DS-UA 301
Advanced Topics in Data Science
Instructor: Parijat Dube
Due: December 3, 2023

Homework 6

General Instructions

This homework must be turned in on both Gradescope and Brightspace by 11:59 pm on the due date. It must be your own work and your own work only—you must not copy anyone's work, or allow anyone to copy yours. This extends to writing code. You may consult with others, but when you write up, you must do so alone. Your homework submission must be written and submitted using Jupyter Notebook (.ipynb). **No handwritten solutions will be accepted**. You should submit:

1. On **Gradescope**: one Jupyter Notebook(.ipynb) containing your solutions for each Problem in this assignment. Each Jupyter Notebook should be named as (*netid*)_hw(*#Homework*)_p(*#Problem*).ipynb.

   For example, *bz2058_hw6_p1.ipynb*.

2. On **Brightspace**: one .zip file containing all the notebooks, datasets, and other supporting materials involved in this homework. The .zip file should be named as (*netid*)_hw(*#Homework*).zip.

   For example, *bz2058_hw6.zip*.

Please make sure your answers are clearly structured in the Jupyter Notebooks:

1. Copy the Template Notebooks provided on Brightspace and write your solutions there.
2. Label each question part clearly. Do not include written answers as code comments. The code used to obtain the answer for each question part should accompany the written answer.
3. All plots should include informative axis labels and legends. All codes should be accompanied by informative comments. All output of the code should be retained.
4. Math formulas can be typesetted in Markdown in the same way as LaTeX. A Markdown Guide is provided on Brightspace for reference.

For more homework-related policies, please refer to the syllabus.

Problem 1 - *Hyperparameter Optimization using H20* 20 points

In this question, you will compare the performances of H2O's grid search and randomized grid search. You will use the `H2ORandomForestEstimator` model, and use the *allyears2k_headers.zip* dataset used in this classification example.

1. *Grid search*
   (a) Perform grid search for identifying the best hyperparameters for the `H2ORandomForestEstimator` model with 'ntrees':[10,30,50,100] and 'max_depth': [1,2,4,6]. **(2)**
   (b) Display the grid results, sorted by accuracy in a decreasing order. **(2)**
   (c) Identify the best model and evaluate the model's performance on a test set and display the AUC score. **(2)**
2. *Randomized grid search*
   (a) Using the same model and hyperparameters grid, perform hyperparameter optimization using randomized grid search. Use a maximum of 10 models. **(2)**
   (b) Display the results sorted by accuracy in a decreasing order. **(2)**

(c) Identify the best model and evaluate the model's performance on a test set and display the auc score. **(2)**

3. *H2O AutoML*

   (a) Now using H20's AutoML find the best deep learning model for the same classification task. Use `H2OAutoML` and test a maximum of 20 models to find the best performing model. **(2)**

   (b) Display the leaderboard, identify the best performing model, and print its parameters. **(2)**

   (c) Display the AUC score of the best model for the test set. **(2)**

   (d) Identify the best *XGBoost* model among all the models tested using log loss as the criteria. **(2)**

Problem 2 - *Automated Feature Engineering* 15 points

In the lab, we looked at AutoFeat, a Python library that automatically does feature engineering and selection for you.

1. Explain the importance of interpretability when training machine learning models. Why is model explainability necessary? **(2)**
2. Perform feature selection for the Diabetes regression dataset using `FeatureSelector()`. How many features are discarded? **(4)**
3. Perform a train-test split on your dataset. Select a regression model from skLearn and fit it to the training dataset. What is the R^2 score on the training and test set? **(4)**
4. Keeping the train and test dataset the same, run 3 feature engineering steps using `AutoFeatRegressor()`. What is the R^2 score on the training and test set now? Mention any five new features generated by the output of `AutoFeatRegressor()`. **(5)**

Problem 3 - *Ray Tune for Hyperparameter Optimization* 15 points

In this problem, we will compare the performance of Grid Search, Bayesian Search, and Hyperband for hyperparameter optimization for a deep learning problem using Ray Tune. We will use the MNIST dataset along with the Lenet model. You can use the same resources per trial and metric as those in the Lab.

The hyperparameters to tune are:

- Number of filters in the first Conv2d layer: 64 to 256
- Learning Rate: 0.001 to 0.1
- Batch Size: 64, 128, 256
- Dropout: probability between 0 and 1

1. Perform Grid Search, Bayesian Search, and Hyperband for the given hyperparameter configurations. For Grid Search, you can either sample uniformly between the given ranges or specify a list of values in the given range (for e.g., filters = [64,128,256], lr=[0.001,0.01,0.1], etc). **(8)**
2. For each of the search techniques in part 1, display the time taken to perform the analysis and display the hyperparameters for the best model. **(4)**
3. What are your observations regarding the time taken and performance of the best model? **(3)**