EECE5644 Summer 1 – Exam 4

Start: Friday, 2019-June-07
Submit: Saturday, 2019-June-15 before 12:00 (noon)

This is the fourth take-home exam of four that we will have this term. Please submit your solutions to erdogmus@ece.neu.edu as email attachments:

- A single PDF file that includes relevant math derivations and results from computer implementations in the form of figures or tables as appropriate;
- A single ZIP file that includes your code that generated the results presented in the PDF.

Note that this is a graded assignment and the entirety of your submission must contain only your own work. When constructing your solutions, you may use written or other forms of literature and software packages available to you, as long as these sources are properly acknowledged in your submission as cited references. Make sure that you use such resources properly (i.e. especially for software packages, understand what they do and make sure they do what you think they do).

Question 1

Train and test Support Vector Machine (SVM) and Multi-layer Perceptron (MLP) classifiers that aim for minimum probability of classification error (i.e. we are using 0-1 loss; all error instances are equally bad). You may use a trusted implementation of training, validation, and testing in your choice of programming language. The SVM should use a Gaussian (sometimes called radial-basis) kernel. The MLP should be a single-hidden layer model with logistic nonlinearities (see hint for Q2) for all perceptrons.

Generate 1000 independent and identically distributed (iid) samples for training and 10000 iid samples for testing. All data for class $l \in \{1,2\}$ should be generated as follows:

$$\mathbf{x} = r_l \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} + \mathbf{n} \tag{1}$$

where $\theta \sim Uniform[-\pi,\pi]$ and $\mathbf{n} \sim N(\mathbf{0},\sigma^2\mathbf{I})$. Use $r_1 = 2, r_2 = 4, \sigma = 1$. Sample Matlab script is provided.

Note: The two class sample sets will be highly overlapping two concentric disks, and due to angular symmetry, we anticipate the best classification boundary to be a circle between the two disks. Your SVM and MLP will try to approximate it.

Use the training data with 10-fold cross-validation to determine the best hyperparameters (overlap penalty weight and Gaussian kernel width for the SVM, number of perceptrons in the hidden layer for the MLP). Once these hyperparameters are set, train your final SVM and MLP classifier using the entire training data set. Apply your trained SVM and MLP classifiers to the test data set and estimate the probability of error from this data set.

Report the following: (1) visual and numerical demonstrations of the K-fold cross-validation process indicating how the hyperparameters for SVM and MLP classifiers are set; (2) visual and numerical demonstrations of the performance of your SVM and MLP classifiers on the test data set. It is your responsibility to figure out how to present your results in a convincing fashion to indicate the quality of training procedure execution, and the test performance estimate.

Hint: For hyperparameter selection, you may show the performance estimates for various choices and indicate where the best result is achieved. For test performance, you may show the data and classification boundary superimposed, along with an estimated probability of error from the samples. Modify and supplement these ideas as you see appropriate.

Question 2

Generate two-dimensional $\mathbf{x} = [x_1, x_2]^T$ samples with the attached Matlab script (the data is generated through iid sampling from a mixture of three Gaussians). Specifically generate 1000 samples for training and 10000 samples for testing.

Train and test a single hidden layer MLP function approximator to estimate the value of x_2 from the value of x_1 by minimizing the mean-squared-error (MSE) on the training set.

Using 10-fold cross-validation to select between logistic (sigmoid) and softplus (SmoothReLu) nonlinearities for the perceptrons in the hidden layer, as well as the number of perceptrons. Leave the output layer linear (no nonlinearity). Once the best model architecture is identified using cross-validation, train the selected model with the entire training set. Apply the trained MLP to the test dataset. Estimate the test performance.

Report the following: (1) visual and numerical demonstrations of the cross-validation process indicating how the model selection was carried out; (2) visual and numerical demonstration of the performance of the trained model on the test data set.

Hint: $logistic(z) = 1/(1+e^{-z}) \quad \& \quad softplus(z) = ln(1+e^{z})$