Convex Optimization in Theory and Practice: Solving the Lasso Problem

Danting Wu, Péter Lelkes

Content

Part I

- *glmnet* library in *R*

Part II

- Convex Optimization Problems
- Lagrangian Duality
- Karush-Kuhn-Tucker Conditions
- Subgradient and Subdifferential

Part III

- Introduction to Descent Methods
- Proximal Gradient Method
- Solving the Lasso Problem
- Solving Constrained Problems

Part I

- *glmnet* library in *R*

Computational Details and *glmnet*

Penalties: The general model of *glmnet* looks like

$$\min_{\beta,\beta_0}\left\{-\frac{1}{N}l(y;\beta,\beta_0)+\lambda\sum_{j=1}^{p}\gamma_j(1-\alpha)\beta_j^2+\alpha\mid\beta_j\mid\right\}$$

and the penalties are λ which controls overall complexity, $\alpha \in [0,1]$ the elastic net parameter that mix up ridge and lasso ($\alpha = 1$ is lasso and $\alpha = 0$ is ridge). γ_j is the modifier that determines how much we want from a predictor, see the example after.

Family: Picking the loss-function and the associated model, including gaussian, multinomial, poisson and cox. By default we use the gaussian model. We will present another example of using binomial.

Objective function of gaussian

$$\min_{(\beta_0,\beta)\in\mathbb{R}^{p+1}} \frac{1}{2N}\sum_{i=1}^{N}(y_i-\beta_0-x_i^T\beta)^2+\lambda[(1-\alpha)\|\beta\|_2^2/2+\alpha\|\beta\|_1]$$

Objective function of Binomial(for logistic regression)

$$\min_{(\beta_0,\beta)\in\mathbb{R}^{p+1}} -\left[\frac{1}{N}\sum_{i=1}^{N} y_i\cdot(\beta_0+x_i^T\beta)-\log\left(1+e^{\beta_0+x_i^T\beta}\right)\right]+\lambda[(1-\alpha)\|\beta\|_2^2/2+\alpha\|\beta\|$$

Coefficient bounds: Gives upper and lower bounds for coefficients.

Weights: How much we value certain parts of the observations. Weights of value 2 means we give double importance to corresponding observations.

Part II

- Convex Optimization Problems
- Lagrangian Duality
- Karush-Kuhn-Tucker Conditions
- Subgradient and Subdifferential

Optimization Models

We present an overview of basic optimization concepts and algorithms for convex problems.

We focus primarily on first order methods (methods that only utilizes first order derivatives).

An important class of optimization problems involves convex cost functions and convex constraints.

Definition

A set $\mathcal{C} \subseteq \mathbb{R}^p$ **is convex** if for all $\beta, \beta^{'} \in \mathcal{C}$ and all scalars $s \in [0, 1]$, all vectors of the form $\beta(s) = s\beta + (1 - s)\beta^{'}$ also belong to $\mathcal{C}$.

A function $f : \mathbb{R}^p \to \mathbb{R}$ **is convex** means that for any tow vectors β, β' in the domain of f and any scalar $s \in (0, 1)$, we have

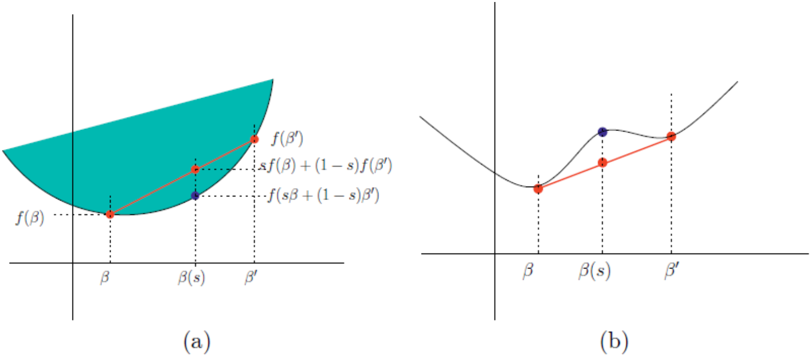
$$f(\beta(s)) = f(s\beta + (1 - s)\beta') \leq sf(\beta) + (1 - s)f(\beta')$$

Question

What’s the geometric meaning of this?

Question

What's the geometric meaning of this?



Hastie, T., Tibshirani, R., Wainwright, M. (2015). Statistical learning with sparsity: The lasso and generalizations. Boca Raton: CRC Press, Taylor Francis Group.

