E9 205 – Machine Learning for Signal Processing

Homework # 3
Due date: Oct. 22, 2018

Analytical submitted in class and coding submitted by email.
Coding assignment submitted to mlsp18 doT iisc aT gmail doT com

October 12, 2018

1. Use the following data source for the remaining two questions
leap.ee.iisc.ac.in/sriram/teaching/MLSP_18/assignments/HW3/Data.tar.gz
Implementing 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 support vector machines. 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) Will the SVM classifier perform better if an LDA is applied at the input.

   (c) Provide your answers with analysis, plots and wrapper code for SVM training and testing.

   (**Points** 25)

2. **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** 20)

3. 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

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

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 10**)

4. **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** 15)

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

5. **Extending k-means** The k-means algorithm for a dataset $\mathbf{X} = [\mathbf{x}_1\ \mathbf{x}_2\ ..\ \mathbf{x}_N]$ consisting of N data points $\mathbf{x} \in \mathbb{R}^D$ is the problem of finding k disjoint clusters $C_1, C_2, .., C_k$ with

$$\min_{C_1,..,C_k} \sum_{i=1}^{k} \sum_{\mathbf{x} \in C_i} ||\mathbf{x} - \boldsymbol{\mu}_i||^2$$

   where $\boldsymbol{\mu}_i = \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{x}$ denotes the mean vector for cluster C_i and $|C_i|$ is the cardinality (number of elements) of cluster C_i.

(a) Let $\mathbf{B} \in \{0,1\}^{N \times k}$ denote a binary cluster membership matrix where $B_{ni} = 1$ when the n-th data point is associated with i-th cluster and $B_{ni} = 0$ otherwise. Also, let $\mathbf{D} = diag\{\frac{1}{|C_1|}, \frac{1}{|C_2|}, .., \frac{1}{|C_k|}\}$ be a diagonal $k \times k$ matrix. Show that the optimization for k-means can be equivalently represented as

$$\max_{B} tr(\mathbf{X}^T\mathbf{X}\mathbf{B}\mathbf{D}\mathbf{B}^T)$$

under the constraint that $\mathbf{B}^T\mathbf{B} = \mathbf{D}^{-1}$. (**Points 10**)

(b) Let $\mathbf{W}$ denote the Gram matrix corresponding to $\phi(\mathbf{x})$ ($[W]_{i,j} = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$). Assume that $[W]_{i,j} \geq 0,\ \forall\ i, j$. Also let $\mathbf{G} = \mathbf{B}\mathbf{D}^{\frac{1}{2}}$. Show that kernel k-means defined above is equivalent to NMF $\mathbf{W} \approx \mathbf{G}\mathbf{G}^T$ with divergence cost given by 2-norm and the additional constraint of $\mathbf{G}^T\mathbf{G} = \mathbf{I}$. (**Points 10**)