Lasso: Algorithms and Extensions

Yuxin Chen
Princeton University, Spring 2017
Outline

- Proximal operators
- Proximal gradient methods for lasso and its extensions
- Nesterov’s accelerated algorithm
Proximal operators
Gradient descent

\[
\text{minimize}_{\beta \in \mathbb{R}^p} \quad f(\beta)
\]

where \( f(\beta) \) is convex and differentiable

\[
\text{Algorithm 4.1 Gradient descent}
\]

\[\text{for } t = 0, 1, \ldots :\]

\[
\beta^{t+1} = \beta^t - \mu_t \nabla f(\beta^t)
\]

where \( \mu_t \): step size / learning rate
\[ \beta^{t+1} = \arg \min_{\beta} \left\{ f(\beta^t) + \langle \nabla f(\beta^t), \beta - \beta^t \rangle + \frac{1}{2\mu_t} \|\beta - \beta^t\|^2 \right\} \]

- When \( \mu_t \) is small, \( \beta^{t+1} \) tends to stay close to \( \beta^t \)

**A proximal point of view of GD**

- Linear approximation
- Proximal term
If we define the **proximal operator**

\[
\text{prox}_h(b) := \arg\min_\beta \left\{ \frac{1}{2} \|\beta - b\|^2 + h(\beta) \right\}
\]

for any convex function \(h\), then one can write

\[
\beta^{t+1} = \text{prox}_{\mu_t f_t}(\beta^t)
\]

where \(f_t(\beta) := f(\beta_t) + \langle \nabla f(\beta_t), \beta - \beta_t \rangle\)
Why consider proximal operators?

\[ \text{prox}_h(b) := \arg \min_{\beta} \left\{ \frac{1}{2} \|\beta - b\|^2 + h(\beta) \right\} \]

- It is well-defined under very general conditions (including nonsmooth convex functions)
- The operator can be evaluated efficiently for many widely used functions (in particular, regularizers)
- This abstraction is conceptually and mathematically simple, and covers many well-known optimization algorithms
Example: characteristic functions

If \( h \) is characteristic function

\[
h(\beta) = \begin{cases} 
0, & \text{if } \beta \in C \\
\infty, & \text{else}
\end{cases}
\]

then

\[
\text{prox}_h(b) = \arg \min_{\beta \in C} \| \beta - b \|_2 \quad \text{(Euclidean projection)}
\]
Example: $\ell_1$ norm

- If $h(\beta) = \|\beta\|_1$, then

$$\text{prox}_{\lambda h}(b) = \psi_{st}(b; \lambda)$$

where soft-thresholding $\psi_{st}(\cdot)$ is applied in an entry-wise manner.
Example: $\ell_2$ norm

\[
\text{prox}_h(b) := \arg\min_{\beta} \left\{ \frac{1}{2} \|\beta - b\|^2 + h(\beta) \right\}
\]

- If $h(\beta) = \|\beta\|$, then

\[
\text{prox}_{\lambda h}(b) = \left(1 - \frac{\lambda}{\|b\|}\right)_+ b
\]

where $a_+ := \max\{a, 0\}$. This is called block soft thresholding.
Example: log barrier

\[
\text{prox}_h(b) := \arg \min_{\beta} \left\{ \frac{1}{2} \|\beta - b\|^2 + h(\beta) \right\}
\]

- If \( h(\beta) = -\sum_{i=1}^{p} \log \beta_i \), then

\[
\left( \text{prox}_{\lambda h}(b) \right)_i = \frac{b_i + \sqrt{b_i^2 + 4\lambda}}{2}
\]
Nonexpansiveness of proximal operators

Recall that when \( h(\beta) = \begin{cases} 0, & \text{if } \beta \in \mathcal{C} \\ \infty, & \text{else} \end{cases} \), \( \text{prox}_h(\beta) \) is Euclidean projection \( \mathcal{P}_\mathcal{C} \) onto \( \mathcal{C} \), which is nonexpansive:

\[
\| \mathcal{P}_\mathcal{C}(\beta^1) - \mathcal{P}_\mathcal{C}(\beta^2) \| \leq \| \beta^1 - \beta^2 \|
\]
Nonexpansiveness of proximal operators

Nonexpansiveness is a property for general $\text{prox}_h(\cdot)$

Fact 4.1 (Nonexpansiveness)

\[
\|\text{prox}_h(\beta^1) - \text{prox}_h(\beta^2)\| \leq \|\beta^1 - \beta^2\|
\]

- In some sense, proximal operator behaves like projection
Proof of nonexpansiveness

