EECS289A Introduction to Machine Learning, Project T Final - Quiz Questions

Han Liu, Peng Tan, Dilu Xu, Jinyan Zhao
Department of Civil Engineering
Universitu of California, Berkeley
Berkeley, CA 94704
{han_liu, tanpeng, diluxu, jinyan_zhao}@berkeley.edu

November 26, 2020

1 Elastic Net (HL)

In class we discussed two types of regularization, l_1 and l_2. Both are useful, and sometimes it is helpful combine them, giving the objective function below (**in this problem we have excluded w_0 for simplicity**):

$$F(\boldsymbol{w}) = \frac{1}{2}\sum_{j=1}^{n}(y^{(j)} - \sum_{i=1}^{d}\boldsymbol{w}_i\boldsymbol{x}_i^{(j)})^2 + \alpha\sum_{i=1}^{d}|\boldsymbol{w}_i| + \frac{\lambda}{2} + \sum_{i=1}^{d}\boldsymbol{w}_i^2 \tag{1}$$

Here, $(\boldsymbol{x}^{(j)}, y^{(j)})$ is j-th example in the training data, w is a d dimensional weight vector, λ is a regularization parameter for the l_2 norm of w, and α is a regularization parameter for the l_1 norm of w. This approach is called the Elastic Net, and you can see that it is a generalization of Ridge and Lasso regression: It reverts to Lasso when $\lambda = 0$, and it reverts to Ridge when $\alpha = 0$. In this question, we are going to derive the coordinate descent (CD) update rule for this objective.

Let g, h, c be real constants, and consider the function of x

$$f_1(x) = c + gx + \frac{1}{2}hx^2 \tag{2}$$

1. [4 points] What is the x^* that minimizes $f_1(x)$? (i.e. calculate $x^* = \arg\min f_1(x)$)
   Answer:
   Take the gradient of $f_1(x)$ and set it to 0.

$$g + hx = 0 \tag{3}$$

$$x^* = -\frac{g}{h} \tag{4}$$

Let α be an additional real constant, and consider another function of x

$$f_2(x) = c + gx + \frac{1}{2}hx^2 + \alpha|x|(h > 0, \alpha > 0) \tag{5}$$

This is a piecewise function, composed of two quadratic functions:

$$f_2^-(x) = c + gx + \frac{1}{2}hx^2 - \alpha x \tag{6}$$

and

$$f_2^+(x) = c + gx + \frac{1}{2}hx^2 + \alpha x \tag{7}$$

Let $\tilde{x}^- = \arg\min_{x \in \mathbb{R}} f_2^-(x)$ and $\tilde{x}^+ = \arg\min_{x \in \mathbb{R}} f_2^+(x)$.
(**Note:** The argmin is taken over $(-\infty, +\infty)$ here.

2. [6 points] What are $\tilde{x}^+$ and $\tilde{x}^+$? Show that $\tilde{x}^- \geq \tilde{x}^+$.
   Answer:
   Using the answer from part 1), we get $\tilde{x}^+ = -\frac{g+\alpha}{h}$ and $\tilde{x}^- = -\frac{g-\alpha}{h}$. Since $\tilde{x}^- - \tilde{x}^+ = \frac{2\alpha}{h} \geq 0$, we have $\tilde{x}^- \geq \tilde{x}^+$.

3. [12 points] Draw a picture of $f_2(x)$ in each of the three cases below:

   (a) $\tilde{x}^- > 0$, $\tilde{x}^+ > 0$

   (b) $\tilde{x}^- < 0$, $\tilde{x}^+ < 0$

   (c) $\tilde{x}^- > 0$, $\tilde{x}^+ < 0$

   For each case, mark the minimum as either 0, $\tilde{x}^-$, or $\tilde{x}^+$. (You do not need to draw perfect curves, just get the rough shape and the relative locations of the minima to the x-axis)
   Answer:
   Fig. 1 gives the example picture of three cases. To understand the answer, note the following:

   - $f_2^+(0) = f_2^-(0)$, this means the two piece of f_2 match each other at x $= 0$.
   - When $\tilde{x}^- \geq 0$, the minimum of the left part is at 0, i.e. $\arg\min_{x \in (-\infty, 0]} f_2^-(x) = 0$
   - When $\tilde{x}^- \leq 0$, the minimum of the left part is at $\tilde{x}^-$, i.e. $\arg\min_{x \in (-\infty, 0]} f_2^-(x) = \tilde{x}^-$
   - Similar rules holds for the right part of the curve.
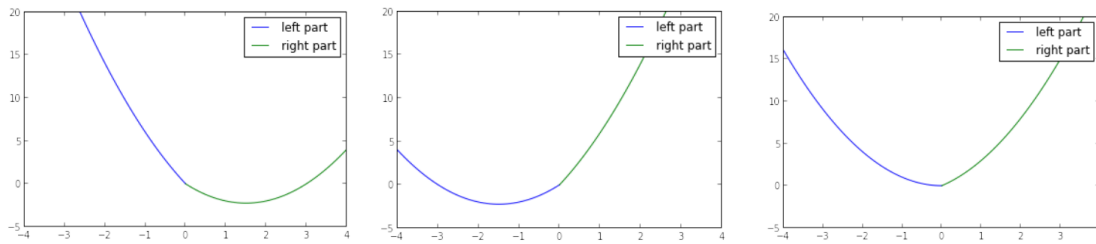
(a) $\tilde{x}^- > 0$, $\tilde{x}^+ > 0$, the minima is at $\tilde{x}^+$.

