4033/5033: Assignment 3

Your Name

Due: Oct 12 (by 11:59pm)

Kernel ridge regression (KRR) takes $O(n^2p + n^3)$ time to learn a model, where n is the number of training instances and p is the original feature dimension. Let's design an accelerated kernel ridge regression (AKRR). method that takes only $O(m^2p + m^3)$ time, where m is a hyper-parameter we can manually pick.

Recall that in KRR, the optimal model has the form

$$\beta = \sum_{i=1}^{n} \alpha_i \phi(x_i). \tag{1}$$

In AKRR, we approximate this form by

$$\tilde{\beta} = \sum_{i=1}^{m} \alpha_i \phi(x_i), \tag{2}$$

which means the optimal model is a linear combination of only the first m training instances. Thus AKRR learns the optimal model (equivalently, optimal $\alpha_1, \ldots, \alpha_m$) by solving

$$\min_{\tilde{\beta}} \sum_{i=1}^{n} (\phi(x_i)^T \tilde{\beta} - y_i)^2 + \lambda \tilde{\beta}^T \tilde{\beta}. \tag{3}$$

As you will mathematically justify in this assignment, the time complexity for AKRR is $O(m^2p+m^3)$, which is more efficient than KRR when $m < n$. You will also empirically justify in this assignment that, the error of AKRR is comparable to the error of KRR especially when m approaches n.

Complete the following three tasks.

1. Derive the optimal $\alpha_1, \ldots, \alpha_m$ in (3). Write your results in matrix form. Clearly explain every notation in your solution (unless it is frequently used in class or clarified in the notation list on Canvas).

Tip: Here are three notations that may be useful.

– $\alpha = [\alpha_1, \ldots, \alpha_m]^T$ is an m-dimensional vector

– $Y = [y_1, \ldots, y_n]^T$ is an n-dimensional vector

– $\tilde{K}$ is an n-by-m matrix where its element at row i and column j is $\tilde{K}_{ij} = k(x_i, x_j)$

2. Justify that the time complexity to calculate the optimal α is $O(m^2p + m^3)$.

3. Implement your solution of AKRR from scratch in Python and report its performance versus m. In experiment, evaluate AKRR on the Community Crime data set. Use the first 75% of data for training and the rest 25% data for testing. Use RBF kernel and choose a proper gamma by yourself. Report your testing RMSE in Figure 1. This figure should contain one curve of testing RMSE versus m (i.e. y-axis is RMSE and x-axis is m). Pick 10 values of m yourself to get a comprehensive understanding of the impact of m on testing error. The last value of m must be n (in which case AKRR is the same as KRR).

Fig. 1.

Submission Instruction

Please submit two files to Canvas.

(i) Submit a 'hw3.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) A Python source code for task 3, named as 'hw3.py'.