Let \( z^1 = \text{prox}_h(\beta^1) \) and \( z^2 = \text{prox}_h(\beta^2) \). Subgradient characterizations of \( z^1 \) and \( z^2 \) read

\[
\beta^1 - z^1 \in \partial h(z^1) \quad \text{and} \quad \beta^2 - z^2 \in \partial h(z^2)
\]

The claim would follow if

\[
(\beta^1 - \beta^2) ^\top (z^1 - z^2) \geq \|z^1 - z^2\|^2 \quad \text{(together with Cauchy-Schwarz)}
\]

\[
\iff \quad (\beta^1 - z^1 - \beta^2 + z^2) ^\top (z^1 - z^2) \geq 0
\]

\[
\iff \quad \begin{cases} 
    h(z^2) \geq h(z^1) + \langle \beta^1 - z^1, z^2 - z^1 \rangle \\
    \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{\( \in \partial h(z^1) \)} \\
    h(z^1) \geq h(z^2) + \langle \beta^2 - z^2, z^1 - z^2 \rangle \\
    \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{\( \in \partial h(z^2) \)}
\end{cases}
\]
Proximal gradient methods
Optimizing composite functions

$$\text{(Lasso) \ minimize}_{\beta \in \mathbb{R}^p} \quad \underbrace{\frac{1}{2} \|X\beta - y\|^2}_f(\beta) + \underbrace{\lambda \|\beta\|_1}_g(\beta) = f(\beta) + g(\beta)$$

where $f(\beta)$ is differentiable, and $g(\beta)$ is non-smooth

- Since $g(\beta)$ is non-differentiable, we cannot run vanilla gradient descent
Proximal gradient methods

One strategy: replace $f(\beta)$ with linear approximation, and compute the proximal solution

$$\beta^{t+1} = \arg\min_{\beta} \left\{ f(\beta^t) + \langle \nabla f(\beta^t), \beta - \beta^t \rangle + g(\beta) + \frac{1}{2\mu_t} \| \beta - \beta^t \|^2 \right\}$$

The optimality condition reads

$$0 \in \nabla f(\beta^t) + \partial g(\beta^{t+1}) + \frac{1}{\mu_t} \left( \beta^{t+1} - \beta^t \right)$$

which is equivalent to optimality condition of

$$\beta^{t+1} = \arg\min_{\beta} \left\{ g(\beta) + \frac{1}{2\mu_t} \| \beta - (\beta^t - \mu_t \nabla f(\beta^t)) \|^2 \right\}$$

$$= \text{prox}_{\mu_t g} \left( \beta^t - \mu_t \nabla f(\beta^t) \right)$$
Alternate between gradient updates on $f$ and proximal minimization on $g$

**Algorithm 4.2** Proximal gradient methods

```
for $t = 0, 1, \cdots$

$$\beta^{t+1} = \text{prox}_{\mu_t g} \left( \beta^t - \mu_t \nabla f(\beta^t) \right)$$
```

where $\mu_t$: step size / learning rate
Projected gradient methods

When $g(\beta) = \begin{cases} 0, & \text{if } \beta \in \mathcal{C} \\
\infty, & \text{else} \end{cases}$ is characteristic function:

$$
\beta^{t+1} = \mathcal{P}_\mathcal{C} \left( \beta^t - \mu_t \nabla f(\beta^t) \right)
$$

$$
:= \arg\min_{\beta \in \mathcal{C}} \| \beta - (\beta^t - \mu_t \nabla f(\beta^t)) \|
$$

This is a first-order method to solve the constrained optimization

$$
\text{minimize}_{\beta} \quad f(\beta)
$$

s.t. \quad $\beta \in \mathcal{C}$
Proximal gradient methods for lasso

For lasso: \( f(\beta) = \frac{1}{2} \| X\beta - y \|^2 \) and \( g(\beta) = \lambda \| \beta \|_1 \),

\[
\text{prox}_g(\beta) = \arg \min_b \left\{ \frac{1}{2} \| \beta - b \|^2 + \lambda \| b \|_1 \right\} \\
= \psi_{st}(\beta; \lambda)
\]

\[\Rightarrow \beta^{t+1} = \psi_{st} \left( \beta^t - \mu_t X^\top (X\beta^t - y); \mu_t \lambda \right) \]

(iterative soft thresholding)
Proximal gradient methods for group lasso

Sometimes variables have a natural group structure, and it is desirable to set all variables within a group to be zero (or nonzero) simultaneously

\[
\text{(group lasso)} \quad \frac{1}{2} \| X\beta - y \|^2 + \lambda \sum_{j=1}^{k} \| \beta_j \|
\]

\[= f(\beta) + g(\beta) \]

where \( \beta_j \in \mathbb{R}^{p/k} \) and \( \beta = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} \).

