Computing the lasso solution

Lasso is often used in high-dimensional problems. Cross-validation involves solving many lasso problems. (Note: the solutions can be computed in parallel with a computer cluster when working with large problems.) How can we efficiently compute the lasso solution? Recall: the lasso objective

$$\|y - X\beta\|_2^2 + \alpha \|\beta\|_1$$

is not differentiable everywhere on $\mathbb{R}^p$. Many strategies exist for solving minimizing the lasso objective function,

We will look at two approaches: coordinate descent, and least-angle regression (LARS).

Coordinate descent optimization

Objective: Minimize a function $f : \mathbb{R}^n \to \mathbb{R}$.

Strategy: Minimize each coordinate separately while cycling through the coordinates.

\[
\begin{align*}
    x_1^{(k+1)} &= \text{argmin}_x f(x, x_2^{(k)}, x_3^{(k)}, \ldots, x_p^{(k)}) \\
    x_2^{(k+1)} &= \text{argmin}_x f(x_1^{(k+1)}, x, x_3^{(k)}, \ldots, x_p^{(k)}) \\
    x_3^{(k+1)} &= \text{argmin}_x f(x_1^{(k+1)}, x_2^{(k+1)}, x, x_4^{(k)}, \ldots, x_p^{(k)}) \\
    &\vdots \\
    x_p^{(k+1)} &= \text{argmin}_x f(x_1^{(k+1)}, x_2^{(k+1)}, \ldots, x_{p-1}^{(k+1)}, x).
\end{align*}
\]

Neglected technique in the past that gained popularity recently. Can be very efficient when the coordinate-wise problems are easy to solve (e.g. if they admit a closed-form solution).
Convergence

Does this procedure always converge to an extreme point of the objective in general? NO!


Convergence (cont.)

Does coordinate descent work for the lasso? YES! We exploit the fact that the non-differential part of the objective is separable. **Theorem:** (See Tseng, 2001). Suppose

\[ f(x_1, \ldots, x_p) = f_0(x_1, \ldots, x_p) + \sum_{i=1}^{p} f_i(x_i) \quad (f \in \mathbb{R}^p) \]

satisfies

- \( f_0 : \mathbb{R}^p \to \mathbb{R} \) is convex and continuously differentiable.
- \( f_i : \mathbb{R} \to \mathbb{R} \) is convex \((i = 1, \ldots, p)\).
- The set \( X^0 := \{ x \in \mathbb{R}^p : f(x) \leq f(x^0) \} \) is compact.
- \( f \) is continuous on \( X^0 \).

Then every limit point of the sequence \((x^{(k)})_{k \geq 1}\) generated by cyclic coordinate descent converges to a global minimum of \( f \).

Lasso: individual step

Fix \( x_j \) for \( j \neq i \). We need to solve:

\[
\min_{x_i} \frac{1}{2} \| y - Ax \|_2^2 + \alpha \sum_{k=1}^{p} |x_k| \\
= \min_{x_i} \frac{1}{2} \sum_{l=1}^{n} \left( y_l - \sum_{m=1}^{p} a_{lm}x_m \right)^2 + \alpha \sum_{k=1}^{p} |x_k|.
\]

Now,

\[
\frac{\partial}{\partial x_i} \frac{1}{2} \sum_{l=1}^{n} \left( y_l - \sum_{m=1}^{p} a_{lm}x_m \right)^2 = \sum_{l=1}^{n} \left( y_l - \sum_{m=1}^{p} a_{lm}x_m \right) \times (-a_{li})
\]

\[
= A_i^T (Ax - y)
\]

\[
= A_i^T (A_{-i} x_{-i} - y) + A_i^T A_i x_i.
\]

What about the non-differential part?

Digression: subdifferential calculus

Suppose \( f \) is convex and differentiable. Then

\[
f(y) \geq f(x) + \nabla f(x)^T (y - x).
\]

We say that \( g \) is a subgradient of \( f \) at \( x \) if

\[
f(y) \geq f(x) + g^T (y - x) \quad \forall y.
\]
We define
\[ \partial f(x) := \{ \text{all subgradients of } f \text{ at } x \}. \]
- \( \partial f(x) \) is a closed convex set (can be empty).
- \( \partial f(x) = \{ \nabla f(x) \} \) if \( f \) is differentiable at \( x \).
- If \( \partial f(x) = \{ g \} \), then \( f \) is differentiable at \( x \) and \( \nabla f(x) = g \).

