# Qualities in Intelligent Systems

Graded Project

In this assignment you are expected to make use of explainable AI concepts and batch serving concepts. You should be familiar with explainability and interpretability methods used to understand ML models decisions.

You can make use of the boiler plate code from the TA session files to begin with.

Programming Language: Python

**Part-I (25 Points)**

**Introduction:**

In this assignment, we will aim to model bike rental prediction using Machine Learning and explore the explainability of the model.

**Dataset:**

You can find the dataset for this task below

http://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset

**Task**:

explainable_AI_starter.ipynb

1. Understand how the h2o package helps in explainability with various plots on relation between attributes and the defect prediction.
2. Build an explainable ML model using the H2o package on the bike rental dataset and display various plots and write your findings. (25 points)
    a. Consider cnt as the target column.
    b. Consider the Day data and hourly data and build two models separately.

c. Try at least two models (AutoML and any other model like gradient boosting) for the Day dataset.
d. Discuss how various plots try to explain the amount of bikes rented correspond to the various environmental conditions.

## Part-II (25 points)

**Introduction:**

In this assignment, we will build a docker container to perform batch serving of a ML model. This is a continuation of the TA session demo.

You need the following files:

- Dockerfile
- Train.py
- Inference.py

This is the EEG brainwave data that has been processed using statistical extraction. There are totally 1300 rows and 162 columns in the train dataset. The feature Letter is used as the target column. It has 26 classes which is a representation of the 26 alphabets.

**Task**:

train.py

1. Add one more model apart from the one used in TA Session (any machine learning model of your choice) in the train.py file and save it. (5 points)
2. Modify the inference.py file to display the output of the above model. (5 points)
3. Build the docker image of the final application and run it and submit the screenshot of the output. (15 points)
    a. Build the Docker File
    b. Run the docker container
    c. Save the screenshot of the output and submit it.

The final submission of this assessment should be made on Olympus.

For Part-I: The file should be submitted in .ipynb and .HTML (both) formats.

For Part-II: Zip all the files and submit it on Olympus. Also please share the screen-shots of the final results.

Regards,

Program Office