EE 516: Mathematical Foundations of Machine Learning	**Winter 2023**

Homework 9
Due: March 19, 2023, 11:59PM PT

Student Name: *Instructor Name: John Lipor*

Problem 1 (5 pts each)

In Homework 8 we focused on solving sparse regression problems by incorporating an ℓ_1-norm regularizer. If we want to encourage a different type of behavior, we can consider different regularizers. Moreover, there are structures in signals besides sparsity that still allow recovery in the case of highly under-determined systems. One popular regularizer is called the *total variation* (TV), which has applications in image denoising and inpainting. For a signal $w \in \mathbb{R}^D$, the TV is defined as

$$\mathrm{TV}(w) = \sum_{i=1}^{D-1} |w_{i+1} - w_i|,$$

where w_i denotes the ith element of w. In this problem, you will implement TV-regularized regression, which is useful for estimating signals that are known to be piecewise constant, as shown in Fig. 1 below.

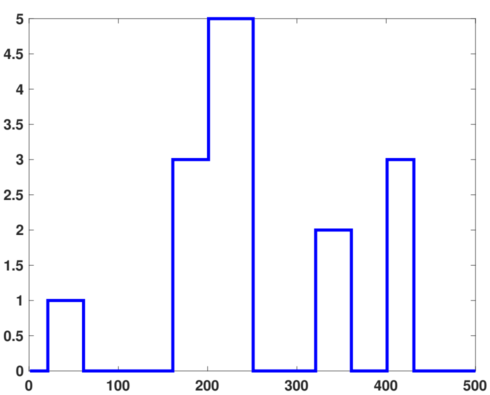

Figure 1: Example piecewise constant function for total variation-penalized regression.

(a) Define a matrix $C \in \mathbb{R}^{D-1 \times D}$ such that

$$\mathrm{TV}(w) = \sum_{i=1}^{D-1} |w_{i+1} - w_i| = \|Cw\|_1.$$

(b) The TV-regularized regression problem can be written as

$$\min_{w\in\mathbb{R}^D} \frac{1}{2}\left\|Xw-y\right\|_2^2+\lambda\left\|Cw\right\|_1 \tag{1}$$

or in summation form as

$$\min_{w\in\mathbb{R}^D} \frac{1}{2}\sum_{i=1}^{N}\left(x_i^T w-y_i\right)^2+\lambda\sum_{i=1}^{D-1}\left|w_{i+1}-w_i\right|.$$

Determine the ADMM updates to solve Eq. (1). You may need to apply the splitting technique and introduce a new optimization variable.

(c) Implement your devised ADMM algorithm by completing `tvADMM` in the files for download and test your algorithm on the script `prob1`. **Turn in:**

- Your `tvADMM` code
- A plot of the true and estimated functions
- A plot of the relative error between iterations as a function of the iteration number, where the error is defined as

$$\mathrm{err}_k=\frac{\left\|w^{(k)}-w^{(k-1)}\right\|}{\left\|w^{(k)}\right\|},$$

where $w^{(k)}\in\mathbb{R}^D$ denotes the estimate of w at the kth iteration.

Problem 2 (5 pts)

In Demo 3, we used the matrix inversion lemma to make ridge regression amenable to the kernel trick. Go through the description of how Eq. (1) in Demo 2 is derived and write out the steps explicitly here.

Problem 3 (5 pts, 5 pts)

On previous assignments, you developed classifiers based on ridge regression/least squares as well as through minimization of both the hinge and logistic loss. One drawback to all of these approaches is that they result in *linear* decision boundaries. However, many datasets are not linearly separable, and hence our goal in this problem is to develop a classifier that allows for nonlinear decision boundaries. To do this, you will use kernel ridge regression (KRR, see demo code) to form a kernelized least squares classifier, which you will test on the MNIST dataset.

(a) Use either my `KRR` function or your own to perform classification on MNIST digits 1 and 2 after reducing the dimensionality to 2 using PCA by completing the `prob3` script, taking $\lambda = 0.01$ in `KRR`. **Turn in** plots showing the separator and the traning data for $\sigma \in \{0.1, 1, 5\}$. Provide the training and test error for each combination and comment on each plot.

(b) Describe in words how the provided code plots the decision boundary formed by `KRR`.

Problem 4 (5 pts)

Find one article, podcast, or talk on the subject of AI ethics that you think would be beneficial to anyone working in this area. Share your resource in the `interesting-reading` channel along with one or two sentences describing why you chose it.

Problem 5 (3 pts + 50 Lipor pts)

You should have received one or more messages requesting that you complete a course evaluation. Please complete this evaluation, and type "I have completed the course evaluation" once you've done so. **Note:** To maintain anonymity, I will not review which students answer affirmatively to this question.