**Basic properties:**
- \( \partial (af) = a \partial f \) if \( a > 0 \).
- \( \partial (f_1 + f_2) = \partial f_1 + \partial f_2 \).

**Example:**
\[
\partial f(x) = \begin{cases} 
\{-1\} & \text{if } x < 0 \\
[-1, 1] & \text{if } x = 0 \\
\{1\} & \text{if } x > 0
\end{cases}
\]

Recall: If \( f \) is convex and differentiable, then
\[ f(x^*) = \inf_x f(x) \iff 0 = \nabla f(x^*). \]

**Theorem:** Let \( f \) be a (not necessarily differentiable) convex function. Then
\[ f(x^*) = \inf_x f(x) \iff 0 \in \partial f(x^*). \]

**Proof:**
\[ f(y) \geq f(x^*) + 0 \cdot (y - x^*) \iff 0 \in \partial f(x^*). \]

Despite its simplicity, this is a very powerful and important result.

The function
\[ f(x_i) := \frac{1}{2} \| y - Ax \|_2^2 + \alpha \sum_{k=1}^p |x_k| \]
is convex. Its minimum is obtained if \( 0 \in \partial f(x^*) \).

Let \( g := \frac{\partial}{\partial x_i} \| y - Ax \|_2^2 = A_i^T (A_i x - y) + A_i^T A_i x_i \).

Then,
\[ \partial f(x) = \begin{cases} 
\{g - \alpha\} & \text{if } x_i < 0 \\
[g - \alpha, g + \alpha] & \text{if } x_i = 0 \\
\{g + \alpha\} & \text{if } x_i > 0
\end{cases} \]

Now,
\[ g - \alpha = 0 \iff x_i = \frac{A_i^T (y - A_{-i} x_{-i}) - \alpha}{A_i^T A_i} = g^* + \frac{\alpha}{\|A_i\|_2^2}. \]

This implies \( 0 \in \partial f(x^*) \) if \( x^* = g^* + \frac{\alpha}{\|A_i\|_2^2} < 0 \).
We have
\[-\frac{\alpha}{\|A_i\|_2^2} \leq g^* \leq \frac{\alpha}{\|A_i\|_2^2} \iff -\frac{\alpha}{\|A_i\|_2^2} \leq \frac{A_i^T(y - A_i^*x_{-i})}{A_i^TA_i} \leq \frac{\alpha}{\|A_i\|_2^2} \]
\[\iff -\alpha \leq A_i^T(y - A_i^*x_{-i}) \leq \alpha.\]

If \(x_i = 0\), then \(g = A_i^T(y - A_i^*x_{-i})\) and so \(0 \in [g - \alpha, g + \alpha]\).
We have therefore shown that \(0 \in \partial f(x^*)\) if \(x^* = 0\) and
\[-\frac{\alpha}{\|A_i\|_2^2} \leq g^* \leq \frac{\alpha}{\|A_i\|_2^2}.
\]

Therefore, the minimum of \(f(x)\) is obtained at
\[x^* = \begin{cases} 
  \frac{g^*}{\|A_i\|_2^2} & \text{if } g^* < -\frac{\alpha}{\|A_i\|_2^2} \\
  \frac{g^*}{\|A_i\|_2^2} & \text{if } g^* > \frac{\alpha}{\|A_i\|_2^2} \\
  0 & \text{if } -\frac{\alpha}{\|A_i\|_2^2} \leq g^* \leq \frac{\alpha}{\|A_i\|_2^2}.
\end{cases}\]
In other words,
\[x^* = \eta_{\alpha/\|A_i\|_2^2}^S(g^*) = \eta_{\alpha/\|A_i\|_2^2}^S \left( \frac{A_i^T(y - A_i^*x_{-i})}{A_i^TA_i} \right),\]
where \(\eta_{\epsilon}\) is the soft-thresholding function.

To solve the lasso problem using coordinate descent:
- Pick an initial point \(x\).
- Cycle through the coordinates and perform the updates
  \[x_i \rightarrow \eta_{\alpha/\|A_i\|_2^2}^S \left( \frac{A_i^T(y - A_i^*x_{-i})}{A_i^TA_i} \right),\]
- Continue until convergence (i.e., stop when the coordinates vary less than some threshold).

Exercise: Implement this algorithm in Python.