# Machine Learning for Economic Analysis
# Problem Set 6

Jonas Lieber*

Due: 11:59pm Wed, March 6, 2024

**Problem 1.** *Coordinate Descent for Regularized Regression*

*In ps6.csv, you are given* $(Y_1, X_1), \ldots, (Y_n, X_n)$*. Consider some* $\alpha \in [0,1]$ *and* $\lambda \geq 0$*. The purpose of this problem is to minimize the function*

$$\begin{aligned} f(y, X, \beta, \lambda, \alpha) &= \frac{1}{n}\frac{1}{2}\sum_{i=1}^{n}\left(y_i - \beta_0 - x_i'\beta\right)^2 + \lambda\left(\alpha\left\|\beta\right\|_1 + (1-\alpha)\frac{1}{2}\left\|\beta\right\|_2^2\right) \\ &= \frac{1}{n}\frac{1}{2}\left\|Y - \beta_0 1_{n\times 1} - X\beta\right\|_2^2 + \lambda\left(\alpha\left\|\beta\right\|_1 + (1-\alpha)\frac{1}{2}\left\|\beta\right\|_2^2\right) \end{aligned}$$

1. *Getting rid of the Ridge penalty by "data-augmentation"*

   (a) *For any two vector* $z_1 \in \mathbb{R}^{m_1}$*,* $z_2 \in \mathbb{R}^{m_2}$*, we define the vector* $z = (z_1, z_2) \in \mathbb{R}^{m_1+m_2}$*. Show that*

   $$\|z\|_2^2 = \|z_1\|_2^2 + \|z_2\|_2^2$$

   (b) *Show that for* $\tilde{y} = (y, 0_p)$*,* $\tilde{I} = (1_{n\times 1}, 0_{p\times 1})^t$ *and* $\tilde{X} = (X^t, \sqrt{(1-\alpha)\lambda}I_p)^t$ *we have*

   $$f(y, X, \beta, \lambda, \alpha) = \frac{1}{2n}\left\|\tilde{y} - \tilde{I}\beta_0 - \tilde{X}\beta\right\|_2^2 + \lambda\alpha\left\|\beta\right\|_1.$$

   *In the rest of the question, we will consider only the LASSO problem and denote the outcome by* $y$ *(instead of* $\tilde{y}$*), the regressors/features by* $X$ *(instead of* $\tilde{X}$*).*

   (c) *Why is this useful?*

2. *Coordinate Descent Update*

   (a) *Suppose that we are given a candidate value* $\tilde{\beta}$*. For some* $j \in \{1, \ldots, p\}$*, we consider the problem*

   $$\min_{\beta} f(y, X, \beta, \alpha) \quad s.t. \quad \beta_l = \tilde{\beta}_l \text{ for all } l \neq j. \tag{1}$$

   *Describe this program in words.*

*Department of Economics, Yale University. jonas.lieber@yale.edu

(b) *Implement a function that takes* $y, X, \lambda, \alpha, \tilde{\beta}, j$ *and returns* $\beta_j^*$*, the* $j$*-th component of the argmin in problem* (1)*. It is given by*

$$\beta_j^* = \frac{1}{x_j^t x_j} S(\hat{y}^t x_j, \alpha\lambda),$$

*where* $S$ *is the soft-thresholding function, i.e. for* $a, b \in \mathbb{R}$*,*

$$S(a, b) = sign(a)(|a| - b)_+,$$

*where*

$$sign(a) = \begin{cases} -1 & \text{if } a < 0 \\ 0 & \text{if } a = 0 \\ 1 & \text{if } a > 0, \end{cases}$$

*and*

$$(a)_+ = \begin{cases} a & \text{if } a > 0, \\ 0 & \text{if } a \leq 0, \end{cases}$$

*and* $\hat{y}$ *is a vector given by*

$$\hat{y}_i = \tilde{\beta}_0 + \sum_{\substack{l=1 \\ l \neq j}} x_{i,j} \tilde{\beta}_j.$$

(c) *Plot the functions* $S(z, 1)$ *and* $S(z, 0)$ $z \in [-3, 3]$*. Interpret these functions.*[1]

