CITS5508 Machine Learning
Semester 1, 2021

Lab Sheet 3

Assessed, worth 10%. Due: 11:59pm, Thursday 1st April 2021

1 Outline

This lab sheet consists of two small projects. The first project asks you to train two decision tree classifiers and compare their performance. The second project asks you to train and test two decision tree regressor and a support vector regressor. This lab sheet is a good practical exercise to test your understanding of the machine learning algorithms covered in Chapters 5–6.

2 Submission

Put your two projects together into a single Jupyter Notebook file and name it as **lab03.ipynb**. Submit it to LMS before the due date and time shown above. You can submit your file multiple times. Only the latest version will be marked.

3 Project 1

The data file and a brief description file of an E Coli bacteria dataset are available in a web repository managed by the UCI (University of California at Irvine) Centre for Machine Learning and Intelligent Systems. See

https://archive.ics.uci.edu/ml/datasets/ecoli

The spreadsheet data is in the `ecoli.data`[1] file and description about the data is in the `ecoli.names` file. You should save the file `ecoli.data` to the same directory with your Jupyter Notebook file. Although the data file name does not end with "*.csv*", you should be able to read it in using the `read_csv` function as usual. You will need to hard code the column headings in your Python code as they are not included in the data file. The class label is in the last column.

Your tasks for this project are:

1. After reading in the `ecoli.data` file, provide a plot for data visualisation.

   Out of the eight classes, you should remove those classes having less than 10 instances as it is not possible to classify them. You will also need to justify what to do with non-numerical data. Write an appropriate function that carries out these two cleaning operations.

2. You should implement two Decision Tree classifiers, one being trained on the raw (unscaled) features and another on the scaled features computed by `StandardScaler`. For each classifier, use `grid search` with 3-fold cross validation on three of the following four hyperparameters: `criterion`, `max_depth`, `min_samples_leaf`, and `max_features`. Investigate 2-3 different values for each hyperparameter.

[1] If you use the MacOS, the operating system may automatically rename the downloaded file as *ecoli.data.txt*. You will need to rename the file back to *ecoli.data*. Otherwise, your code would not work when we mark it.

3. Compare the average F1-scores of your two classifiers from the 3-fold cross validation. For visualisation purpose, display also the confusion matrices (using any suitable function from Scikit-learn) of these classifiers.

Hints:

- When calling the function `read_csv`, there is an optional argument that you can use to specify the character that separates the columns of the data.
- The data cleaning process for this project involves removing some data instances. You can use various functions, such as `loc` and `drop`, from the `pandas.DataFrame` package to remove rows and columns of the `DataFrame` object. Just beware that even after some rows are correctly removed, the index locations of the remaining rows in the `DataFrame` object may not be automatically updated. This means that you would still be able to access the removed rows and you would see NAN (meaning *not a number*, i.e., *undefined*) for those rows. To overcome this problem, you will need to explicitly call the `reset_index` function to renumber the index locations. Alternatively, you can set the `inplace` parameter appropriately when you call the `drop` function.

4 Project 2

The UCI Machine Learning Repository webpage below is a *Concrete Slump Test* dataset:

https://archive.ics.uci.edu/ml/datasets/Concrete+Slump+Test

suitable for testing the performance of regressors. The data file `slump_test.data` has 11 columns, including 7 input columns and 3 output columns. In this project, we will try to use the 7 input columns to predict the last output column (`Compressive Strength (28-day)(Mpa)`). So the feature dimension is 7. You should copy the data file `slump_test.data` and put it in the same directory with your **lab03.ipynb** file and use an appropriate function from `pandas` to read the data.

Your tasks for this project are:

1. After reading in the data file, show a plot of the data for visualisation. Similar to project 1, write a small function which should perform the data preparation steps to form your feature matrix `X` and the ground truth output array `y`.
2. Implement three regressors:

   - a *decision tree regressor* trained on the raw features;
   - a *decision tree regressor* trained on the scaled features computed by `StandardScaler`;
   - a *support vector regressor* trained on the same scaled features above.

   For the two DT regressors, set `criterion` to `mse` and use *grid search* to experiment with two more hyperparameters, each of which should have 3 values. For the support vector regressor, choose 3 hyperparameters from the list below: `kernel`, `C`, `gamma`, `max_iter`. Each hyperparameter should have 2 to 3 values. You should use 3-fold cross validation for all the three regressors.

   Since the operations carried out by the three regressors are very similar, you should try to modularise your code, i.e., you should write a function which accepts appropriate arguments so this function can be used by all the three regressors.
3. Your Python code should output the mean squared error of the prediction of each regressor from the 3-fold cross validation. A brief comparison of the performance of these regressors should be included in your markdown cell(s). A figure (which may contain subplots) illustrating the prediction results from the three regressors with a brief explanation should also be included.

5 Penalty on late submissions

See the URL below about late submission of assignments:
https://ipoint.uwa.edu.au/app/answers/detail/a_id/2711/~/consequences-for-late-assignment-submission