Pattern Recognition and Machine Learning
Lab - 5 Assignment
Bagging and Boosting

Early Bird Submission Deadline: Tuesday Batch: 27 Feb, 11:59 PM
Thursday Batch: 1 Mar, 11:59 PM
Late Submission Deadline: Tuesday Batch: 28 Feb, 2023, 12:00 Midnight (20% penalty)
Thursday Batch: 2 Mar, 2023, 23:59 (20% penalty)
Final deadline: Tuesday Batch: 1 Mar, 2023, 23:59 (addl. 20% penalty, total penalty = 40%)
Thursday Batch: 3 Mar, 2023, 23:59 (addl. 20% penalty, total penalty = 40%)

Guidelines for submission

1. Perform all tasks in a single colab file.
2. Create a report regarding the steps followed while performing the given tasks. The report should not include excessive unscaled preprocessing plots.
3. Try to modularize the code for readability wherever possible
4. Link for In-Lab Submission: Link
5. Submit the colab[.ipynb], python[.py] and report[.pdf] files here : Link
6. Plagiarism will not be tolerated

Guidelines for Report:

1. The report should be to the point. Justify the space you use!

2. Explanations for each task should be included in the report. You should know the 'why' behind whatever you do.

3. Do not paste code snippets in the report.

[In-Lab Submission Question 1 part 1]

Question 1: [60 marks] [Bagging]

1. Create a dataset with 1000 samples, using the 'make_moon's function of sklearn (choose random_state=42, noise=0.3). Perform appropriate preprocessing, train and test split of the dataset.
   Plot the generated dataset.

   Train a simple decision tree classifier from sklearn and plot the decision boundary for the same. Perform hyperparameter tuning for finding the best value of max_depth of the decision tree. [5 marks]

Train a BaggingClassifier from sklearn, on the same dataset, and plot the decision boundary obtained. [5 marks]

Train a RandomForest classifier from sklearn and plot its decision boundary. Compare the models (all 3), their decision boundaries, and their accuracy metrics. [5+ 5 marks]

Vary the number of estimators for the BaggingClassifier and RandomForestClassifier, and comment on the obtained decision boundaries and their accuracies. [10 marks]

2. Implement a Bagging algorithm from scratch. [20 marks]
   Note: The code should be well commented and the role of each function should be mentioned clearly.
   Apply the above scratch bagging algorithm with n_estimators = 10, train it on the same dataset as above. Summarize how each of the separate trees performed (both numerically and visually). How do they perform on average? [10 marks]

Question 2: [30 marks][Boosting]

Note: For installing XgBoost write the following command in one of the colab cell.
!pip install xgboost

For installing LightGBM write the following
!pip install lightgbm

Using the same dataset as in question 1,

1. Train a AdaBoost Model.
2. Train a XGBoost Model in which subsample=0.7.
3. Print the accuracy on the training set and test set.
4. Train a LightGBM model and choose different values for num_leaves.
5. Analyze the relation between max_depth and num_leaves, and check for which value the model starts overfitting.
6. Report which parameters can be used for better accuracy and also which parameter can be used for avoiding overfitting.
7. Plot the decision boundaries for all the 3 models and compare their performance.

Question 3: [10 marks]

Train a Bayes classification model on the above dataset, (using sklearn)(tune the hyperparameters accordingly)

From all the above trained models, choose any 3 models of your choice (which are giving good accuracy). Group them along with the trained Bayes Classification model, in a

VotingClassifer from sklearn. Train the VotingClassfier again. And compare its performance with the models which were individually trained. [10 marks]