PCSE 595 - Spring 2021

Assignment 3
Scikit-learn and Keras

Assignment 3 focuses on using existing packages (scikit-learn and keras) to perform classification on a dataset. The dataset is the Breast Cancer Wisconsin (Diagnostic) Data Set (https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29) which has been provided to you as wpbc.data.

Sample code is provided to help you, but you are not required to use it. I expect that the code will have to be modified significantly to efficiently run the experiments. The code is written for the iris dataset, so make sure that if you use it, you modify it for the breast cancer dataset.

The dataset and algorithms you are expected to use are shown below:

Dataset = Breast Cancer Wisconsin (Diagnostic) Data Set:

Data contained in "wpbc.data"
Binary classification task to predict if breast cancer are recurrent (R) or Nonrecurrent (N)
See wpbc.names for a more complete description of the dataset
Contains some missing values which need to be dealt with (I replaced them with a 0)

Algorithms

Scikit-learn (https://scikit-learn.org/stable/)

1. K Nearest Neighbors (KNN, neighbors.KNeighborsClassifier)
2. Ridge Regression (linear_model.Ridge)
3. Artificial Neural Network (ANN, neural_network.MLPClassifier)
4. Support Vector Machine (SVM, svm.SVC)
5. Decision Tree (tree.DecisionTreeClassifier)
6. Boosting (AdaBoost, ensemble.AdaBoostClassifier)

Keras (https://keras.io/):

1. Deep neural network

Note: you may get warnings when running Keras. That is OK as long as it runs.

Project Requirements

You will turn in a document and your code. The document will contain the following items specified by "**Report**". Use tables to summarize the results. Put a brief description of each table in the document so that I know what I am looking at.

For each scikit learn algorithm:

- Read the description online. Now that you have a theoretical understanding of the algorithms, hopefully most of it makes sense to you.

- Use 5-fold cross validation to determine the optimal hyperparameter settings for each algorithm. You must experiment with at least one hyper-parameter per algorithm (I put some in the code for you already. You don't have to try every possible hyperparameter, just one or maybe two for each algorithm).

    a. **Report in a table** the average cross-validation F1-score for the algorithm with all default values for hyper-parameters in a table
    b. **Report in a table** the best hyperparameters you found and the average cross-validation F1 score for the algorithm with the optimal hyperparameters you found in a table
    c. Run a t-test to determine if the optimal hyperparameters are statistically significantly different than the default values. **Report in a table** the p-value and your conclusion based on an alpha = 0.05.

- Once you've found the optimal hyperparameters for each algorithm, train the algorithm on the whole training set. Then, evaluate its performance on the test set.
    a. **Report in a table** each algorithm's performance on the test set
    b. Run a statistical significance test to determine the ranked (ties are ok) order of algorithms from best to worst on this dataset.
    c. **Report in a table** the p-values of each of the statistical significance tests and the conclusion you drew (an algorithm by algorithm table of tests is probably easiest).

For Keras:

- Play with the neural network architecture.
- Implement a 5 layer densely connected deep neural network
- Implement a 10-layer densely connected deep neural network
- Implement a 3 layer densely connected deep neural network
- Implement your own deep neural network (any architecture you want) to try and get a good classification result (just show me that you thought about the results and try and think of what might work well).
- **Report in a table** the F1-score of each network, and use statistical significance testing to determine which network performed the best. You should use 5-fold cross-validation to determine which performs the best (since the network architecture is more or less a hyper-parameter)

Lastly:

- Run a statistical significance test to determine if your best performing scikit-learn algorithm performed differently than your best performing Keras algorithm

- **Report** which algorithms you compare, their F1-scores, your p-value, and conclusion with alpha = 0.05. You should train on the full training set and use performance on the test set to make the conclusion.