}\beta+\mathbf{B}\theta-c||_2^2$
Augmented Lagrangian
$L_\rho(\beta,\theta,\mu):=f(\beta)+g(\theta)+\langle\mu,\mathbf{A}\beta+\mathbf{B}\theta-c\rangle+\frac{\rho}{2}||\mathbf{A}\beta+\mathbf{B}\theta-c||_2^2$

Method of Multipliers b/c ρ und μ
Augmented: scalar product with μ gets added

Dual Ascent Step

Alternating Direction Method of Multipliers

$$\beta^{t+1} = \underset{\beta \in \mathbb{R}^m}{\arg\min}\, L_\rho(\beta, \theta^t, \mu^t)$$

$$\theta^{t+1} = \underset{\theta \in \mathbb{R}^m}{\arg\min}\, L_\rho(\beta^{t+1}, \theta, \mu^t)$$

$$\mu^{t+1} = \mu^t + \rho(\mathbf{A}\beta^{t+1} + \mathbf{B}\theta^{t+1} - c)$$

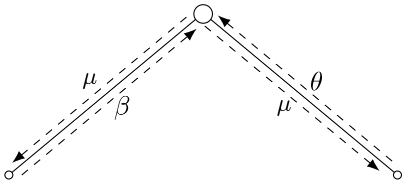
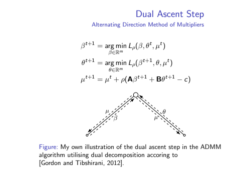


Figure: My own illustration of the dual ascent step in the ADMM algorithm utilising dual decomposition accoring to [Gordon and Tibshirani, 2012].



first two steps is why it is called alternating direction ... cause once we do it in the β and once we do it in the θ direction
last step is called a dual variable update, this dual has nothing to do with two, but is connected to what is called a dual problem
dual ascent step, we are working in the dual problem as "min L", thus convex problem, thus "dual decomposition" into subproblems which is possible by "18-dual-uses.pdf",p. 22,

think of it as only the last line, sending μ to the updaters for β and θ

ADMM - Why?

- convex problems with nondifferentiable constraints
- blockwise computation
  - sample blocks
  - feature blocks

ADMM - Why?
- convex problems with nondifferentiable constraints
- blockwise computation
  - sample blocks
  - feature blocks

Detailsfor blockwise computation in Exercise 5.12.

ADMM for the Lasso

Problem in Lagrangian form

$$\underset{\beta\in\mathbb{R}^p,\theta\in\mathbb{R}^p}{\text{minimize}} \left\{ \frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \|\theta\|_1 \right\} \quad \text{such that } \beta - \theta = 0$$

Update

$$\begin{aligned} \beta^{t+1} &= (\mathbf{X}^T\mathbf{X} + \rho\mathbf{I})^{-1}(\mathbf{X}^T\mathbf{y} + \rho\theta^t - \mu^t) \\ \theta^{t+1} &= \mathcal{S}_{\lambda/\rho}(\beta^{t+1} + \mu^t/\rho) \\ \mu^{t+1} &= \mu^t + \rho(\beta^{t+1} - \theta^{t+1}) \end{aligned}$$

where $\mathcal{S}_{\lambda/\rho}(z) = \text{sign}(z)(|z| - \frac{\lambda}{\rho})_+$.

ADMM for the Lasso

Problem in Lagrangian form

$$\underset{\beta\in\mathbb{R}^p,\theta\in\mathbb{R}^p}{\text{minimize}} \left\{ \frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \|\theta\|_1 \right\} \quad \text{such that } \beta - \theta = 0$$

Update

$$\begin{aligned} \beta^{t+1} &= (\mathbf{X}^T\mathbf{X} + \rho\mathbf{I})^{-1}(\mathbf{X}^T\mathbf{y} + \rho\theta^t - \mu^t) \\ \theta^{t+1} &= \mathcal{S}_{\lambda/\rho}(\beta^{t+1} + \mu^t/\rho) \\ \mu^{t+1} &= \mu^t + \rho(\beta^{t+1} - \theta^{t+1}) \end{aligned}$$

where $\mathcal{S}_{\lambda/\rho}(z) = \text{sign}(z)(|z| - \frac{\lambda}{\rho})_+$.

