[IE801] SPECIAL TOPICS IN INDUSTRIAL ENGINEERING II - MANUFACTURING & AI

HW #1

Due: 11:59 PM, March 18

Introduction

This assignment focuses on implementing boosting algorithms for defect classification in the battery industry. The provided dataset (`battery.csv`) contains inspection results across several battery manufacturing processes. Each sample represents a battery unit, and the objective is to classify whether a battery is defective (minority class) or not (majority class). Students will implement three boosting algorithms from scratch using primitive functions: AdaBoost, AdaCost, AdaAUC (AUC-based AdaBoost).

Dataset Description

The dataset consists of 9,099 samples and 29 input features, with one class column representing the target label. The class column has two values: 0 (non-defective battery), 1: (defective battery). Some features are continuous, while others are categorical or count-based. Nevertheless, it is acceptable to consider all features as continuous.

Homework Tasks

- Students must implement the following algorithms using primitive functions (i.e., without using existing boosting libraries): AdaBoost, AdaCost, and AdaAUC.
- Use decision trees as weak learners with limited depth (to maintain their “weak” nature).
- Perform stratified random partitioning to create training and test sets. Train the models on the training set and evaluate them on the test set. Report the following metrics for both training and test results: TPR, FPR, and AUC
- For the AdaCost algorithm, experiment with different cost settings to find the best AUC result.
- Submit your code and results by uploading the files to the KLMS system.

Notes

- While students may use AI tools for guidance, they must fully understand their implementations.
- Observing the effect of different boosting approaches on classification performance is important.