STRONG RULES FOR EFFICIENT LASSO COMPUTATIONS AND THE BASIL ALGORITHM

NOTES BY PARKER KNIGHT

1. Strong rules for the LASSO

1.1. Subgradients. Let $f : \mathbb{R}^m \to \mathbb{R}$ be convex.

Recall the following first order condition: if f is differentiable, then

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) \quad \forall x, y \in \text{dom}(f)$$

What if f is not differentiable? This motivates the following definition: Call $g \in \mathbb{R}^m$ a *subgradient* of f at x iff

$$f(y) \geq f(x) + g^T (y - x) \quad \forall x, y \in \text{dom}(f)$$

The subdifferential of f at x, denote $\partial f(x)$, is the set of all subgradients. The following facts will be useful:

(1) If f is differentiable at x, then $\partial f(x) = \{\nabla f(x)\}$
(2) For $\alpha_1, \alpha_2 \geq 0$, then $\partial\left[\alpha_1 f_1(x) + \alpha_2 f_2(x)\right] = \alpha_1 \partial f_1(x) + \alpha_2 \partial f_2(x)$
(3) x^* minimizes f iff $0 \in \partial f(x^*)$

where we define set addition as $A + B = \{a + b | a \in A, b \in B\}$.

1.2. The LASSO. Recall the LASSO loss function:

$$Q_\lambda(\beta) = \|Y - X\beta\|_2^2 + \lambda \|\beta\|_1$$

where the LASSO solution $\hat{\beta}(\lambda)$ satisfies

$$Q_\lambda(\hat{\beta}(\lambda)) = \min_\beta Q_\lambda(\beta)$$

Note that $Q_\lambda(.)$ is not differentiable everywhere, but it is convex. So $\hat{\beta}_\lambda$ minimizes Q iff $0 \in \partial Q_\lambda(\hat{\beta}(\lambda))$.

But how do we find ∂Q?

$$\partial Q_\lambda(b) = -X^T(Y - Xb) + \lambda\gamma$$

where $\gamma \in \partial ||b||_1$. For $g(x) = |x|$, we have

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

since g is differentiable when $x \neq 0$, and when $x = 0$ we have $|y| \geq \alpha y$ iff $\alpha \in [-1, 1]$. We extend this component-wise[1] to get the expression for γ:

$$\gamma_j \in \begin{cases} \{1\} & b_j > 0 \\ \{-1\} & b_j < 0 \\ [-1, 1] & b_j = 0 \end{cases}$$

for $j = 1, ..., p$. Using this, we have that $\hat{\beta}(\lambda)$ minimizes Q iff

$$X_j^T(Y - X\hat{\beta}(\lambda)) = \lambda\gamma_j$$

for $j = 1, ..., p$ with γ_j defined as above for $\hat{\beta}(\lambda)$. This admits a key detail: $|X_j^T(Y - X\hat{\beta}(\lambda))| < \lambda$ implies $\hat{\beta}_j(\lambda) = 0$.

1.3. **Strong rules.** Suppose we have a grid of tuning parameters $\lambda_1, ..., \lambda_L$, which which we seek exact LASSO solutions. The optimality condition described above can lead us to an algorithm which will return exact solutions, but only needs to solve a series of smaller sub-problems.

First we need an additional assumption. Let $c_j(\lambda_k) = X_j^T(Y - X\hat{\beta}(\lambda_k))$ for $j = 1, ..., p$.

Assumption 1. $|c_j(\lambda) - c_j(\tilde{\lambda})| \leq |\lambda - \tilde{\lambda}| \quad \forall \lambda, \tilde{\lambda} > 0$

This says that $c_j(.)$ is non-expansive in its argument.

Theorem 1. *Suppose* $|c_j(\lambda_{k-1})| < 2\lambda_k - \lambda_{k-1}$ *and Assumption 1 holds. Then* $\hat{\beta}_j(\lambda_k) = 0$.

Proof. Observe

$$\begin{aligned} |c_j(\lambda_k)| &= |c_j(\lambda_k) - c_j(\lambda k - 1) + c_j(\lambda_{k-1})| \\ &\leq |c_j(\lambda_k) - c_j(\lambda k - 1)| + |c_j(\lambda_{k-1})| \\ &< (\lambda_{k-1} - \lambda_k) + (2\lambda_k - \lambda_{k-1}) \\ &= \lambda_k \end{aligned}$$

By the optimality condition for the LASSO problem this implies $\hat{\beta}_j(\lambda_k) = 0$. □

Theorem 1 yields a natural algorithm for computing the LASSO solution for a grid of tuning parameters.

1.3.1. *The strong rules algorithm.* This method requires data X, outcome Y, tuning parameters $\lambda_1 ... \lambda_L$, and an initial estimate $\hat{\beta}(\lambda_1)$[2] Then, for $k = 1, ..., L - 1$:

I Let $\upsilon \subset \{1, ..., p\}$ denote the set of eligible predictors, and let $S(\lambda_k) \subset \{1, ..., p\} = \{j : |c_j(\lambda_k)| >= 2\lambda_{k+1} - \lambda_k\}$. Set $\upsilon = S(\lambda_k)$.
II Solve the LASSO problem using only the predictors in υ.
III Check the subgradient optimality condition at *all* predictors in X. If none of them violate the condition, we are done, yielding $\hat{\beta}(\lambda_{k+1})$. If any of them violate the condition, add these predictors to υ and repeat steps two and three.

[1] Proof is simple, but requires a bit more subgradient calculus to do formally.
[2] This initial estimate could correspond to an intercept-only model, for instance.

2. BASIL

References