3. *Implement a function for cyclic coordinate descent by using this update for* $j = 0$ *(which* $\lambda$ *is used here?) and then* $j = 1, \ldots, p$*, then again* $j = 1$*,* $j = 1, \ldots, p$*.*

4. *Add an option in this function for an active set strategy: After cycling through the* $\beta$*s, you cycle only through the non-zero parameters until you have convergence for these parameters. Only then you cycle through all* $\beta$*s again.*

5. *Compute the* $\lambda$ *sequence of the glmnet package:*

   (a) *Choose a length* $l$ *of the sequence, with default* $l = 100$*.*

   (b) *Define* $\lambda_{\max}$ *as*

$$\lambda_{\max} = \max_{j \in \{1, \ldots, p\}} \left| \frac{(x_j - \bar{x}_j)^t y}{\sqrt{\frac{1}{n}(x_j - \bar{x}_j)^t (x_j - \bar{x}_j)}} \right|.$$

[1] Hints: One of these two functions relates to coordinate descent for OLS. Why? How "likely" is it that either function is zero (for example if z is uniformly distributed between $-3$ and 3)? How is this related to sparsity of the estimates?

(c) *Choose a small* $\delta > 0$*, with default* $\delta = 0.0001$ *and set*

$$\lambda_{\min} = \delta \lambda_1.$$

(d) *Take an equidistant sequence of length* $l$ *from* $\log(\lambda_{\min})$ *to* $\log(\lambda_{\max})$*. The exponential of this sequence is the lambda sequence.*

6. *Add an option in your cross-validation sequence to use "warm-starts": As a starting point for the highest value of* $\lambda$*, use* $\beta = 0$*. As you move to smaller* $\lambda$ *values, use the minimizer for the next higher* $\lambda$ *as a starting value.*

7. *Run Lasso with this grid and 5-fold CV. Plot the parameter estimates as a function of* $\lambda$ *(all in one plot). Report the running time with and without active set strategy and with and without warm starts for cross-validation.*

8. *Run Ridge with this grid and 5-fold CV. Plot the parameter estimates as a function of* $\lambda$*. Report the running time with and without active set strategy and with and without warm starts for cross-validation.*

9. *Now we consider the elastic net. As a grid for* $\alpha$*, use* $0, 0.1, 0.2, \ldots, 0.9, 1$*. Run the elastic net where you cross-validate both* $\alpha$ *and* $\lambda$ *with 5-fold CV. Report the running time with and without active set strategy and with and without warm starts for cross-validation.*

10. *Which estimator(s) leads to sparse estimates?*

11. *(**graduate students & groups of 3 only**) Now benchmark your implementation of coordinate descent for OLS (which* $\lambda$ *do you have to choose?) with your implementation for gradient descent from problem set 4 using reg.csv from problem set 4. Redo the plot from problem set 4 with batch gradient descent and cyclic coordinate descent.* [2]

**Problem 2.** *Kernel Ridge*

1. *Consider ps6.csv again. Consider the Ridge estimator for* $\lambda = 1$*. Show numerically that*

$$(X'X + \lambda I_p)^{-1} X'y = X'(XX' + \lambda I_n)^{-1} y.$$

*When do you prefer which formula?*

2. *Now consider the DGP from problem 1 of problem set 1 again. Using an RBF kernel with* $\sigma = 1$*, use a kernel Ridge regression of* $y$ *on* $x$ *and plot the estimated function together with the true function and the data for* $\lambda = 0.1, 0.5, 1, 5, 10$*.*

3. *In Using an RBF kernel with* $\sigma = 1$*, use a kernel Ridge regression of* $y$ *on* $x$ *and plot the estimated function together with the true function and the data for* $\sigma = 0.1$ *and* $\lambda = 0.005$*. Briefly compare this plot to the plot of fitted functions in problem set 1.*

[2] Hint: It is important to think about what to use on the x-axis. What is the number of iterations for cyclic coordinate descent? A single coordinate update? 10 coordinate updates? A cycle of coordinate updates? One way to make these two comparable in the plot might be to use time on the x-axis.