Computational cost: Initially $\mathcal{O}(p^3)$, which is a lot, for the SVD(singular value decomposition of $\mathbf{X}$), after that comparable to coordinate descent or composite gradient from earlier

Minorization-Maximization Algorithms (MMA)

- Problem: minimize $f(\beta)$ over $\beta \in \mathbb{R}^p$ for f possibly non-convex
- Introduce additional variable θ
- Use θ to majorize (bound from above) the objective function to be minimized

Majorization-Minimization Algorithms work analoguosly.

Minorization-Maximization Algorithms (MMA)

- Problem: minimize $f(\beta)$ over $\beta \in \mathbb{R}^p$ for f possibly non-convex
- Introduce additional variable θ
- Use θ to majorize (bound from above) the objective function to be minimized

Majorization-Minimization Algorithms work analoguosly.

MMA visually

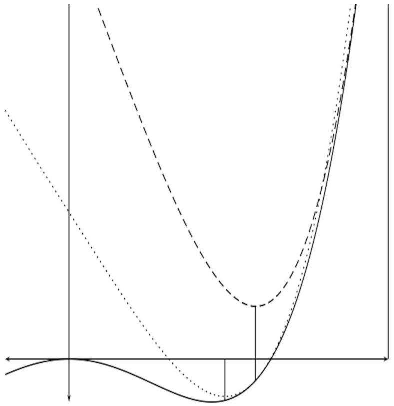


Figure: Figure from [de Leeuw, 2015]

MMA visually

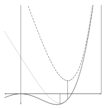


Figure: Figure from [de Leeuw, 2015]

MMA analytically I

Def. $\Psi : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ majorizes f at $\beta \in \mathbb{R}^p$ if

$$\forall \theta \in \mathbb{R}^p \quad \Psi(\beta, \theta) \geq f(\beta)$$

with equality for $\theta = \beta$.

Minor-Maxxalgorithm

- initialize β^0
- update with $\beta^{t+1} = \arg\min_{\beta \in \mathbb{R}^p} \Psi(\beta, \beta^t)$

MMA analytically I

Def. $\Psi : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$ majorizes f at $\beta \in \mathbb{R}^p$ if

$$\forall \theta \in \mathbb{R}^p \quad \Psi(\beta, \theta) \geq f(\beta)$$

with equality for $\theta = \beta$.

Minor-Maxxalgorithm

- initialize β^0
- update with $\beta^{t+1} = \arg\min_{\beta \in \mathbb{R}^p} \Psi(\beta, \beta^t)$

MMA analytically II

This scheme generates a sequence of β's for which the cost $f(\beta^t)$ is nonincreasing, because

$$f(\beta^t) \overset{(i)}{=} \Psi(\beta^t, \beta^t) \overset{(ii)}{\geq} \Psi(\beta^{t+1}, \beta^t) \overset{(iii)}{\geq} f(\beta^{t+1})$$

where

(i) & (iii) Definiton of majorize

(ii) β^{t+1} is a minimizer of $\beta \mapsto \Psi(\beta, \beta^t)$

MMA analytically II

This scheme generates a sequence of β's for which the cost $f(\beta^t)$ is nonincreasing, because

$$f(\beta^t) \overset{(i)}{=} \Psi(\beta^t, \beta^t) \overset{(ii)}{\geq} \Psi(\beta^{t+1}, \beta^t) \overset{(iii)}{\geq} f(\beta^{t+1})$$

where

(i) & (iii) Definiton of majorize

(ii) β^{t+1} is a minimizer of $\beta \mapsto \Psi(\beta, \beta^t)$

for inequalities: show previous slide

Biconvexity

Let's consider an example . . .

