# Final Course Project 2021

Submit by: 01/08/2021

**Introduction**:

The goal of the final exercise is to evaluate the performance of Deep Learning ensemble methods that were published in the professional literature that weren't covered during the course.

**Stage 1: Selecting the algorithm for evaluation**

1. Select an algorithm for evaluation. Choose either
    a. an algorithm from this list
    OR
    b. You may instead choose a deep learning ensemble method which doesn't appear on the list. In this case the article must have either been published in the last 20 years in a top (Tier 1) conference or journal or published in a pre-print repository (like Arxiv) in the last 3 years. If you choose this option please send an email to your instructor to make sure your choice is acceptable.
2. The algorithm selection will be "First Come First Serve" - If a group selects an algorithm the other groups may not select the same algorithm.
3. Enter the algorithm you selected at this link. Please make sure that the algorithm was not selected already.
4. In the project report include:
    a. A description of the algorithm you chose.
    b. The advantages of the algorithm.
    c. The disadvantages of the algorithm.

**Stage 2: Suggesting an improvement**

Suggest an improvement to the algorithm you chose in Stage 1. Give a detailed explanation of your proposed improvement and why you think it might improve the algorithms performance.

**Stage 3: Select a well-known algorithm for comparison**

Select a well-known algorithm as a baseline to compare with the algorithm you are evaluating.

**Stage 4: Evaluating the algorithms you selected in stages 1, 2 and 3**

Versions:

If the algorithm you selected has multiple versions select the version that has the best performance in the original article.

Data:

Select 20 datasets from the one of the following links depending on the type of task you selected.

A. If you selected a regression task select 20 datasets from this list.
B. If you selected a classification task select 20 datasets from this list.

You may add up to 10 additional datasets. If you do, provide the details in your report.

Evaluation Protocol

A. External 10-fold Cross Validation - to separate between the Train and Test and reporting the performance metrics.
B. Internal (within Train) 3-fold Cross Validation – for hyperparameter optimization

Hyperparameter Optimization

Do hyperparameter optimization for the algorithms that you selected for stages 1,2 and 3. For hyperparameter optimization use Bayesian Optimization or Random Search with at least 50 trials and over a search space of at least 2 hyperparameters (justify your choice). The hyperparameter performance should be measured with an appropriate metric of your choosing using the 3-fold cross-validation.

Performance metrics for evaluation

A. Accuracy – Under the assumption that the classification is the Class with the highest probability.
B. TPR
C. FPR
D. Precision
E. AUC – Area Under the ROC Curve
F. Area under the Precision-Recall
G. Training time
H. Inference time for 1000 instances

Reporting of the results should be done in an excel sheet for each problem (Dataset), Algorithm (from stage 1, 2 and 3) and each fold of the cross validation in the following format:

| Dataset Name | Algorithm Name | Cross Validation [1-10] | Hyper-Parameters Values | Accuracy | TPR | FPR | Precision | AUC | PR-Curve | Training Time | Inference Time |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | |

**Stage 5: Statistical significance testing of the results**

Choose one of the performance metrics (AUC for example) and use the Friedman test as was presented in the lecture to determine whether the differences are statistically significant. If the results are statistically significant (the null-hypothesis is rejected), do a Post-Hoc test to test the differences between the algorithms. Report and analyze your results and conclusions,

**Stage 6: Reporting your conclusions**

Present the three algorithms you are evaluating (stages 1-3). Include the pseudo code, results and conclusions.

**General Instructions:**

**A. Programming language:**
   a. Python (preferred)
   b. R
**B. Program versions:**

It is recommended to use the latest version of the well-known algorithms.

**C. What affects the grade**

- Completeness – all of stages were executed
- Correctness
- Clarity – Clear writing of the report
- Originality
- Efficiency of the implementation
- Content – the complexity of the project
- Detailed evaluation of the algorithm (selecting appropriate metrics, optimizing algorithm parameters based on multiple parameters, selecting appropriate baselines, varied datasets)
- Coming to interesting conclusions that are supported by the experimental results
- Code documentation

**D. Plagiarism**

All the work you submit (Report, Figure, Table, Equation, Slide, Software code etc.) must be your personal work unless otherwise explicitly indicated by a reference to the source. You are allowed to copy excerpts for the article, but you must note this explicitly. An attempt to submit other work as your own (plagiarism) will result in a grade of 0 in the course. If this project is also being used for any other purpose (like for another course) you must explicitly state this and emphasize what additions were specific to this course.

**E. Project submission**

- Source code, detailed instructions on how to run the code, and detailed algorithm hyperparameters should be uploaded to a GitHub repository for this project. Submit the link you project on GitHub
- A zip file that that includes
    - A link to the GitHub repository
    - Detailed results of your experiments
    - A report according to the instructions above