Fall 2021: CSCI 597 Machine Learning Design

Assignment #2 To be completed in Python

As a simplified example of character recognition, we will compare several supervised learning classifiers with validation on a larger version of the MNIST digit recognition dataset. In this assignment we will use a much larger dataset than that used for assignment 1; this should represent a better distribution of the natural variability in hand written 8s and 9s.

Download (from moodle), NumberRecognitionBigger.mat. Note the dataset includes data samples for all handwritten digits 0 to 9, but we will be using only 8 and 9 for this assignment.

As machine learning application developers, you will be expected to be able to read online reference information for any given machine learning technique that is publicly available. It is an extremely useful skill to be able to implement models that make use of existing learning algorithms based on reading their respective documentations online. We strongly recommend trying this first, then asking the TA for assistance to overcome challenges that you encounter (through the MS Teams TA Questions group). Here are some example learning and validation packages that we recommend you make use of for this assignment.

Python

```
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier as RF
```

Scikit-learn also provides some useful official guides for using the above SVM classifier, and Random Forest classifier functions. Also strongly consider using:

```
from sklearn.model_selection import cross_validate
from sklearn.model_selection import StratifiedKFold, StratifiedShuffleSplit
```

And possibly using the `random_state` argument for either `StratifiedShuffleSplit` or `StratifiedKFold` to ensure each classifier is trained with the same data.

Question 1: Implement K-Fold cross validation (K=5). Within the validation, you will train and compare a Support Vector Machine (a linear kernel and a RBF kernel where the trainer sets the kernel parameter: gamma/sigma), a Random Forest (Number of Trees = 100), and a K-NN (K=1, K=5 and K=10) classifier. The validation loop will train these models for predicting 8s and 9s. NOTE: for a fair comparison, K-Fold randomization should only be performed once, with any selected samples for training applied to the creation of all classifier types (SVM, RF, KNN) in an identical manner (i.e. the exact same set of training data will be used to construct each model being compared to ensure a fair comparison).

Provide a K Fold validated error rate for each of the classifiers. Provide your code. Answer the following questions:

a) Which classifier performs the best in this task?

b) Why do you think this classifier outperforms the others?
c) How does KNN compare to the results obtained in assignment 1? Why do you observe this comparative pattern?

It was previously announced on multiple occasions that each student is required to assemble their own dataset compatible with supervised learning based classification (i.e. a collection of measurements across many samples/instances/subjects that include a group of interest distinct from the rest of the samples). If you are happy with your choice from assignment 1, then re-provide your answer to Assignment 1, Question 2 below. If you want to change your dataset for this assignment, for a future assignment or for your project, you are free to do so, but you have to update your answer to Question 2 based on your new dataset choice.

Question 2: (Repeat) Describe the dataset you have collected: total number of samples, total number of measurements, brief description of the measurements included, nature of the group of interest and what differentiates it from the other samples, sample counts for your group of interest and sample count for the group not of interest. Write a program that analyzes each measurement individually. For each measurement, compute the area under the receiver operating characteristic curve (AUC). Provide an output of the 10 leading measurements (highest AUC), making it clear what those measurements represent in your dataset (these are the measurements with the most obvious potential to inform prediction in any given machine learning algorithm), and what the corresponding AUC values are. Provide this code.

Question 3: Adapt your code from Question 1 to be applied to the dataset that you've organized for yourself. Provide the error rates for the different classifiers and your code. Answer the following question: is the best performing classifier from Question 1 the same in Question 3? Elaborate on those similarities/differences – what about your dataset may have contributed to the differences/similarities observed?

Deadline: October 15th, 2021.