CCE3503—Practical Machine Learning

Assignment 2: Multi-label classification

Trevor Spiteri
trevor.spiteri@um.edu.mt

Last updated: 2024-11-27

Aim

The aim of this assignment is to train neural networks as classifiers for a multi-label problem, perform cross-validation and hyperparameter optimization, and evaluate the performance using suitable metrics.

Dataset

In this assignment, you will be using the multi-label yeast dataset, which can be downloaded from the VLE. The Yeast dataset is formed by micro-array expression data and phylogenetic profiles with 2,417 samples. There are 103 descriptive features per gene. Each gene is associated with a set of functional classes. In this version of the dataset there are 14 functional classes and a gene can be associated with any number of these.

Important note: It is part of your task to split this dataset into training and testing subsets if and when required during this assignment. Make sure to clearly point this out at the relevant stage or stages.

Requirements

- The assignment involves some Python 3 coding using Jupyter notebooks.
- You can use any of the NumPy, SciPy, pandas, Matplotlib, Seaborn, scikit-learn, and scikit-multilearn libraries.

- Use markdown cells to separate task sections and to include text required by the tasks below.

Tasks

1 Problem transformation (30 marks)

- Comment on the difference between the binary relevance approach and the classifier chains approach.
- With a fixed neural network architecture, use the binary relevance approach to obtain and train a multi-label classifier for the yeast dataset.
- With the same fixed neural network architecture, use the classifier chains approach to obtain and train a multi-label classifier for the yeast dataset.

2 Adapted algorithm (40 marks)

In this task you are to use hyperparameter optimization (HPO) with K-fold cross-validation to obtain a suitable neural network.

In this task, you are *not* required to adapt the problem transformation approaches of Task 1, but to adapt a neural network.

Important: Since this is a multi-label classification problem, make sure that the neural network is adapted so that its outputs are suitable for multi-label classification, and comment on how this is achieved/confirmed.

- Determine at least three hyperparameters that can be optimized when selecting a suitable neural network. Comment on your choice of hyperparameters to optimize.
- Select an HPO technique supported by the scikit-learn toolkit. Comment on your choice.
- HPO is to be performed using K-fold cross-validation. Choose a suitable value of K for the cross-validation, and comment on your choice.
- Using HPO, obtain a neural network multi-label classifier for the dataset.

3 Performance evaluation (20 marks)

- Select suitable evaluation metrics for the multi-label classification problem of the assignment. Comment on the choice of metrics.
- Use the selected evaluation metrics to compare the performance of
  - the binary relevance model of Task 1,
  - the classifier chains model of Task 1, and
  - the adapted algorithm model of Task 2.
- Write down any observations you make.

Hint: If you use the exact match ratio as one of the metrics, it is possible that it yields low values of accuracy. In this case, make sure to comment on whether it is expected or unexpected that the value is low, and why.

4 Conclusion (10 marks)

- Discuss the main challenges encountered during this assignment.
- Highlight the strengths of the different approaches used, explaining what worked well and why.
- Comment on any limitations of the methods used and of the results obtained.
- Suggest potential improvements based on your observations.

Plagiarism

All work must be your own individual work, except where acknowledged and referenced.

While you can use online tools and resources, all code, analysis and documentation must be your own work. Copying code or explanations directly from online sources, including tools like ChatGPT, is considered plagiarism.

Submission

- The submission deadline is **2025-01-17 16:00**.
- A report must be submitted as both
    1. an executable Jupyter notebook and
    2. a pdf that should be generated from the Jupyter notebook.

    Make sure that the Jupyter notebook executes cleanly. A good idea is to restart the Python kernel, run all the cells in order immediately after restarting the kernel, and then ensure that no errors have occurred.
- A filled in plagiarism declaration form must be submitted as well.