Ian Dardik
CS6140
Pathwise Coordinate Descent

1 Pathwise Coordinate Descent

Pathwise coordinate descent is a convenient algorithm for implementing regression algorithms such as LASSO and Elastic Net. Pathwise coordinate descent works by holding tuning parameters (λ, α) constant and optimizing each parameter individually[1]. Given data $\mathbf{X}$, response variable $\mathbf{Y}$, and tuning parameters λ and α, we can find the solution to an optimization problem

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}}\{O(\theta, X, Y, \lambda, \alpha)\}$$

by considering each parameter j as an individual optimization problem where we hold all other parameters $k \neq j$ constant. We can find $\hat{\theta}_j$ by finding the stationary points of a convex O with respect to θ_j:

$$\hat{\theta}_j = \theta_j \text{ s.t. } \frac{\partial O}{\partial \theta_j} = 0$$

In Pathwise Coordinate Descent we cycle through each parameter and make the appropriate update. After each complete cycle, we check to see if the difference between the previous and the newly updated parameters is smaller than a given tolerance, and terminate if so.

2 Standardized Data

We will consider the LASSO and the Elastic Net in the following sections. In both regression problems we will assume that our data is standardized so that each data vector in our design matrix has mean 0 and variance 1. Thus we can omit the intercept term as well in our regression, since the solution will be equal to the mean[1].

3 LASSO

LASSO regression solves the following optimization problem for each θ_j[1]:

$$\hat{\theta}_j = \underset{\theta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{N} \left(Y_i - \sum_{k=1}^{p} x_{ik}\theta_k \right)^2 + \lambda \sum_{k=1}^{p} |\theta_k| \right\}$$

Where $\lambda \geq 0$ is a tuning parameter. Let O be the formula we are optimizing above. We can solve for θ_j by finding the stationary points of O:

$$\begin{aligned} \frac{\partial O}{\partial \theta_j} &= -\sum_{i=1}^{N} X_{ij}(Y_i - X_{ij}\theta_j - \sum_{k \neq j}^{p} X_{ik}\theta_k) + \lambda \text{ sign}(\theta_j) \\ &= -\sum_{i=1}^{N} X_{ij}(Y_i - \sum_{k \neq j}^{p} X_{ik}\theta_k) + \sum_{i=1}^{N} X_{ij}^2\theta_j + \lambda \text{ sign}(\theta_j) \\ &= -\sum_{i=1}^{N} X_{ij}(Y_i - \sum_{k \neq j}^{p} X_{ik}\theta_k) + (N-1)\theta_j + \lambda \text{ sign}(\theta_j) \end{aligned} \tag{1}$$

The last step follows from the fact that our data is standardized, and thus $\sum_{i=1}^{N} X_{ij}^2 = N - 1$.

Set $\frac{\partial O}{\partial \theta_j} = 0$ and, for convenience, let $Q = \sum_{i=1}^{N} X_{ij}(Y_i - \sum_{k \neq j}^{p} X_{ik}\theta_k)$ and and we get:

$$(N-1)\theta_j + \lambda \text{ sign}(\theta_j) = Q$$

To solve for θ_j notice that when $\theta_j \geq 0$, $\theta_j = \frac{Q-\lambda}{N-1}$ so $Q \geq \lambda$. However when $\theta_j \leq 0$, $\theta_j = \frac{Q+\lambda}{N-1}$ so $Q \leq -\lambda$. Notice the following two properties:

(1) $|Q| \geq \lambda$

(2) $\theta_j \geq 0$ iff $Q \geq 0$ and $\theta_j \leq 0$ iff $Q \leq 0$

Therefore, we can write the solution for θ_j as:

$$\theta_j = \frac{\text{sign}(Q)(|Q| - \lambda)_+}{N-1}$$

Where $()_+$ is the soft-thresholding operator and guarantees property (1), while multiplying by $\text{sign}(Q)$ makes use of (and guarantees) property (2). If we let[1] $S(t) = \text{sign}(t)(|t| - \lambda)$ then our final solution is:

$$\theta_j = S\left(\sum_{i=1}^{N} X_{ij}(Y_i - \sum_{k \neq j}^{p} X_{ik}\theta_k)\right)/(N-1)$$

Notice this yields a solution nearly identical to Elements of Statistical Learning[1]; the difference is that we divide by $N-1$ and I'm not sure why they omit this in the book.

4 Elastic Net

The Elastic Net uses the tuning parameter $\alpha \in [0, 1]$ to compromise between Ridge and Lasso penalization. It solves the following optimization problem:

$$\hat{\theta}_j = \underset{\theta}{\text{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{N} \left(Y_i - \sum_{k=1}^{p} x_{ik}\theta_k \right)^2 + \lambda \sum_{k=1}^{p} \left(\alpha\theta_k^2 + (1-\alpha)|\theta_k| \right) \right\}$$

This problem is very similar to LASSO above so I will write slightly more concisely here. Let O be the formula we are optimizing above. We can solve for θ_j by finding the stationary points of O:

$$\frac{\partial O}{\partial \theta_j} = -\sum_{i=1}^{N} X_{ij}(Y_i - \sum_{k \neq j}^{p} X_{ik}\theta_k) + (N-1)\theta_j + 2\lambda\alpha\theta_j + \lambda(1-\alpha) \text{ sign}(\theta_j) \tag{2}$$

Set $\frac{\partial O}{\partial \theta_j} = 0$ and, for convenience, let $Q = \sum_{i=1}^{N} X_{ij}(Y_i - \sum_{k \neq j}^{p} X_{ik}\theta_k)$, $A = (N-1) + 2\lambda\alpha$, and $B = \lambda(1-\alpha)$ and and we get:

$$A\theta_j + B \text{ sign}(\theta_j) = Q$$

Notice that since $A \geq 0$ and $B \geq 0$, we can use the same two case logic from the LASSO section, yielding the solution:

$$\theta_j = \frac{\text{sign}(Q)(|Q| - B)_+}{A}$$

If we define $S_{en}(t) = \text{sign}(t)(|t| - \lambda(1-\alpha))$ then our final solution is:

$$\theta_j = S_{en}\left(\sum_{i=1}^{N} X_{ij}(Y_i - \sum_{k \neq j}^{p} X_{ik}\theta_k)\right) / \left((N-1) + 2\lambda\alpha\right)$$

References

[1] Trevor Hastie, Robert Tibshirani, and Jerome Friedman, The Elements of Statistical Learning 2nd Edition