IE 506: Machine Learning: Principles and Techniques **Jan-May 2023**

Programming Challenge: Due On 26^{th} April 2023 (11 59 PM IST)

1 Instructions

This is a programming challenge which needs to be attempted individually by each course participant.

The final code and trained model should be made available in a suitable form amenable to be tested using a private test data. In particular, an interface to use the code and model for predicting on test data must be provided. More details are given in Section 2. A clear guide on how to use the code must be provided in the report. The submissions will be evaluated for a maximum of 50 marks.

Codes must be in Python language. **Other programming languages are not allowed.** The participants might refer to any resource for completing this challenge. Some pointers are given below. All resources referred to must be cited in the report and in the code.

The form to collect submissions (including the code, model, report and other related files) will be posted later.

IMPORTANT: Submissions which contain copied code and ideas will not be evaluated.

2 Programming Challenge Question

Consider the data set in file `multilabel_train_data.txt` posted in moodle (use your IITB Google SSO login to access the data).

Each line in the file `multilabel_train_data.txt` contains details about a particular sample and has the following format:

`multi_labels FeatureID:val FeatureID:val FeatureID:val ... FeatureID:val`

For example if the i-th line in the file is of the form `0,1,12 10:1 13:0.5 19:2 135:5` then it means that the i-th sample is associated with multiple labels `0,1,12` and the 10-th feature of i-th sample has a value 1, 13-th feature of i-th sample has a value 0.5, 19-th feature of i-th sample has a value 2 and 135-th feature of i-th sample has a value 5. All other features of i-th sample have value 0. Note that a sample is associated with multiple labels. This task is called multi-labeled classification.

Consider the input space as $\mathcal{X}$ and output space as $\mathcal{Y}$. The training data for multi-label classification is of the form $\{(x^i, y^i)\}_{i=1}^n$, where $x^i \in \mathbb{R}^d$ and $y^i \subseteq \mathcal{Y}$. Since a sample might be associated with multiple labels, the aim of multi-label classification is to find a map $h : \mathcal{X} \to 2^{\mathcal{Y}}$ using the training data (note that $2^{\mathcal{Y}}$ denotes the power set of $\mathcal{Y}$).

The following metrics are useful for measuring the performance of multi-label classifiers:

- Accuracy $= \frac{1}{n}\sum_{i=1}^{n} \frac{|y_i \cap \hat{y}^i|}{|y_i \cup \hat{y}^i|}$,
- F1-score $= \frac{1}{n}\sum_{i=1}^{n} \frac{|y_i \cap \hat{y}^i|}{|y_i| + |\hat{y}^i|}$,

where $y^i, \hat{y}^i$ denote respectively the actual and predicted multi-labels for sample i. The notation $y^i \cup \hat{y}^i$ denotes the labels that are present in y^i or $\hat{y}^i$. The notation $y^i \cap \hat{y}^i$ denotes the labels that are present in both y^i and $\hat{y}^i$. The notation $|y^i|$ denotes the size of multi-label set y^i.

As part of the challenge, you will be answering the following:

1. Adapt the code provided in moodle for AdaBoost to handle multi-label classification. You can consult the following resource for the extension: `https://arxiv.org/pdf/1312.6086v1.pdf` (Note that direct use of Adaboost from *scikit-learn* package is not allowed.)
2. Use the data in `multilabel_train_data.txt` to train your model.
3. The performance of your algorithm must be measured using accuracy and F1 scores.
4. In the report, include the details of the training procedure used for training AdaBoost for multi-label classification, details of hyperparameter tuning, cross-validation procedure used in training and other related details.
5. In the report, provide plots obtained for training accuracy and F1 scores vs. rounds. Also include a plot for accuracy and F1 scores computed on the data in `sample_multilabel_test_data.txt`. Any other relevant plot can be included.
6. Along with the report, the entire code should also be submitted.
7. All your files should be named according to the conventions `IE506_YOURROLLNO_CHALLENGE_CODE.ipynb`, `IE506_YOURROLLNO_CHALLENGE_REPORT.pdf`, `IE506_YOURROLLNO_CHALLENGE_MODEL.pkl`, etc. Files with other naming conventions will not be considered for evaluation.

The code needs to allow the following options for the user:

1. The code should have options to read the file `multilabel_train_data.txt` and perform the training.
2. The code should save the trained model in a suitable form (e.g. `pkl` format). The stored model should be provided in the submission.
3. The code for testing should load the model and allow user to input a test file.
4. The code should then provide the predictions for the test set in the test file.
5. Along with the predictions, the code should compute and output accuracy and F1 scores for the private test set.

The participant is encouraged to make the code very interactive. A clear guide on how to use the code must be provided in the report.

All files related to your submission for the challenge question should be placed in a Google drive in a folder named `IE506_YOURROLLNO_CHALLENGE`. The link to folder should be shared for evaluation purposes. Please make sure that the link is accessible by the TAs and Instructor. If the folder is inaccessible, the submission will not be evaluated. The form to collect the link will be posted around the submission deadline.

The submissions which give top 5 performances in terms of accuracy and F1 score on a private test set will be highlighted and all submissions with significant efforts will be awarded extra marks which would be considered in the final grading.