DEEP LEARNING ASSIGNMENTS

Important Note:

In addition to working code, examiner will review the following when grading/evaluating the submission:

- Efforts to Improve the evaluation metrics (such as by making adjustments in hyperparameters or feature engineering)
 (keep both before and after in the notebook)
- Explanation in comments on the design choices and the metrics you paid attention to when evaluating the model. What you chose to optimize?
- Comments on various tradeoffs for the choices and selections you made (such as training time vs optimizing evaluation results)

Assignment 1 - Predict Diabetes Onset with an ANN

Dataset: Pima Indians Diabetes (UCI)

Task: Build and optimise an ANN that predicts the Outcome (diagnosed diabetes).

Steps:

💪 Problem framing	Define baseline metric (e.g., ROC-AUC ≥ 0.80).
👲 Data handling	Load the CSV from the URL, inspect class balance, split 60-20-20 (train/val/test).
* Pre-processing	Scale numeric features, handle any zeros as missing for medical realism.
Modelling	1) Baseline logistic regression 2) Build a Keras ANN (≥ 2 hidden layers).
Optimisation	Experiment with units, activation functions, dropout, learning-rate schedules, early-stopping.
ii Evaluation	Accuracy, precision/recall/F1, ROC-AUC, confusion-matrix heat-map.
Reporting	Compare ANN vs. baseline, justify architecture choices, discuss error patterns.

Starter notebook: Provided in the assignment folder

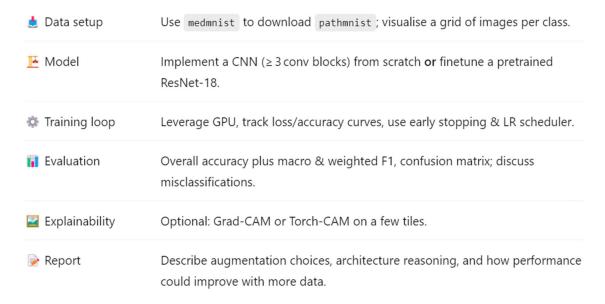
Submit:

Link to completed notebook with the outputs in it

Assignment 2 - Classify Histopathology Images with a CNN (Optional)

Dataset: PathMNIST (9-class tissue-type images, part of MedMNIST). Small (4-MB) yet realistic.

Task: Train a CNN in PyTorch to label pathology tiles.



Starter notebook:

Provided with the assignment.

Submit:

Link to completed notebook with the outputs

Assignment 3 - Sentiment Analysis with an LSTM

Dataset: IMDB Movie Reviews (large movie-review corpus, 25 000 training + 25 000 test examples, balanced positive/negative)

Task: Build and optimise an LSTM-based sequence model that predicts sentiment from raw text.

Steps:

Data ingestion	Load the dataset with tensorflow_datasets; create 80-20 train/validation split from the official training set.
13 Text prep	Fit a Tokenizer (20 k vocab), convert to integer sequences, and pad/trim to a fixed length (e.g., 300 tokens).
 ★ Baseline	Implement a TF-IDF + Logistic Regression classifier and report validation accuracy.
™ Modelling	Build a Bidirectional LSTM with an Embedding layer (≥ 128 dims) and at least one stacked LSTM layer; add dropout/regularisation.
Optimisation	Tune embedding size, LSTM units, learning rate, and early-stopping patience to beat the baseline.
ii Evaluation	Accuracy, precision/recall/F1, and a confusion matrix on the held-out test set.
Reporting	Compare baseline vs. LSTM, discuss misclassified examples, and propose next improvements.

Starter notebook:

Provided with the assignment

Submit:

Link to completed notebook with the outputs