CMPG-765/CMPT- 465 Neural Networks and Learning Systems
Homework-6

Assignment for graduate students.

You may get 130 points

"**Iris**" is a famous, very popular benchmark 3-class classification problem.

The "Iris" dataset consists of 150 instances, 50 instances per each of 3 classes. These classes correspond to 3 different types of iris flowers.

Each data instance is characterized by 4 features followed by the class label (desired output).

In this homework, you will test 3 classifiers (MLP with 3 output neurons, SVM with polynomial, Gaussian and linear kernels, and MLMVN-LLS-SM) using the "Iris" dataset and employing **k-fold cross-validation**.

The Iris_Data.mat file, which is shared with you, contains all data sets which you need for your experiments.

1. (**5 points** if you do this by hands **or 20 points** if you design a function to do this). Divide each one of 3 datasets (for MLP, SVM, and MLMVN) into 5 folders containing 30 samples each and 10 samples from each of 3 classes, respectively.
2. (**20 points**) Run 5 learning/testing experiments with MLP using each of 5 folders as a test set, while other 4 folders should be merged into a learning set.
   Use 4 hidden neurons in all these experiments.
   Increase the number of hidden neurons and repeat your experiments for at least 3 different network topologies (with 3 different numbers of hidden neurons).
   You should use functions, which were shared with you - LearningMLPkOut for learning and TestingMLP3kOut for testing.
3. (**20 points**) Run 5 learning/testing experiments with SVM using each of 5 folders as a test set, while other 4 folders should be merged into a learning set.
   Repeat these experiments for three different SVM kernels ('gaussian', 'polynomial', and 'linear').
   You should use Matlab functions to simulate SVM: **templateSVM** to determine an SVM kernel, **fitcecoc** for learning, and **predict** for testing – you should do that similarly to how it was done in demo_SVM_Iris Matlab program, which was shared with you.

10 extra credit points will be added if you design a program where all 5 learning/testing experiments with SVM will be organized and performed by running that program.

4. (**20 points**) Run 5 learning/testing experiments with MLMVN using each of 5 folders as a test set, while other 4 folders should be merged into a learning set.

   Use 4 hidden neurons in all these experiments. Use 0.39 as an angular global threshold and 0 as a local threshold.

   Increase the number of hidden neurons and repeat your experiments for at least 3 different network topologies (with 3 different numbers of hidden neurons).

   You should use MLMVN simulator supporting a network with k output neurons, which was shared with you.

5. (**10 points**). Find average classification rates for your five experiments with MLP (separately for each topology), SVM (separately for each kernel), and MLMVN (separately for each topology). Create a table with a summary of your results: a classification rate for each experiment with MLP, a classification rate for the same folder for each experiment with SVM, a classification rate for each experiment with MLMVN and average classification rates over all experiments with MLP, SVM, and MLMVN, respectively.

6. (**25 points**) Run 5 learning with validation/testing experiments with MLMVN (use NetLearnL.m function for learning with validation). You should use here a single network topology, which showed the best results in # 4.
   To create a validation set for each of your experiments, you need to split a selected 'test' folder into two subfolders (15 samples from this folder, 5 from each class, shall be put in a corresponding validation set, while remaining 15 samples, 5 from each class, shall be put in a corresponding test set).
   You need to run 5 experiments creating a validation set and a test set from each of 5 data folders , while other 4 folders should be merged into a learning set.
   Find average classification rates for your five experiments with MLMVN and learning with validation.