It guarantees that a convex function cannot have any local minima that are not also globally minimal.

Rewriting Constraints

Consider the constrained optimization problem

$$\min_{\beta \in \mathcal{C} \subset \mathbb{R}^p} f(\beta)$$

where $f : \mathbb{R}^p \to \mathbb{R}$ is a convex objective function and $\mathcal{C}$ a convex constraint set. A necessary and sufficient condition for a vector $\beta^* \in \mathcal{C}$ to be a global optimum is

$$\langle \nabla f(\beta^*), \beta - \beta^* \rangle \ \geq \ 0 \quad \forall \beta \in \mathcal{C}$$

We will prove sufficiency of this condition by considering the following inequality (a special case is when $\mathcal{C} = \mathbb{R}^p$)

$$f(\beta) \ \geq \ f(\beta^*) \ + \langle \nabla f(\beta^*), \beta - \beta^* \rangle \ \geq \ f(\beta^*)$$

We can further simplify the convex constraint set $\mathcal{C}$ into a convex sublevel set of of some convex constraint function $g(\beta)$.
It follows from the definition of convexity that the set $\{\beta \in \mathbb{R}^p \mid g(\beta) \leq 0\}$ is always a convex set.
The optimization problem turns into

$$\min_{\beta \in \mathbb{R}^p} f(\beta) \quad \text{such that} \quad g_j(\beta) \leq 0 \quad \text{for } j = 1, \cdots, m$$

were g_j are convex functions that express the constraints.

The Lagrangian

The Lagrangian $L : \mathbb{R}^p \times \mathbb{R}_+^m \to \mathbb{R}$ is defined by

$$L(\beta; \lambda) = f(\beta) + \sum_{j=1}^{m} \lambda_j g_j(\beta)$$

where the non-negative weights λ_j are the **Lagrange multipliers**.

They impose a penalty whenever the constraint $g_j(\beta) \leq 0$ is violated.

We can rewrite the original problem as

$$\sup_{\lambda \geq 0} L(\beta; \lambda) = \begin{cases} f(\beta) \text{ , if } g_j(\beta) \leq 0 \ \forall j \\ +\infty \text{ , otherwise} \end{cases}$$

and thus the optimal value f^* of the optimization problem would be $f^* = \inf_{\beta \in \mathbb{R}^p} \sup_{\lambda \geq 0} L(\beta; \lambda)$

Lagrangian Duality

Assume strong duality holds, the problem we are solving is equivalent to

$$f^* = \min_{\beta \in \mathbb{R}^p} \max_{\lambda \geq 0} L(\beta; \lambda) = \max_{\lambda \geq 0} \min_{\beta \in \mathbb{R}^p} L(\beta; \lambda) =: \max_{\lambda \geq 0} d(\lambda)$$

where $d(\lambda) = \min_{\beta \in \mathbb{R}^p} L(\beta; \lambda)$ is known as the **dual function**. Solving it is known as the **dual problem**.

Since $d(\lambda)$ is a pointwise minimum of affine functions ($L(x; \lambda)$ is affine, i.e. linear, in λ), it is a concave function. Therefore, maximizing $d(\lambda)$ over λ is a convex optimization problem, i.e. an easy problem.

Lagrangian duality works because our Lagrangian, which uses multipliers λ, acts as a lower bound of the primal problem, which uses infinite step functions I.

Primal problem:

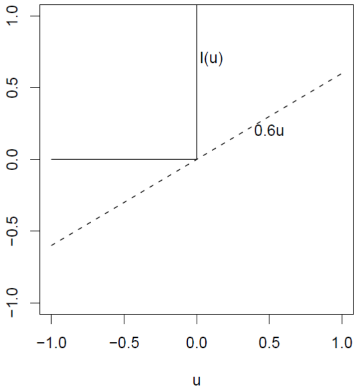
$$J(\beta) = \begin{cases} f(\beta) \text{ , if } g_j(\beta) \leq 0 \; \forall j \\ +\infty \text{ , otherwise} \end{cases}$$

$$= f(\beta) + \sum_i I\left[g_i(\beta)\right]$$

where $I[u]$ is a infinite step function

$$I(u) = \begin{cases} 0 \text{ , if } u \leq 0 \\ +\infty \text{ , otherwise} \end{cases}$$

When the strong duality condition is satisfied (aka when our optimization problem is convex and a strictly feasible point exists), the primal and dual problem share the same optimal point.



