Project Part 2 [10 points] Experimenting with SVM (Due Nov. 16 2021, 11:59pm)

In this project, you will study how to use a common SVM package through doing some classification tasks.

Data Set: The given dataset contains 50 categories/classes. The **training set** has 4786 samples in the file 'trainData.mat', and the **testing set** has 1833 samples in the file 'testData.mat'. Each sample is described by the rows of 3 different **feature** matrices i.e., $\boldsymbol{X}_1$, $\boldsymbol{X}_2$, and $\boldsymbol{X}_3$ in the corresponding file, and the category vector is always $\boldsymbol{Y}$. All the 3 features are normalized histograms, which means the elements are non-negative and the sum of each feature equals to 1 (i.e., $\sum_j \boldsymbol{X}_k(i,j) \equiv 1$).

You may use the following piece of code to read the dataset in Python (or you may use the load filename command in Matlab, since these are .mat files):

```
import scipy.io
data = scipy.io.loadmat('matlabfile.mat')
```

SVM Package:

There are many SVM toolboxes. In this project, you will use libSVM. You can either use Python or Matlab for this project.

For Python users, the installation instructions are provided on the following link:
http://www.csie.ntu.edu.tw/~cjlin/libsvm/

For Matlab users, you can directly use the code in folder 'libSVM'. Run 'make.m' to install on your PC. The instructions and examples on how to use the package can be found in the file 'README0'.

We will use the function *svm_train* in Python (*svmtrain* in Matlab) to train and *svm_predict* in Python (*svmpredict* in Matlab) for prediction.

NOTE: You may fix the penalty parameter to '-c 10' and use linear kernel '-t 0' for *svm_train/svmtrain*.

Step 0: Classification by individual features.

Output: The classification accuracy for the testing set in the follow cases (1) and (2).

Instructions:

(1) For each of the 3 features in the training set, $\boldsymbol{X}_k$ $(1 \le k \le 3)$, train a multi-class linear SVM classifier, i.e., $h_k(\mathbf{x})$. Get the prediction result of $h_k(\mathbf{x})$ based on the same feature $\boldsymbol{X}_k$ in the testing set and compare to $\boldsymbol{Y}$ for computing the classification accuracy.

(2) Based on the SVM classifiers $h_k(\mathbf{x})$, we can also obtain $p_k(w_i|\mathbf{x})$, the (posterior) probability of sample $\mathbf{x}$ that it belongs to the i-th category (w_i) according to feature $\boldsymbol{X}_k$ $(1 \leq k \leq 3)$. This can be done by using the parameter '-b 1' option in training and testing (check http://www.csie.ntu.edu.tw/~cjlin/libsvm/ for more details). Train the SVM classifiers with this option and report the classification accuracies on the testing set based on the 3 features respectively.

Step 1: Feature combination by fusion of classifiers.

Output: The classification accuracy in the testing set and compare it to that of (2) in Step 0.

Instructions: Directly combine the 3 SVM classifiers with probability output i.e., $p_k(w_i|\mathbf{x})$ $(1 \leq k \leq 3)$, in (2) of Step 0. Combine the 3 classifiers by probability fusion as $p(w_i|\mathbf{x}) = \sum_k p_k(w_i|\mathbf{x})/3$. The final recognition result is $w_{i*} = \mathrm{argmax}_i\, p(w_i|\mathbf{x})$.

Step 2: Feature combination by simple concatenation.

Output: The classification accuracy in the testing set and compare it to that of (1) in Step 0.

Instructions: Directly concatenate the 3 features $\mathbf{X}_k, 1 \leq k \leq 3$ to form a single feature, i.e. $\mathbf{X} = [\mathbf{X}_1\,, \cdots, \mathbf{X}_K]$; train a linear SVM classifier based on $\mathbf{X}$ and obtain the classification accuracy for the testing set.

What to submit:

1. Your code for the above steps.
2. A report summarizing the results, e.g., the testing accuracies. Include in your report brief description of the tasks and steps for them, any intermediate results/observations that you deem helpful for illustrating the partial results.

Note: There is no minimum or maximum length requirement for the report. Writing the report is the opportunity for you to reflect on your understanding of the problems/tasks through organizing your results.

3. The report should be typed (*handwritten reports are not allowed*) and in a .pdf format.

The data files for the project are uploaded in the Files/Assignments folder:
trainData.mat
testData.mat