4033/5033: Assignment 1

Your Name

Due: Sep 23 (by 11:59pm)

Problem 1. Weighted least square is a technique to learn linear regression models which weighs instances during training. Its objective function is

$$J(\beta) = \sum\nolimits_{i=1}^{n} w_i \cdot (x_i^T \beta - y_i)^2 \tag{1}$$

where $w_i \in \mathbb{R}$ is the weight for instance $x_i \in \mathbb{R}^p$. If a w_i is large, then β will focus on fitting x_i and thus gain lower error on x_i (in theory). After β is learned, we can apply it to predict label for any instance z by $z^T\beta$. Complete the following three tasks.

Task 1. Derive the matrix form of $J(\beta)$. In the result, you can use the following notations: X is an n-by-p matrix with x_i^T on the i_{th} row and Y is an n-dimensional vector with y_i being its i_{th} element.[1]

Task 2. Derive an analytic solution of β and present it in the matrix form.

Task 3. Implement your solution in Python from scratch. Test your implementation on the given Community Crime data set and report experimental results. Use the following experiment design.

– Randomly select 75% data for training and use the rest 25% data for testing. Repeat the random trial for 10 times and report the average testing error (RMSE).

– Let us define two groups in the community: (i) high crime rate group contains community instances whose labels are bigger than 0.8; (ii) low crime rate group contains community instances whose labels are no bigger than 0.8. After experimenting weighted least square, let us report its testing errors on different groups in Table 1. (We will use w_h to denote weight for instances in the high crime rate group and w_ℓ to denote weight for instances in the low crime rate group.)

– Report the average number of training instances in the two groups in Table 2.

Choice of Weights	$w_\ell = 1, w_h = 1$	$w_\ell = 1, w_h = 10$	$w_\ell = 1, w_h = 50$	$w_\ell = 1, w_h = 0.1$
Error on all testing instances	...	...	...	...
Error on high crime rate group	...	...	...	...
Error on low crime rate group	...	...	...	...

Table 1. Performance of Weighted Least Square

# Instances in the High Crime Group	....
# Instances in the Low Crime Group	....

Table 2. Number of Instances in the Two Groups

[1] Tip: use the matrix form of $\sum_i w_i a_i^2$.

Problem 2. Implement Lasso plus coordinate descent (CD) and evaluate it on the Community Crime data set. Use the following experiment design.

– Use the first 75% data for training and the rest 25% for testing.

– Pick a proper regularization coefficient yourself so you can get a sparse model. (You can examine the model coefficients after it is learned.)

– Fix the picked coefficient, report performance of your implemented algorithm in Figure 1 and Figure 2.

Figure 1 should contain a curve of testing error versus the number of CD updates (y-axis is testing error and x-axis is number of CD updates). You should choose a proper range of x-axis so we can observe convergence of your testing error.

Fig. 1. Testing Error versus CD Updates

Figure 2 should contain a curve of the number of non-zero elements in your model versus the number of CD updates (y-axis is number of non-zero elements, and x-axis is number of CD updates). The range of x-axis should be same as in Figure 1.

Fig. 2. Number of Non-Zero Elements in β versus CD Updates

Submission Instruction

Please submit three files to Canvas.

(i) Submit a 'hw1.pdf'. It should contain your answers to all the questions. For mathematical questions, you can write the answers on a paper, scan it and include it in the pdf file; or, you can also directly type the answers in Latex and compile them into pdf. For experimental questions, you need to draw the figures using Python and include them in the pdf file.

(ii) Submit a 'hw1_WLS.py'. It should be the code of your implemented weighted least square. Please make sure we can directly run your code without changing any parts (expect directory of the data set maybe).

(iii) Submit a 'hw1_Lasso.py'. It should be the code of your implemented Lasso. Please make sure we can directly run your code without changing any parts (expect directory of the data set maybe).