$f(\alpha, \beta) = (1 - \alpha\beta)^2$

Biconvexity
Let's consider an example . . .
$f(\alpha, \beta) = (1 - \alpha\beta)^2$

Mathematica: `3D plot (1-xy)^2, x in [-2,2], y in [-2,2]`
The formula is a link.

Biconvexity

Let's consider an example ...

$f(\alpha,\beta) = (1-\alpha\beta)^2$

Def. A function $f(\alpha,\beta) : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ is biconvex, if for each $\alpha \in \mathbb{R}^m$ the function $\alpha \mapsto f(\alpha,\beta)$ is convex and for each $\beta \in \mathbb{R}^n$ the function $\beta \mapsto f(\alpha,\beta)$ is convex. Analoguosly, a set $\mathcal{C} \subseteq \mathcal{A} \times \mathcal{B}$, for $\mathcal{A},\mathcal{B}$ convex sets, is called <u>biconvex</u>, if it is convex

Biconvexity

Let's consider an example ...

$f(\alpha,\beta) = (1-\alpha\beta)^2$

Def. A function $f(\alpha,\beta) : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ is biconvex, if for each $\alpha \in \mathbb{R}^m$ the function $\alpha \mapsto f(\alpha,\beta)$ is convex and for each $\beta \in \mathbb{R}^n$ the function $\beta \mapsto f(\alpha,\beta)$ is convex. Analoguosly, a set $\mathcal{C} \subseteq \mathcal{A} \times \mathcal{B}$, for $\mathcal{A},\mathcal{B}$ convex sets, is called biconvex, if it is convex

Mathematica: `3D plot (1-xy)^2, x in [-2,2], y in [-2,2]`
The formula is a link.

Alternate Convex Search

Block coordinate descent applied to α and β blocks

1. Initialize (α^0, β^0) at some point in the biconvex set to minimize over
2. For $t = 0, 1, 2, \dots$
   (i) Fix $\beta = \beta^t$ and update $\alpha^{t+1} \in \underset{\alpha \in \mathcal{C}_{\beta^t}}{\arg\min}\, f(\alpha, \beta^t)$
   (ii) Fix $\alpha = \alpha^{t+1}$ and update $\beta^{t+1} \in \underset{\alpha \in \mathcal{C}_{\alpha^{t+1}}}{\arg\min}\, f(\alpha^{t+1}, \beta)$

For a function bounded from below, the algorithm converges to a partial optimum (i.e. as biconvexity, only optimal in one coordinate if the other coordinate is fixed).

Alternate Convex Search

Block coordinate descent applied to α and β blocks

1. Initialize (α^0, β^0) at some point in the biconvex set to minimize over
2. For $t = 0, 1, 2, \dots$
   (i) Fix $\beta = \beta^t$ and update $\alpha^{t+1} \in \underset{\alpha \in \mathcal{C}_{\beta^t}}{\arg\min}\, f(\alpha, \beta^t)$
   (ii) Fix $\alpha = \alpha^{t+1}$ and update $\beta^{t+1} \in \underset{\alpha \in \mathcal{C}_{\alpha^{t+1}}}{\arg\min}\, f(\alpha^{t+1}, \beta)$

For a function bounded from below, the algorithm converges to a partial optimum (i.e. as biconvexity, only optimal in one coordinate if the other coordinate is fixed).

