E9 205 – Machine Learning for Signal Processing

Homework # 4
Due date: April 7, 2022

Analytical in writing and submitted in class or with TA.
Source code also need to be included.
Name of file should be "Assignment4_FullName.pdf" submitted to teams channel.
Assignment should be solved individually without consent.

March 30, 2022

1. **Kernel LDA** Deepak has learnt about linear discriminant analysis in his course. In a job interview, he is asked to find a way to perform dimensionality reduction in non-linear space. Specifically, he is given a set of N data points $\{\boldsymbol{x}_1,\ \boldsymbol{x}_2, .., \ \boldsymbol{x}_N\}$ and a non-linear transformation $\boldsymbol{\phi}(\boldsymbol{x})$ of the data. When he is asked is to define LDA in the non-linear space, he defines the within-class and between-class scatter matrices for a two-class problem as,

$$\begin{aligned} \boldsymbol{S}_B &= (\boldsymbol{m}_2^{\phi} - \boldsymbol{m}_1^{\phi})(\boldsymbol{m}_2^{\phi} - \boldsymbol{m}_1^{\phi})^T \\ \boldsymbol{S}_W &= \sum_{k=1}^{2} \sum_{n \in C_k} \left[\boldsymbol{\phi}(\boldsymbol{x}_n) - \boldsymbol{m}_k^{\phi}\right]\left[\boldsymbol{\phi}(\boldsymbol{x}_n) - \boldsymbol{m}_k^{\phi}\right]^T \end{aligned}$$

where $\boldsymbol{m}_k^{\phi} = \frac{1}{N_k}\sum_{n \in C_k} \boldsymbol{\phi}(\boldsymbol{x}_n)$ for $k = 1, 2$ and C_k denotes the set of data points belonging to class k. He also defines the Fisher discriminant as

$$J = \frac{\boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{S}_W \boldsymbol{w}}$$

where $\boldsymbol{w}$ denotes the projection vector. He goes on to say that he can solve the generalized eigen value problem to find $\boldsymbol{w}$ which maximizes the Fisher discriminant. At this point, the interviewer suggests that $\boldsymbol{\phi}(\boldsymbol{x})$ can be infinite dimensional and therefore LDA suggested by Deepak cannot be performed. Deepak counters by saying that he could solve for the LDA using kernel function $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\phi}(\boldsymbol{x}_i)^T \boldsymbol{\phi}(\boldsymbol{x}_j)$. He goes on and shows that LDA can indeed be formulated in a kernel space and the projection of a new data point can be done using kernels (without computing $\boldsymbol{\phi}(\boldsymbol{x})$). How would you have found these two solutions if you were Deepak ? (**Points** 15)

2. By definiton, a kernel function $k(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \boldsymbol{\phi}(\boldsymbol{x})^T \boldsymbol{\phi}(\hat{\boldsymbol{x}})$. A neccessary and sufficient condition for defining a kernel function is that the Gram matrix $\boldsymbol{K}$ is positive definite. Using

either of these definitions, prove the following kernel rules

$$\begin{aligned}
k(\boldsymbol{x}, \hat{\boldsymbol{x}}) &= ck_1(\boldsymbol{x}, \hat{\boldsymbol{x}}) \\
k(\boldsymbol{x}, \hat{\boldsymbol{x}}) &= f(\boldsymbol{x})k_1(\boldsymbol{x}, \hat{\boldsymbol{x}})f(\hat{\boldsymbol{x}}) \\
k(\boldsymbol{x}, \hat{\boldsymbol{x}}) &= \boldsymbol{x}^T \boldsymbol{A}\hat{\boldsymbol{x}} \\
k(\boldsymbol{x}, \hat{\boldsymbol{x}}) &= k_1(\boldsymbol{x}, \hat{\boldsymbol{x}}) + k_2(\boldsymbol{x}, \hat{\boldsymbol{x}}) \\
k(\boldsymbol{x}, \hat{\boldsymbol{x}}) &= k_1(\boldsymbol{x}, \hat{\boldsymbol{x}})k_2(\boldsymbol{x}, \hat{\boldsymbol{x}})
\end{aligned}$$

where k_1,k_2 denote valid kernel functions, $c > 0$ is any scalar, $f(\boldsymbol{x})$ is any scalar function and $\boldsymbol{A}$ is symmetric positive definite matrix.