Lagrangian Duality for Dummies - Stanford University. (n.d.). Retrieved from https://cs.stanford.edu/people/davidknowles/lagrangian$_d$*uality.pdf*

Geometric visualization

As a result, any optimum β^*, in addition to satisfying the feasibility constraints $g_j(\beta^*) \leq 0$, must also be a zero-gradient point of the Lagrangian, namely:

$$0 = \nabla_\beta L(\beta^*; \lambda^*) = \nabla f(\beta^*) + \sum_{j=1}^{m} \lambda_j^* \nabla g_j(\beta^*)$$

When there is only a single constraint function g, this condition reduces to

$$\nabla f(\beta^*) = -\lambda^* \nabla g(\beta^*)$$

What's the geometric meaning of this?

Geometric visualization

As a result, any optimum β^*, in addition to satisfying the feasibility constraints $g_j(\beta^*) \leq 0$, must also be a zero-gradient point of the Lagrangian, namely:

$$0 = \nabla_\beta L(\beta^*; \lambda^*) = \nabla f(\beta^*) + \sum_{j=1}^{m} \lambda_j^* \nabla g_j(\beta^*)$$

When there is only a single constraint function g, this condition reduces to

$$\nabla f(\beta^*) = -\lambda^* \nabla g(\beta^*)$$

What's the geometric meaning of this?
The gradient of the objective function and constraint function must be parallel (and opposite).

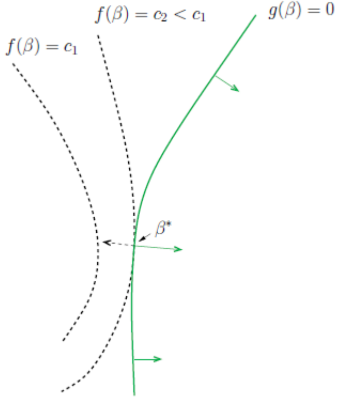


Figure 5.2 *Illustration of the method of Lagrange multipliers. We are minimizing a function f subject to a single constraint $g(\beta) \leq 0$. At an optimal solution β^*, the normal vector $\nabla f(\beta^*)$ to the level sets of the cost function f points in the opposite direction to the normal vector $\nabla g(\beta^*)$ of the constraint boundary $g(\beta) = 0$. Consequently, up to first order, the value of $f(\beta^*)$ cannot be decreased by moving along the contour $g(\beta) = 0$.*

Hastie, T., Tibshirani, R., Wainwright, M. (2015). Statistical learning with sparsity: The lasso and generalizations. Boca Raton: CRC Press, Taylor Francis Group.

Karush Kuhn Tucker Conditions

The KKT conditions are the equivalent conditions for the global minimum of a constrained convex optimization problem. The first condition, which is also known as the **Primal Feasibility Conditions** are already defined in setting

$$g_j(\beta^*) \leq 0 \quad \text{and} \quad \lambda_j^* \geq 0 \quad \forall j$$

We've also seen the second condition

$$0 = \nabla_\beta L(\beta^*; \lambda^*) = \nabla f(\beta^*) + \sum_{j=1}^{m} \lambda_j^* \nabla g_j(\beta^*)$$

which is the **Lagrangian Condition**

Complementary Slackness

By definition we have

$$\min_{\beta \in \mathbb{R}^p} L(\beta; \lambda^*) = f(\beta^*) \;\leq\; f(\beta^*) + \sum_{j=1}^{m} \lambda_j^* g_j(\beta^*) \;\leq\; f(\beta^*)$$

where the second part holds because $\sum_{j=1}^{m} \lambda_j^* g_j(\beta^*) \;\leq\; 0$.
We see that $\sum_{j=1}^{m} \lambda_j^* g_j(\beta^*) \; = 0$. With $\lambda_j^* \geq 0$ and $g_j(\beta^*) \leq 0$ we have the third condition, known as **Complementary Slackness**

$$\lambda_j^* g_j(\beta^*) \; = 0 \qquad \forall j$$

Non-differentiable Functions and Sub-gradients

In practise, we often have convex but non-differentiable cost functions. For instance, the ℓ_1 norm $g(\beta) = \sum_{j=1}^{p} |\beta_j|$ is not differetiable at any $\beta_j = 0$.

We have to generalize the notion of gradient, to the so-called **subgradients**.