(b) $\tilde{x}^- < 0$, $\tilde{x}^+ < 0$, the minima is at $\tilde{x}^-$.

(c) $\tilde{x}^- > 0$, $\tilde{x}^+ < 0$, the minima is at 0.

Fig. 1. Example picture of three cases.

4. [8 points] Write x^*, the minimum of $f_2(x)$, as a piecewise function of g. (Hint: what is g in the cases above?)
Answer:
Summarizing the result from the previous question, we have

$$
x^* = \begin{cases} \tilde{x}^+ & \text{if } \tilde{x}^+ > 0 \\ \tilde{x}^- & \text{if } \tilde{x}^- < 0 \\ 0 & \text{if } \tilde{x}^+ < 0, \tilde{x}^- > 0 \end{cases} \tag{8}
$$

This is equivalent to the soft-threshold function

$$
x^* = \begin{cases} -\frac{g+\alpha}{h} & \text{if } g < -\alpha \\ 0 & \text{if } g \in [-\alpha, \alpha] \\ -\frac{g-\alpha}{h} & \text{if } g > \alpha \end{cases} \tag{9}
$$

5. [8 points] Now let's derive the update rule for w_k. Fixing all parameters but w_k, express the objective function in the form of Eq. (5) (here w_k is our x). You do not need to explicity write constant factors, you can put them all together as c. What are g and h? Then putting it all together, write down the coordinate descent algorithm for the Elastic Net (**again excluding** w_0). You can write the update for w_k in terms of the g and h you calculated in the previous question.
Answer:

$$
g = \sum_{j=1}^n x_k^{(j)} \left(\sum_{i \neq k} \boldsymbol{w}_i \boldsymbol{x}_i^{(j)} - y_i \right) \tag{10}
$$

$$
h = \lambda + \sum_{j=1}^n (x^{(j)})^2 \tag{11}
$$

The algorithm is shown in Algorithm. 1. You can compare it to the CD algorithm for Lasso, the only difference is adding λ to h.

Algorithm 1: Coordinate Descent for Elastic Net

---
while *not converged* **do**
  for $k \in \{1, 2, \cdots, d\}$ **do**
    $g = \sum_{j=1}^n x_k^{(j)} (\sum_{i \neq k} \boldsymbol{w}_i \boldsymbol{x}_i^{(j)} - y_i)$;
    $h = \lambda + \sum_{j=1}^n (x^{(j)})^2$;
    $w_k = \begin{cases} -\frac{g+\alpha}{h} & \text{if } g < -\alpha \\ 0 & \text{if } g \in [-\alpha, \alpha] \\ -\frac{g-\alpha}{h} & \text{if } g > \alpha \end{cases}$;
  end
end

---

2   True or False questions (HL)

Answer True or False. Justify your answer very briefly in 1-2 sentences.

1. Increasing the regularization parameter λ in lasso regression leads to sparser regression coefficients.
   Answer:
   True. Larger regularization parameter penalizes non-zero coefficients more, leading to sparser solution.

2. In Lasso regression, can the regulariser increase the sparsity of the resulting solutions?
   Answer:
   True, Lasso introduces the L1 norm penalty, so increasing the penalty will cause more and more of the parameters to be driven to zero. As seen in Fig. 2, where the red contours are representing the error function and the solid blue areas the constraint regions for the parameters, in the Lasso case, the function is likely to hit the constraint region on one of the four corners. This will make a parameter equal with zero.
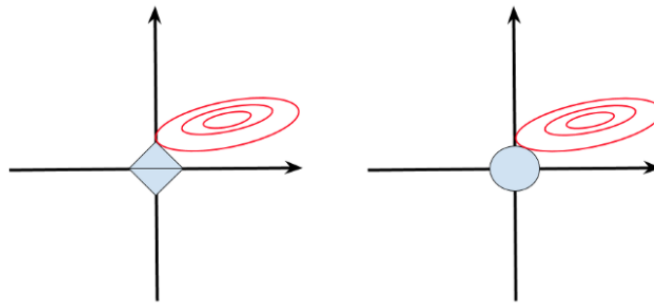

Fig. 2. Spam-Estimation picture for the LASSO (left) and Ridge regression (right).

3. Is the objective of Lasso regression differentiable and convex?
   Answer:
   False, it is not differentiable. As Lasso uses L1 norm penalty, this L1 loss function is defined based on the absolute value of the difference between two values. Since the loss function is a modulus function, there will exist an inflection point where the slope cannot be computed and the function is not differentiable. In contrast, for the case of Ridge regression, where we use L2 norm penalty, we can get the slope at any point and the objective is differentiable.

4. Are the objectives of Ridge and Lasso regression both convex and have closed-form solutions?
   Answer:
   False, they are both convex, but Lasso does not always have a closed-form solution. LASSO regression must be solved numerically by using quadratic programming or more general convex optimisation methods.

5. Is it true that the L1 term in Lasso has the following purposes: performing feature selection, compensating for overfitting and smoothing?
   Answer:

False. The first two purposes are true, but not the last one. As it is mentioned early, both L1 and L2 terms are introduced to compensate for overfitting. By making the parameters equal with zero, the L1 norm discards some features and, thus, performs feature selection. In terms of smoothing, we cannot say L1 has this purpose, as the L1 loss function does not have continuous derivatives.