(**Points 5)**

3. **One-class SVM** Let $\boldsymbol{X} = \{\boldsymbol{x}_1,\ \boldsymbol{x}_2, .., \ \boldsymbol{x}_l\}$ be dataset defined in $\mathbb{R}^n$. An unsupervised outlier detection method consist of finding a center $\boldsymbol{a}$ and radius R of the smallest sphere enclosing the dataset in the high dimensional non-linear feature space $\boldsymbol{\phi}(\boldsymbol{x})$. In a soft margin setting, non-negative slack variables ζ_j (for $j = 1, .., l$) can be introduced such that, $||\boldsymbol{\phi}(\boldsymbol{x}_j) - \boldsymbol{a}||^2 \leq R^2 + \zeta_j$

   The objective function in this case is to minimize radius of the sphere with a weighted penalty for slack variables, i.e., $R^2 + C \sum_{j=1}^{l} \zeta_j$ where C is a penalty term for allowing a trade-off between training errors (distance of points outside the sphere) and the radius of the smallest sphere.

   (a) Give the primal form Lagrangian and the primal constraints for the one-class SVM. (**Points** 5)

   (b) Find the dual form in terms of kernel function and the KKT constraints for the one-class SVM. What are the support vectors ? Will support vectors change when $C > 1$ is chosen ? Give a numerically stable estimate of R (**Points** 10)

   (c) For a new data point $\boldsymbol{x}$, how will we identify whether it is an outlier or not (using kernel functions) ? (**Points** 5)

4. Use the following data source for the remaining two questions
   leap.ee.iisc.ac.in/sriram/teaching/MLSP22/assignments/data/Data.tar.gz
   Implementing Linear SVMs - 15 subject faces with happy/sad emotion are provided in the data. Each image is of 100 × 100 matrix. Perform PCA to reduce the dimension from 10000 to K. Implement a classifier on the training images with linear kernel based support vector machine. One potential source of SVM implementation is the LIBSVM package
   http : //www.csie.ntu.edu.tw/ cjlin/libsvm/

   (a) Use the SVM to classify the test images. How does the performance change for various choice of kernels, parameter C and ϵ. How does the performance change as a function of K.

   (b) Compare the SVM classifier with LDA classifier and comment on the similarity and differences in terms of the problem formulation as well as the performance.

(**Points** 20)

5. **Supervised Sentiment Analysis** - Download the movie review data (each line is a individual review)
$http : //www.leap.ee.iisc.ac.in/sriram/teaching/MLSP22/assignments/movieReviews1000.txt$

a Split the data into two subsets. One for training (first 3000 reviews) and the other for testing (last 1000 reviews).

b Use TF-IDF features and train PCA (using the training data) to reduce the data to 30 dimensions.

c Split the training data randomly into set of 2500 for model training and 500 for validation. Train a logistic regression model. Implement the stochastic gradient descent algorithm by hand without using any tools. Show the loss value for each epoch on the training and validation dataset (for 20 epochs). Test on the test data and report the performance in terms of review classification accuracy. Compare the performance for different choices - batch size [32,64,128], learning rate [1e-3,1e-2, 1e-1]. Show the loss curves for each case.

d Implement the logistic regression with L2 weight regularization. Show the loss curves for regularization coefficient value of $1e-2$, $1e-1$ and 1. Do you see any overfitting/under-fitting for any of these choices. Test on the test data and report the performance on the test set in terms of review classification accuracy.

(**Points** 40)