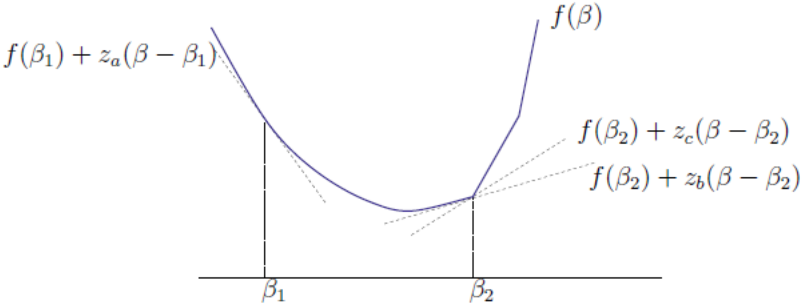
The basic idea is to utilize the fact that for differentiable convex functions, the first-order tangent approximation always provides a lower bound

Definition

Given a convex function $f : \mathbb{R}^p \to \mathbb{R}$, a vector $z \in \mathbb{R}^p$ is said to be a **subgradient** of f at β if

$$f(\beta') \geq f(\beta) \ + \ \langle z, \beta' - \beta \rangle \ \ \forall \beta' \in \mathbb{R}^p$$

The set of all subgradients of f at β is called the **subdifferential**, denoted by $\partial f(\beta)$


$f(\beta)$
$f(\beta_1) + z_a(\beta - \beta_1)$
$f(\beta_2) + z_c(\beta - \beta_2)$
$f(\beta_2) + z_b(\beta - \beta_2)$
β_1
β_2

Hastie T., Tibshirani, R., Wainwright, M. (2015). Statistical learning with sparsity: The lasso and generalizations. Boca Raton: CRC Press, Taylor Francis Group.

At the point β_1, the function is differentiable and hence there is only one subgradient, namely, $f'(\beta_1)$. At the point β_2, it is not differentiable, and there are multiple subgradients; each one specifies a tangent plane that provides a lower bound on f.

Take $f(\beta) = |\beta|$, we have

$$\partial f(\beta) = \begin{cases} \{+1\} \text{ , if } \beta > 0 \\ \{-1\} \text{ , if } \beta < 0 \\ [-1, +1] \text{ , if } \beta = 0 \end{cases}$$

We write $z \in \text{sign}(\beta)$ to mean that z belongs to subdifferential of the absolute value function at β.

Generalized KKT Condition

Now, going back to our original problem. How do we make sense of the zero-gradient Lagrangian condition? (Hint: naively thinking...)

Generalized KKT Condition

Now, going back to our original problem. How do we make sense of the zero-gradient Lagrangian condition? (Hint: naively thinking...)

$$0 \in \partial f(\beta^*) + \sum_{j=1}^{m} \lambda_j^* \partial g_j(\beta^*)$$

Part III

- Introduction to Descent Methods
- Proximal Gradient Method
- Solving the Lasso Problem
- Solving Constrained Problems

Unconstrained Problem

Assumption: $f : \mathbb{R}^p \to \mathbb{R}$ convex, differentiable function. It achieves its (unique!) global minimum.

In this case:

$$\min_{\beta \in \mathbb{R}^p} f(\beta) = \beta^* \iff \bigtriangledown f(\beta^*) = 0$$

To find β^*, we use **Descent Methods:**

- $\beta_0 \in \mathbb{R}^p$ arbitrary
- assume that the iteration is at some $\beta_t \in \mathbb{R}^p$
- $\triangle_t \in \mathbb{R}^p$ such that $\langle - \bigtriangledown f(\beta_t), \triangle_t \rangle > 0$ (the angle of $- \bigtriangledown f(\beta_t)$ and $\triangle_t$ is $< 90°$)
- $s_t \in \mathbb{R}$ called **step size** (See later!)
- **iteration step:** $\beta_{t+1} = \beta_t + s_t \triangle_t$

Examples of Descent Methods

(1) Gradient Descent: $\triangle_t = -\bigtriangledown f(\beta_t)$

(2) Diagonally-scaled Grad. Desc.: $\triangle_t = -D_t^{-1} \bigtriangledown f(\beta_t)$, where D_t is diagonal with $(D_t)_{ii} > 0$ for $\forall i$