\[\text{prox}_g(\beta) = \psi_{\text{bst}} (\beta; \lambda) := \left[ \left( 1 - \frac{\lambda}{\| \beta_j \|} \right) + \beta_j \right]_{1 \leq j \leq k} \]

\[\Rightarrow \beta^{t+1} = \psi_{\text{bst}} \left( \beta^t - \mu_t X^\top (X\beta^t - y); \mu_t \lambda \right)\]
Proximal gradient methods for elastic net

Lasso does not handle highly correlated variables well: if there is a group of highly correlated variables, lasso often picks one from the group and ignores the rest.

- Sometimes we make a compromise between lasso and $\ell_2$ penalties (elastic net)

$$
\frac{1}{2} \| X\beta - y \|^2 + \lambda \left\{ \| \beta \|_1 + \frac{\gamma}{2} \| \beta \|_2^2 \right\}
$$

$$
:= f(\beta)
$$

$$
:= g(\beta)
$$

$$
\text{prox}_{\lambda g}(\beta) = \frac{1}{1 + \lambda \gamma} \psi_{st}(\beta; \lambda)
$$

$$
\implies \beta^{t+1} = \frac{1}{1 + \mu_t \lambda \gamma} \psi_{st} \left( \beta^t - \mu_t X^\top (X\beta^t - y); \mu_t \lambda \right)
$$

- soft thresholding followed by multiplicative shrinkage
Interpretation: majorization-minimization

\[ f_{\mu t}(\beta, \beta^t) := f(\beta^t) + \langle \nabla f(\beta^t), \beta - \beta^t \rangle + \frac{1}{2\mu t} \|\beta - \beta^t\|^2 \]

linearization

trust region penalty

majorizes \( f(\beta) \) if \( 0 < \mu t < \frac{1}{L} \), where \( L \) is Lipschitz constant\(^1\) of \( \nabla f(\cdot) \)

Proximal gradient descent is a majorization-minimization algorithm

\[ \beta^{t+1} = \arg\min_{\beta} \left\{ f_{\mu t}(\beta, \beta^t) + g(\beta) \right\} \]

minimization

majorization

\(^1\)This means \( \|\nabla f(\beta) - \nabla f(b)\| \leq L\|\beta - b\| \) for all \( \beta \) and \( b \)
Convergence rate of proximal gradient methods

Theorem 4.2 (fixed step size; Nesterov ’07)

Suppose $g$ is convex, and $f$ is differentiable and convex whose gradient has Lipschitz constant $L$. If $\mu_t \equiv \mu \in (0, 1/L)$, then

$$f(\beta^t) + g(\beta^t) - \min_{\beta} \{ f(\beta) + g(\beta) \} \leq O(\frac{1}{t})$$

- Step size requires an upper bound on $L$
- May prefer backtracking line search to fixed step size
- **Question**: can we further improve the convergence rate?
Nesterov’s accelerated gradient methods
Nesterov’s accelerated method

Problem of gradient descent: zigzagging

Nesterov’s idea: include a momentum term to avoid overshooting
Nesterov’s accelerated method

**Nesterov’s idea:** include a momentum term to avoid overshooting

\[
\beta^t = \text{prox}_{\mu t} \left( b^{t-1} - \mu t \nabla f \left( b^{t-1} \right) \right)
\]

\[
b^t = \beta^t + \alpha_t \left( \beta^t - \beta^{t-1} \right)
\]

- momentum term

- A simple *(but mysterious)* choice of extrapolation parameter
  \[
  \alpha_t = \frac{t - 1}{t + 2}
  \]

- Fixed size \( \mu_t \equiv \mu \in (0, 1/L) \) or backtracking line search

- Same computational cost per iteration as proximal gradient
Theorem 4.3 (Nesterov ’83, Nesterov ’07)

Suppose $f$ is differentiable and convex and $g$ is convex. If one takes
\[ \alpha_t = \frac{t-1}{t+2} \]
and a fixed step size $\mu_t \equiv \mu \in (0, 1/L)$, then

\[
f(\beta^t) + g(\beta^t) - \min_{\beta} \{f(\beta) + g(\beta)\} \leq O \left( \frac{1}{t^2} \right)
\]

In general, this rate cannot be improved if one only uses gradient information!
Numerical experiments (for lasso)

Figure credit: Hastie, Tibshirani, & Wainwright ’15
Reference


