EECE5644 Fall 2020 – Take Home Exam 4

Submit: Thursday, 2020-December-03 before 10:00ET (morning)

Please submit your solutions on Canvas in a single PDF file that includes all math, numerical and visual results, and code (as appendix or as a link to an online code repository). Only the contents of this PDF will be graded. Do not link from the pdf to external documents online where results are presented.

This is a graded assignment and the entirety of your submission must contain only your own work. You may benefit from publicly available literature including software (not from classmates), as long as these sources are properly acknowledged in your submission.

Copying math or code from each other are not allowed and will be considered as academic dishonesty, and will be treated. There cannot be any written material exchange between classmates, but verbal discussions are acceptable. Discussing with the instructor, the teaching assistant, and classmates at open office periods to get clarification or to eliminate doubts are acceptable.

By submitting a PDF file in response to this take home assignment you are declaring that the contents of your submission, and the associated code is your own work, except as noted in your citations to resources.

Question 1 (30%)

The data generation script for this question is called exam4q1_generateData.m. Generate 1000 training sample pairs and 10000 test sample pairs using this function. Assuming that $y = f(x) + v$ where v is assumed to be a zero-mean σ^2-variance additive Gaussian noise, train a single hidden layer MLP with a first layer nonlinearity of your choice (e.g., logistic as a sigmoid choice, or softplus as a smooth ReLu style choice). This model will approximate the y values as functions of x in the form of a neural network.

For parameter optimization, use the maximum likelihood parameter estimation method, which will simplify to minimum mean squared error (MSE) optimization under the assumed data model. Use 10-fold cross-validation to select the best number of perceptrons in the first layer (using minimum average MSE on validation partitions across the 10 experiments as the model selection criterion). Report the average MSE of the best model in this 10-fold cross-validation experiment. Demonstrate visual results and explanations indicating how model selectioin has been conducted.

Once the best model structure is identified using cross-validation, train an MLP with the selected number of perceptrons with the entire training set. Apply the trained MLP to the test set and visualize the predictions of the model overlaid on the test data samples in a scatter plot. Also calculate and report the MSE of the model on the test dataset. You may use existing software packages for all aspects of this solution. Make sure to clearly demonstrate that you are using the packages properly.

Hint: We used the logistic function earlier. If you choose softplus as your nonlinearity, it is $softplus(z) = ln(1 + e^z)$ *Note: The theoretical minimum-MSE estimator is the conditional expectation of y given x, and the neural network model you constructed here is an approximation of that function.*

Question 2 (35%)

For this question use the *generateMultiringDataset.m* function to sample traning and testing data. Generate a two-class training set with 1000 and testing set with 10000 samples. Train and evaluate a support vector machine classifier with a Gaussian kernel (radial-basis function (RBF) kernel) on these datasets. Specifically, use a spherically symmetric Gaussian/RBF kernel.

Using 10-fold cross-validation, select the best box constraint hyperparameter C and the Gaussian kernel width parameter σ (notation based on previously covered math in class from the SVM tutorial). Use minimum-average-cross-validation-probability-of-error to select best hyperparameters. Train a final SVM using the best combination of hyperparameters with the entire training set. Classify the testing dataset samples with this trained SVM to assess performance; estimate the probability error using the test set. Demonstrate numerical and visual results and explain how you trained and evaluated your SVM classifier.

Question 3 (35%)

In this question, you will use GMM-based clustering to segment the color images *3096_color.jpg* and *42049_color.jpg* from the Berkeley Image Segmentation Dataset. We will refer to these images as the airplane and bird images, respectively. As preprocessing, for each pixel, generate a 5-dimensional feature vector as follows: (1) append row index, column index, red value, green value, blue value for each pixel into a raw feature vector; (2) normalize each feature entry individually to the interval $[0,1]$, so that all of the feature vectors representing every pixel in an image

fit into the 5-dimensional unit-hypercube. All segmentation algorithms should operate on these normalized feature vectors. For each image do the following: (1) Using maximum likelihood parameter estimation, fit a GMM with 2-components, use this GMM to segment the image into two parts; (2) Using 10-fold cross-validation, and maximum average validation-log-likelihood as the objective, identify the best number of Gaussian components (clusters), then fit a new GMM with this best number of components and use this GMM to segment the image into as many parts as there are number of Gaussians. For GMM-based clustering, use the GMM components as class/cluster-conditional pdfs and assign cluster labels using the MAP-classification rule. Present the original images and your GMM-based segmentation labels (in the form of an image for easy visual assessment of results) side by side.

Hint: To get started, see the Matlab code for 2018 Summer 2, Exam3 Question 1 in
$G/Practice/EECE5644\_2018Summer2/Exam3Q1.m$
to see how K-means clustering is used to segment both grayscale and color versions of these images.