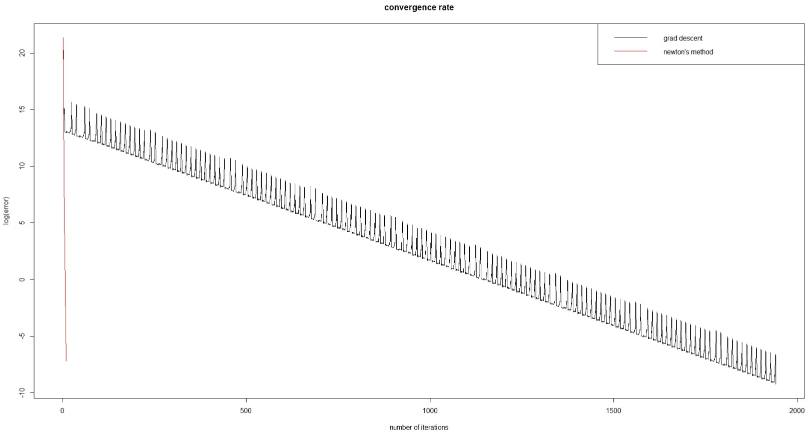
D_t^{-1} modulates the speed of the descent in each direction individually.

- 1st order methods (i.e. only use $\bigtriangledown f(\beta_t)$)

(3) Newton's Method: $\triangle_t = -(\bigtriangledown^2 f(\beta_t))^{-1} \bigtriangledown f(\beta_t)$

- 2nd order method (i.e. uses $\bigtriangledown^2 f(\beta_t)$)
- **pro:** quadratic rate of convergence
- **con:** computationally expensive

R Simulation: Comparing Grad. Desc. and Newton’s Method in case of the RSS function



See also the R code!

How to determine the step size?

Constant step size does not guarantee convergence!

Limited Minimization Rule: $s_t = \arg\min_{s \in [0,1]} f(\beta_t + s\triangle_t)$

- **con:** creates another minimization problem

Backtracking (Armijo) Rule: Take $\alpha \in (0, 0.5)$, $\gamma \in (0, 1)$ and $s = 1$. Update $s \leftarrow \gamma s$ until

$$f(\beta_t + s\triangle_t) \leq f(\beta_t) + \alpha s \langle \bigtriangledown f(\beta_t), \triangle_t \rangle$$

Can be proven that both guarantee convergence of the descent method!

Proximal Gradient Method

Take the following iteration ($\beta_0 \in \mathbb{R}^p$ arbitrary):

$$\beta_{t+1} = \arg\min_{\beta \in \mathbb{R}^p} \{ f(\beta_t) + \langle \bigtriangledown f(\beta_t), \beta - \beta_t \rangle + \frac{1}{2s_t} \|\beta - \beta_t\|_2^2 \}$$

The solution is $\beta_{t+1} = \beta_t - s_t \bigtriangledown f(\beta_t)$ (**gradient descent step**).

Can be proven by setting the first derivative to 0.

Proximal Gradient Method

Take a more general class of object functions: $f = g + h$, where

- $g : \mathbb{R}^p \rightarrow \mathbb{R}$ **convex** and **differentiable**
- $h : \mathbb{R}^p \rightarrow \mathbb{R}$ **convex** and **non-diff.**

E.g. object function of the Lasso problem!

Modify the above iteration to find the minimum of this class of functions:

$$\beta_{t+1} = \underset{\beta \in \mathbb{R}^p}{\arg\min} \{ g(\beta_t) + \langle \bigtriangledown g(\beta_t), \beta - \beta_t \rangle + \frac{1}{2s_t} \|\beta - \beta_t\|_2^2 + h(\beta) \}$$

Proximal Gradient Method

Proximal Map:

- $h : \mathbb{R}^p \rightarrow \mathbb{R}$
- $prox_h(z) : \mathbb{R}^p \rightarrow \mathbb{R}^p$

$$prox_h(z) := \arg\min_{\beta \in \mathbb{R}^p} \{ \frac{1}{2} \|z - \beta\|_2^2 + h(\beta) \}$$

Claim: Take $f = g + h$ as on the previous slide. Then

$$prox_{s_t h}(\beta_t - s_t \bigtriangledown g(\beta_t)) =$$

$$\arg\min_{\beta \in \mathbb{R}^p} \{ g(\beta_t) + \langle \bigtriangledown g(\beta_t), \beta - \beta_t \rangle + \frac{1}{2s_t} \|\beta - \beta_t\|_2^2 + h(\beta) \}$$

Corollary: Once we know the proximal map, we can calculate the modified iteration step.

