ECEN 5712 (M. Varanasi) **Homework 4** **Due Mar. 13, 2022**

1. (A simple binary classifier) Given training data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ with $y_i \in \mathcal{Y} = \{-1, 1\}$, define the two average vectors obtained by averaging the inputs in each category as

$$\mu_1 = \frac{1}{m_1} \sum_{\{i:y_i=+1\}}^{m_1} \mathbf{x}_i \qquad \mu_{-1} = \frac{1}{m_{-1}} \sum_{\{i:y_i=-1\}}^{m_{-1}} \mathbf{x}_i$$

where m_1 and m_{-1} are the number of inputs with positive and negative labels, respectively.

   a. Consider the following binary classifier for a test input $\mathbf{x}$:

$$\hat{y}(\mathbf{x}) = \arg\min_{y \in \mathcal{Y}} \|\mathbf{x} - \mu_y\|$$

   where $\|\cdot\|$ denotes Euclidean norm. Show that this classifier is a linear classifier and that it can be expressed in standard form as

$$\hat{y}(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x} + w_0)$$

   and express the weights $\mathbf{w}$ and bias w_0 in terms of the vectors μ_1 and μ_{-1}.

   b. Show that the hyperplane that describes the classifier in part (a) is the perpendicular bisector of the line joining the vectors μ_1 and μ_{-1}.

   c. Implement the above classifier on the linearly separable data sets Classify-2D-wLabels-1.txt and Classify-2D-wLabels-2.txt using all 250 points as training data and plot the classifier on the scatter plot for the data points (color-coding the points in the two categories). Obtain the mis-classification rate (the fraction of errors) obtained in each case.

   d. Implement the above classifier on the data sets Classify-3DwLabels-2.txt and Classify-3DwLabels-3.txt. Recall you used these data sets in Problem 3 of Homework 3 for your soft-SVM classifier. For the best hyperparameter you found in Part b of that problem, compare the results you obtain for the new classifier described in part (a) of this problem with the soft-SVM classifier in terms of mis-classification rates.

2. Solve the following problems about kernels.

   a. Show that for $\mathbf{x}, \mathbf{x}' \in R^d$,

$$K(\mathbf{x}, \mathbf{x}') = \sum_{i=1}^{d} \cos(x_i - x_i')$$

   where x_i and x_i' are the i^{th} elements of $\mathbf{x}$ and $\mathbf{x}'$, resp., is a positive definite symmetric kernel.

b. Show that for $\mathbf{x}, \mathbf{x}' \in R^d$,

$$K(\mathbf{x}, \mathbf{x}') = (c + \mathbf{x}^T\mathbf{x}')^p$$

is a positive definite symmetric kernel when $c > 0$. Find its associated mapping Φ explicitly (i.e., how must $\mathbf{x}$ be transformed to $\Phi(\mathbf{x})$? What is the dimension of $\Phi(\mathbf{x})$ as a function of p and d?

3. In this problem, we will extend the classifier of Problem 1 so that it can have non-linear decision boundaries and implement it on data sets you worked with in previous homeworks.

   a. Show that the classifier in Problem 1 depends on the test input $\mathbf{x}$ and input vectors $\{\mathbf{x}_i\}$ only through the inner products $< \mathbf{x}, \mathbf{x}_i >$. Moreover, show that the offset parameter w_0 depends on the input vectors $\{\mathbf{x}_i\}$ only through the inner products $< \mathbf{x}_i, \mathbf{x}_j >$ for $i, j \in [1 : m]$. Hence, the classifier can be kernelized with any kernel $K : \mathcal{X} \times \mathcal{X} \to R$ by replacing $< \mathbf{x}, \mathbf{x}_i >$ with $K(\mathbf{x}, \mathbf{x}_i)$.

   b. Use the non-homogeneous polynomial kernel $K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x}^T\mathbf{x}')^p$ to do classification in its associated Polynomial(p) feature space. Modify your computer program (i.e., classifier implementation) from Problem 1 to one that uses Polynomial(p) feature space using the results in Part (a) of this problem. There should be no explicit inner product computations of the form $< \phi(\mathbf{x}), \phi(\mathbf{x}_i) >$ but rather these should be efficiently computed as $K(\mathbf{x}, \mathbf{x}_i)$ in your program. Similarly, the bias parameter w_0 should be computed via the inner products $K(\mathbf{x}_i, \mathbf{x}_j)$.

   c. Using all 250 points in the data set Classify-2D-wLabels-3.txt, implement the kernelized classifier of Part (b) and report mis-classification rates for $p = 1, 2, 3, 4$. Compare with the hard SVM results you obtained in Problem 2 of Homework 3 for $p = 3, 4$. Re-run the hard SVM algorithm using all 250 data points in this comparison.

4. In this problem, you will kernelize the soft-SVM classifier you implemented in Problem 3 of Homework 3.

   a. Specifically, use the non-homogeneous polynomial kernel

   $$K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x}^T\mathbf{x}')^p$$

   to kernelize your soft-SVM classifier. Nowhere in your program should you explicitly compute inner products in feature space. Your algorithm must instead evaluate the kernel at pairs of (training and/or test) inputs.

   b. Use the training data in MNIST for optical character recognition (OCR) to solve the following binary classification problems:

      (i) Classify the letters 1 and 7 by working with the subset of the training set with just inputs having labels 1 and 7. Using the validation data set corresponding to these labels, evaluate the empirical validation loss as a function of the C hyperparameter and find the value of C that gives you the best validation loss. Report the "confusion matrix" that includes the numbers of false negatives, false positives, true negatives, and true positives in a 2×2 matrix.

(ii) Repeat the experiment of part (i) for the subset of the training set with inputs corresponding to labels 3 and 8.

(iii) Use the kernelized classifier of Part b of Problem 3 for the subsets of training data in parts (i) and (ii) and obtain validation losses (in the form of confusion matrices) in the two cases. Compare with the results you obtain for the soft-SVM in parts (i) and (ii).