HD Task

Step-1

This task is designed to assess the High Distinction level expectations.

Step-2

Your tutor will then review your submission and will give you feedback. If your submission is incomplete the tutor will ask you to include missing parts. Tutor can also ask follow-up questions, either to clarify something that you have submitted or to assess your

Feedback and submission deadlines

Feedback deadline: Not applicable

Submission deadline: Before creating and submitting portfolio.

Required documents

1. Submit a report (pdf format) in Ontrack (https://ontrack.deakin.edu.au)
2. Complete the problem credit task and submit your code file (.ipynb) separately in the OnTrack (https://ontrack.deakin.edu.au).

Background

Heart disease remains the leading global cause of death. The attached article presents a stacking-based ensemble learning framework to improve early prediction of heart attacks using a curated clinical dataset. Your objective is threefold: (1) reproduce the published findings, (2) design and evaluate your own machine learning solution, and (3) prepare a 3-minute video presentation demonstrating your understanding and process.

Datasets Description

Download the heart disease dataset from Kaggle:
https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset

Evidence of Learning

Part 1 – Reproducing Published Results

Read the attached article and reproduce the results for the following classifiers: Logistic Regression, Decision Tree, Random Forest, Extreme Gradient Boosting (XGBoost), K-Nearest Neighbors (KNN), Naive Bayes, and the Stacking Ensemble proposed by the authors.

Ensure that your implementation uses the same set of features, train/test splitting approach, preprocessing, and evaluation metrics (Accuracy, Precision, Recall, F1 Score, AUC) as reported in the article. Match classifier parameter settings where available. If any model details are missing, make and justify your own assumptions. You must present your results using tables and/or graphs and clearly compare your outcomes with those presented in the original paper (e.g., Tables 11–13). You are expected to explain any discrepancies and variations in your results with evidence and reasoning.

Important notes:

- If you find any issue in reproducing results due to incomplete description of models in the provided article, then make your own assumption and explain the reason. If your justification is correct, your solution will be accepted.
- Variation in results due to randomness of data splitting will also be considered valid if clearly explained.
- The level of performance is based on the number of ML methods correctly reproduced. Missing results indicate that the submission is not at HD level.
- Your submitted Python code must generate the results you report. Otherwise, you will receive no credit for that part.

Part 2 – Design and Develop Your Own ML Solution

Design and develop your own machine learning solution for the heart disease prediction task. The proposed solution must be substantially different from the approaches presented in the attached article. However, this does not mean you are required to choose a new ML algorithm. You can develop a novel solution by changing the feature selection approach, using different preprocessing techniques, exploring alternative model combinations, or restructuring the training pipeline. The goal is to propose a system that is meaningfully distinct from the authors' stacking ensemble method.

Your technical report must include:

- A clear motivation for the proposed solution
- A description of how your method differs from the article
- A well-structured experimental protocol
- Evaluation using Accuracy, Precision, Recall, F1 Score, and AUC

- Results presented in tables and/or graphs
- Comparative discussion with the results reported in the article
- Discussion of results with respect to existing literature
- References using IEEE numbered citation format

Part 3 – Video Presentation (Mandatory)

Prepare a 3-minute video presentation that:
Explains the problem and your understanding of the task
Walks through your methodology and key findings
Reflects on outcomes, challenges, and what you learned
shows code execution or model demonstration
The video must include screen-sharing and voice narration. Upload it to YouTube (unlisted) and submit the link with your report and code.

Deliverables

- Python code files (.ipynb)
- Technical report with Video Presentation link