Proximal Gradient Method

Proof:

$$prox_{s_t h}(\beta_t - s_t \bigtriangledown g(\beta_t)) =$$

$$\underset{\beta \in \mathbb{R}_p}{\arg\min} \{ \frac{1}{2} \|\beta_t - s_t \bigtriangledown g(\beta_t) - \beta\|_2^2 + s_t h(\beta) \} =$$

$$\underset{\beta \in \mathbb{R}_p}{\arg\min} \{ \frac{1}{2s_t} (\|\beta - \beta_t\|_2^2 + \|s_t \bigtriangledown g(\beta_t)\|_2^2 + 2s_t \langle \bigtriangledown g(\beta_t), \beta - \beta_t \rangle) + h(\beta) \} =$$

$$\underset{\beta \in \mathbb{R}_p}{\arg\min} \{ \frac{1}{2s_t} \|\beta - \beta_t\|_2^2 + \langle \bigtriangledown g(\beta_t), \beta - \beta_t \rangle + h(\beta) \} =$$

$$\underset{\beta \in \mathbb{R}_p}{\arg\min} \{ \frac{1}{2s_t} \|\beta - \beta_t\|_2^2 + \langle \bigtriangledown g(\beta_t), \beta - \beta_t \rangle + h(\beta) + g(\beta_t) \}$$

Proximal Gradient Method and Lasso

In case of the **Lasso problem** $f = g + h$ is the following:

- $g(\beta) = \frac{1}{2N}\|Y - X\beta\|_2^2$
- $h(\beta) = \lambda\|\beta\|_1$

Cheap to compute the proximal map!

$$prox_{s\lambda\|\beta\|_1}(z) = \arg\min_{\beta \in \mathbb{R}^p}\{\frac{1}{2}\|z - \beta\|_2^2 + s\lambda\|\beta\|_1\} =$$

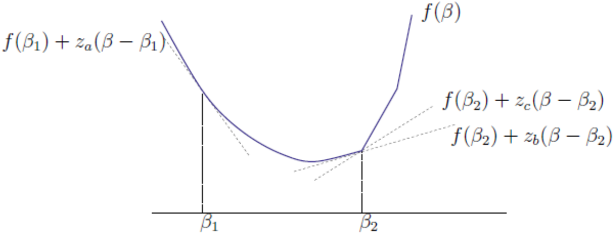
$$\sum_{i=1}^{N} \arg\min_{\beta_i \in \mathbb{R}}\{\frac{1}{2}((z_i - \beta_i)_2^2 + s\lambda|\beta_i|)\}$$

Enough to compute the optimal β_i in terms of z_i.

Proximal Gradient Method and Lasso

We want to find the global minimum of $\frac{1}{2}(z_i - \beta_i)^2 + s\lambda|\beta_i|$ (in terms of z_i).

Since $|\beta_i|$ is convex and non-differentiable at 0, we use the concept of the **subdifferential**.


$f(\beta)$
$f(\beta_1) + z_a(\beta - \beta_1)$
$f(\beta_2) + z_c(\beta - \beta_2)$
$f(\beta_2) + z_b(\beta - \beta_2)$
β_1
β_2

Source: Hastie, T. (2015). Statistical Learning with Sparsity.

Three properties of the subdifferential

Take $f : \mathbb{R}^p \to \mathbb{R}$ convex function.

(1) If $f(x)$ is differentiable at x, then $\partial f(x) = \bigtriangledown f(x)$.

(2) $\partial(f(x) + g(x)) = \partial f(x) + \partial g(x)$

(3) x^* is the minimum of $f \iff 0 \in \partial f(x^*)$

Proximal Gradient Method and Lasso

We want to find the global minimum of $\frac{1}{2}(z_i - \beta_i)^2 + s\lambda|\beta_i|$ (in terms of z_i).

Using the above properties, that means we look for β_i^* (in terms of z_i) such that

$$0 \in \partial(\frac{1}{2}(z_i - \beta_i^*)^2 + s\lambda|\beta_i^*|) = -(z_i - \beta_i^*) + s\lambda\partial(|\beta_i^*|)$$

Subdifferential of $|x|$

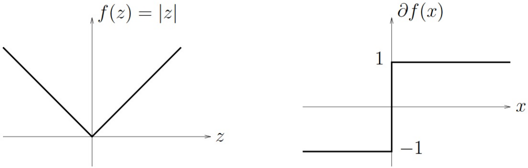


Figure 3: The absolute value function (left), and its subdifferential $\partial f(x)$ as a function of x (right).

$$\partial(|x|) = \begin{cases} \{1\}, & 0 < x \\ \{[-1, 1]\}, & x = 0 \\ \{-1\}, & x < 0 \end{cases}$$

Proximal Gradient Method and Lasso

$$0 = -(z_i - \beta_i^*) + s\lambda\partial(|\beta_i^*|)$$

$$s\lambda\partial(|\beta_i^*|) = z_i - \beta_i^*$$

If $\beta_i^* > 0$, then $\partial(|\beta_i^*|) = \{1\}$. Therefore $\beta_i^* = z_i - s\lambda$ (and $z_i > s\lambda$).

