Homework # 4

Foundations of Computer and Data Science CS-596

Problem 1: We would like to test the ability of the first kernel method (discussed in the class) to approximate pdfs. Therefore we generate 1000 realizations of a random variable uniformly distributed in $[0, 1]$. a) Approximate the corresponding pdf using the Gaussian kernel

$$\mathsf{K}(x, h) = \frac{1}{\sqrt{2\pi h}} e^{-\frac{1}{2h}x^2}.$$

Plot the resulting approximations for different values of h. Do not limit your graph in $[0, 1]$ but extend it beyond the two boundaries. What do you observe as far as capturing the support of the random variable (the interval $[0, 1]$) and the constant value of the pdf (equal to 1 in $[0, 1]$) is concerned? b) Complete the same steps for the Laplacian kernel

$$\mathsf{K}(x, h) = \frac{1}{2h} e^{-\frac{1}{h}|x|}.$$

Problem 2: The Matlab data file `hw4-2data.mat` contains two matrices: `stars` and `circles` each being a list of 2 dimensional (2-D) vectors. Each 2-D vector identifies a point in the 2-D space which is labeled either as star or circle. We are interested in developing a classifier that distinguishes between the two sets. We would like to use the kernel method to find a nonlinear separating boundary. To achieve this we assign the label "1" to `stars` and the label "-1" to `circles` and if $\phi(X), X = [x_1, x_2]^\intercal$ is the transformation we would like to apply to the data then we want to solve the following minimization problem to find the optimum $\phi(X)$

$$\min_{\phi \in \mathcal{V}} \left\{ \sum_{X_i \in \text{stars}} \big(1 - \phi(X_i)\big)^2 + \sum_{X_j \in \text{circles}} \big(1 + \phi(X_j)\big)^2 + \lambda \|\phi(X)\|^2 \right\}, \tag{1}$$

where $\mathcal{V}$ is the vector space of functions generated by the Gaussian kernel

$$\mathsf{K}(X, Y) = e^{-\frac{1}{h}\|X - Y\|^2} = e^{-\frac{1}{h}\{(x_1 - y_1)^2 + (x_2 - y_2)^2\}}.$$

a) Use the Representer theorem to prove that for the first two sums we can replace $\phi(X)$ by its orthogonal projection $\hat{\phi}(X)$ onto the linear subspace generated by $\mathsf{K}(X, X_i), X_i \in$ stars and $\mathsf{K}(X, X_j), X_j \in$ circles where

$$\hat{\phi}(X) = \sum_{X_i \in \text{stars}} \alpha_i \mathsf{K}(X, X_i) + \sum_{X_j \in \text{circles}} \beta_j \mathsf{K}(X, X_j). \tag{2}$$

b) For the term $\|\phi(X)\|^2$ use the orthogonality principle to show that

$$\|\phi(X)\|^2 = \|\hat{\phi}(X)\|^2 + \|\phi(X) - \hat{\phi}(X)\|^2 \geq \|\hat{\phi}(X)\|^2.$$

Does this suggest that we can replace $\phi(X)$ with $\hat{\phi}(X)$ everywhere in the original cost function in (1)? If yes, then WHY? c) Using the form of $\hat{\phi}(X)$ defined in (2), find the optimum coefficients α_i, β_j. d) Once you identify the optimum $\hat{\phi}(X)$ explain how you are going to use it to classify a new point X_{new} as "star" or "circle" given, of course, that $\hat{\phi}(X_{\text{new}})$ will not be exactly equal to 1 or -1. e) After you have specified your final classification rule in d) find (numerically) the separating boundary for the two classes in the 2-D space (also place the training points on the 2-D plane to verify the quality of your boundary). Repeat the process for different values of h and λ.

Problem 3: Suppose that the scalar random variable y and the random vector X are related through the following model

$$y = \theta_*^\intercal X + w, \tag{3}$$

where w is a scalar random variable with mean 0 and independent from X and θ_* a deterministic vector. Assuming that we have knowledge of the statistical behavior of the random pair (y, X) we would like to *estimate* θ_* by solving the following minimization problem

$$\min_{\theta} \mathsf{E}[(y - \theta^{\intercal} X)^2]. \tag{4}$$

a) Find the optimum θ and prove that it is equal to θ_*. b) Assume now that the statistical behavior of the pair (y, X) is not available and, instead, you are observing pairs (y_t, X_t) that are independent realizations of (y, X). Find the learning (adaptive) algorithm that provides a new estimate θ_t every time a new pair (y_t, X_t) becomes available. The resulting algorithm is known as *Least Mean Squares* (LMS) and constitutes the most well known algorithm in Adaptive Signal Processing. c) Use your own favorite θ_*, of length 5, generate pairs (y_t, X_t) using the model in (3), where X_t is a Gaussian vector of length 5 with independent and identically distributed elements of mean 0 and variance 1 and w_t is again Gaussian independent from X_t with mean 0 and variance 0.1. Apply your algorithm from b) and verify its convergence towards θ_*. Select a small learning rate so that your estimates are not very "noisy". Plot the squared norm $\|\theta_t - \theta_*\|^2$ as a function of the iteration t and examine whether it converges to something small. Use logarithmic scale ("semilogy" in Matlab) to be able to observe differences between small quantities. d) Repeat the previous simulation but with a learning rate which is half the learning rate you used in question c). What do you observe as far as convergence rate and steady state error is concerned? Can you compare the performance in the two cases and claim that one is "better" than the other?

There will be a meeting on Friday, December 06, at 5PM, in Core-101 to discuss the problems.

Your report, in *hard copy*, must be submitted on Monday, December 09, in CBIM to our TA, Mr. Stathopoulos, between 10:00-11:00AM.