Screening Rules

Dual Polytope Projection (DPP)

Suppose we want to calculate a lasso solution at $\lambda < \lambda_{\text{max}}$. The DPP rule discards the j^{th} variable if

$$\left|\mathbf{x}_j^T \mathbf{y}\right| < \lambda_{\text{max}} - \|\mathbf{x}_j\|_2 \|\mathbf{y}\|_2 \frac{\lambda_{\text{max}} - \lambda}{\lambda}$$

Sequential DPP rule

Suppose we have the lasso solution $\hat{\beta}(\lambda')$ at λ' and want to screen variables for solutions at $\lambda < \lambda'$. We discard the j^{th} variable if

$$\left|\mathbf{x}_j^T (\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda'))\right| < \lambda' - \|\mathbf{x}_j\|_2 \|\mathbf{y}\|_2 \frac{\lambda_{\text{max}} - \lambda}{\lambda}$$

Dual Polytope Projection (DPP)

Suppose we want to calculate a lasso solution at $\lambda < \lambda_{\text{max}}$. The DPP rule discards the j^{th} variable if

$$\left|\mathbf{x}_j^T \mathbf{y}\right| < \lambda_{\text{max}} - \|\mathbf{x}_j\|_2 \|\mathbf{y}\|_2 \frac{\lambda_{\text{max}} - \lambda}{\lambda}$$

Sequential DPP rule

Suppose we have the lasso solution $\hat{\beta}(\lambda')$ at λ' and want to screen variables for solutions at $\lambda < \lambda'$. We discard the j^{th} variable if

$$\left|\mathbf{x}_j^T (\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda'))\right| < \lambda' - \|\mathbf{x}_j\|_2 \|\mathbf{y}\|_2 \frac{\lambda_{\text{max}} - \lambda}{\lambda}$$

Global Strong Rule

Suppose we want to calculate a lasso solution at $\lambda < \lambda_{\text{max}}$. The global strong rule discards the j^{th} variable if

$$\left|\mathbf{x}_j^T \mathbf{y}\right| < \lambda - (\lambda_{\text{max}} - \lambda) = 2\lambda - \lambda_{\text{max}}$$

Sequential Strong Rule

Suppose we have the lasso solution $\hat{\beta}(\lambda')$ at λ' and want to screen variables for solutions at $\lambda < \lambda'$. We discard the j^{th} variable if

$$\left|\mathbf{x}_j^T (\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda'))\right| < 2\lambda - \lambda'$$

Global Strong Rule

Suppose we want to calculate a lasso solution at $\lambda < \lambda_{\text{max}}$. The global strong rule discards the j^{th} variable if

$$\left|\mathbf{x}_j^T \mathbf{y}\right| < \lambda - (\lambda_{\text{max}} - \lambda) = 2\lambda - \lambda_{\text{max}}$$

Sequential Strong Rule

Suppose we have the lasso solution $\hat{\beta}(\lambda')$ at λ' and want to screen variables for solutions at $\lambda < \lambda'$. We discard the j^{th} variable if

$$\left|\mathbf{x}_j^T (\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda'))\right| < 2\lambda - \lambda'$$

References

Trevor Hastie, Robert Tibshirani, and Martin Wainwright (2015)
Statistical learning with sparsity: the Lasso and generalizations
CRC Press; Boca Raton, FL

Jan De Leeuw (2015)
Block Relaxation Methods in Statistics
doi.org/10.13140/RG.2.1.3101.9607 (last accessed: 02.10.18)

Geoff Gordon and Ryan Tibshirani (2012)
Uses of Duality
https://www.cs.cmu.edu/~ggordon/10725-F12/slides/18-dual-uses.pdf (last accessed: 14.10.18)

References

Trevor Hastie, Robert Tibshirani, and Martin Wainwright (2015)
Statistical learning with sparsity: the Lasso and generalizations
CRC Press; Boca Raton, FL

Jan De Leeuw (2015)
Block Relaxation Methods in Statistics
doi.org/10.13140/RG.2.1.3101.9607 (last accessed: 02.10.18)

Geoff Gordon and Ryan Tibshirani (2012)
Uses of Duality
https://www.cs.cmu.edu/~ggordon/10725-F12/slides/18-dual-uses.pdf (last accessed: 14.10.18)

Comments . . .
Questions . . .
Suggestions . . .

Comments . . .
Questions . . .
Suggestions . . .

That's it.
Thanks for listening.

Fill out your feedback sheets!

That's it.
Thanks for listening.

Fill out your feedback sheets!