Applying this on all three cases, we get:

$$\beta_i^*(z_i) = \begin{cases} z_i - s\lambda, & z_i > s\lambda \\ 0, & z_i \in [-s\lambda, s\lambda] \\ z_i + s\lambda, & z_i < -s\lambda \end{cases} \iff \beta_i^* = S_{s\lambda}(z_i)$$

Soft-thresholding operator: $S_{s\lambda}(z_i) := sign(z_i)(|z_i| - s\lambda)_+$

Proximal Gradient Method and Lasso

Soft-thresholding operator: $S_{s\lambda}(z_i) := sign(z_i)(|z_i| - s\lambda)_+$

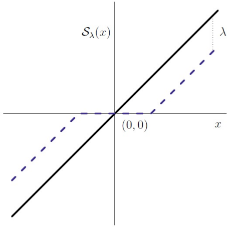


Figure 2.4 *Soft thresholding function* $\mathcal{S}_\lambda(x) = \mathrm{sign}(x)\,(|x| - \lambda)_+$ *is shown in blue (broken lines), along with the* 45° *line in black.*

Source: Hastie, T. (2015). Statistical Learning with Sparsity.

Proximal Gradient Method and Lasso

The optimal $\beta^* \in \mathbb{R}^p$ (in terms of z) is simply a vector with $\beta_i^* = S_{s\lambda}(z_i)$ in each coordinate. I.e.

$$prox_{s\lambda\|\beta\|_1}(z) = \sum_{i=1}^{N} \arg\min_{\beta\in\mathbb{R}^p}\{\frac{1}{2s}((z_i - \beta_i)_2^2 + \lambda|\beta_i|)\} = S'_{s\lambda}(z)$$

where $S'_{s\lambda}(z) : \mathbb{R}^p \to \mathbb{R}^p$ and $(S'_{s\lambda}(z))_i = S_{s\lambda}(z_i)$

Proximal Gradient Method and Lasso

In case of the **Lasso problem** $f = g + h$ is the following:

- $g(\beta) = \frac{1}{2N}\|Y - X\beta\|_2^2$
- $h(\beta) = \lambda\|\beta\|_1$

Gradient descent step of g:

$$\beta_t - s_t \frac{1}{N} X^T(Y - X\beta_t)$$

Proximal gradient step of Lasso:

$$\beta_{t+1} = S'_{s_t\lambda}(\beta_t - s_t \frac{1}{N} X^T(Y - X\beta_t))$$

Cyclic Coordinate Descent

Our knowledge about the soft-thresholding operator gives another algorithm to solve the Lasso problem, the **cyclic coordinate descent**.

Assume we have **one predictor** and it is standardized (i.e. $\sum_{i=1}^{N} x_i^2 = 1$). Then the solution to the Lasso problem is

$$\underset{\beta\in\mathbb{R}}{\arg\min}\{\frac{1}{2N}\sum_{i=1}^{N}(y_i - \beta x_i)^2 + \lambda|\beta|\} = S_\lambda(\frac{1}{N}\langle y, x\rangle)$$

Same calculations with the subdifferential as previously!

Cyclic Coordinate Descent

In case of **multiple predictors**, take an arbitrary but fixed order (e.g. $k = 1, 2, ..., p$). Cycle repeatedly through k's in the following manner:

$$\arg\min_{\beta_k \in \mathbb{R}} \{ \frac{1}{2N} \sum_{i=1}^{N} (y_i - \sum_{j \neq k} \beta_j x_{ij} - \beta_k x_{ik})^2 + \lambda \sum_{j \neq k} |\beta_j| + \lambda |\beta_k| \} =$$

$$S_\lambda(\frac{1}{N} \langle y - \sum_{j \neq k} \beta_j x^{(j)}, x^{(k)} \rangle) := S_\lambda(\frac{1}{N} \langle r^{(k)}, x^{(k)} \rangle)$$

$r^{(k)}$ is the so-called partial residual.

In every cycle update β_k:

$$\beta_k \leftarrow S_\lambda(\frac{1}{N} \langle r^{(k)}, x^{(k)} \rangle)$$

Proximal Gradient Method

- moves all the coefficients together
- efficient when multiplying by X and X^T is fast

Cyclic Coordinate Descent

- exploits **sparsity** (See the next lecture!)
- no step size optimization

Not clear which one is more efficient. Comparison via simulation in the next lecture!

Constraint and Proximal Gradient Method

Proximal Gradient Method can also be used to solve **constrained minimization problem** of a convex, differentiable function ($g : \mathbb{R}^p \to \mathbb{R}$) on a C convex set.

Define the following function:

$$I_C(\beta) = \begin{cases} 0 & \textit{if}\, \beta \in C \text{ convex set} \\ +\infty & \text{otherwise} \end{cases}$$

Constraint and Proximal Gradient Method

$$I_C(\beta) = \begin{cases} 0 & if \, \beta \in C \text{ convex set} \\ +\infty & \text{otherwise} \end{cases}$$

Claim: Take g and I_C. Then

$$prox_{I_C}(\beta_t - s_t \bigtriangledown g(\beta_t)) =$$

$$\arg\min_{\beta \in \mathbb{R}^p} \{ g(\beta_t) + \langle \bigtriangledown g(\beta_t), \beta - \beta_t \rangle + \frac{1}{2s_t} \|\beta - \beta^t\|_2^2 + I_C(\beta) \} =$$

$$\arg\min_{\beta \in C} \{ g(\beta_t) + \langle \bigtriangledown g(\beta_t), \beta - \beta_t \rangle + \frac{1}{2s_t} \|\beta - \beta_t\|_2^2 \}$$

Constraint and Proximal Gradient Method

Therefore we have the analogous iterative solution but to the **constrained** problem. Take an arbitrary $\beta_0 \in \mathbb{R}^p$ and

$$\beta_{t+1} = prox_{I_C}(\beta_t - s_t \bigtriangledown g(\beta_t))$$

The proximal map in this case simply reduces to:

$$prox_{I_C}(z) = \arg\min_{\beta \in \mathbb{R}^p}\{\frac{1}{2}\|z - \beta\|_2^2 + I_C(\beta)\} = \arg\min_{\beta \in C}\{\|z - \beta\|_2^2\}$$

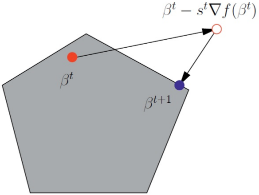
Follows from the definition of the proximal map and I_C.

Then $prox_{I_C}$ is simply the orthogonal projection of z to C.

Constraint and Proximal Gradient Method

Using the **Claim** and our knowledge of $prox_{I_C}$, we get that an iteration step has two parts:

1. taking a **gradient descent step**
2. orthogonally **projecting** the result on C



Source: Hastie, T. (2015). Statistical Learning with Sparsity.

Convergence of the Proximal Gradient Method

Applying the Proximal Gradient Method on the introduced $f = g + h$ function class, we have the following **convergence results**.

(1) **sublinear con.:** If $\| \bigtriangledown g(\beta_1) - \bigtriangledown g(\beta_2)\|_2^2 \leq L\|\beta_1 - \beta_2\|$ for $\forall \beta_1, \beta_2 \in \mathbb{R}^p$. Then with a constant step size $s \in (0, \frac{1}{L}]$, there exists $C \in \mathbb{R}$ that:

$$f(\beta_t) - f(\beta^*) \leq \frac{C}{t+1}\|\beta_t - \beta^*\|_2^2 \text{ for } \forall t = 1, 2, ...$$

(2) **geometric con.:** If the differentiable component g is **strongly convex** then there exists a $\kappa \in (0, 1)$ that:

$$f(\beta_t) - f(\beta^*) \leq C\kappa^t \text{ for } \forall t = 1, 2, ...$$

Summary

Part I

- Functions and parameters of the *glmnet* library

Part II

- Introduction of the Langrangian form of constrained convex optimization problems
- Utilizing Lagrangian duality to find optimum
- Characterisation of the optimum by the KKT conditions
- Generalization of KKT conditions to non-diff. function by the concept of subdifferential

Summary

Part III

- Solving unconstrained problems with different descent methods
- Applying the proximal gradient method on more general object functions
- Comparing the solution of Lasso by proximal gradient and cyclic coordinate descent
- Applying the proximal gradient method on constrained convex problems
- Convergence conditions of the proximal gradient method

Thank you for your attention!

Questions?