Project

Objective:
This project is designed to develop students' practical skills in artificial neural network (ANN) design, implementation, and evaluation, with a focus on solving real-world problem solving. Students are required to demonstrate their understanding and application of ANNs by working with real-world datasets and implementing one of the following neural network models:

- Multilayer Perceptrons (MLP)
- Convolutional Neural Networks (CNN)

Students will work in teams of up to three (3) students to develop, implement, and analyze a neural network model. The project will emphasize network design, training, evaluation, and interpretation of results.

Project Proposal Submission (Due: 3/26/25)
To begin, submit a 1-2 page proposal outlining the following details:

1) Project Title: Provide a clear and concise title for the project
2) Team Members: List all participating team members
3) Objectives: Define the project goals. What problem are you solving? What insights do you seek?
4) Task Description and Methodology:
   - Provide an overview of the problem and its significance.
   - Describe the dataset you plan to use.
   - Explain the ANN techniques you will employ (e.g., MLP for classification, CNNs for object recognition).
   - Outline your network architecture, training process, and evaluation strategy.
5) Roles and Responsibilities: Clearly define the contributions of each team member to ensure balanced workload distribution.

Project Guidelines:

1. **Topic Selection (5 pts)**
   - Choose a real-world problem that can be effectively addressed using artificial neural networks.
2. **Data Collection and Preprocessing (10 pts)**
   - Obtain a suitable dataset (e.g., UCI Machine Learning Repository, Kaggle, OpenML, etc.)
   - Perform data cleaning and preprocessing (handle missing values, normalize/scale features, remove duplicates, etc.).
   - Conduct exploratory data analysis (EDA) to understand feature distributions and relationships.
   - Justify the preprocessing steps taken.
3. **Neural Network Implementation (40 pts)**
   - Design and implement a neural network suited to the problem.
   - Use Python with TensorFlow or PyTorch for implementation.
   - Train the network using backpropagation and optimization techniques.
   - If applicable, compare multiple architectures and justify your choice.

- Explain model design, activation functions, and the learning process.

4. Model Evaluation (20 pts)

- Evaluate model performance using appropriate metrics (e.g., MSE, R^2, accuracy, precision, recall, F1-score).
- Visualize results (e.g., regression curves, learning curves, decision boundaries, feature maps (for CNNs), or activation visualizations).
- Discuss training challenges, overfitting/underfitting, and possible improvements.

5. Report and Presentation (25 pts)

Final report (7-10 pages) including:

- Introduction and problem statement
- Description of the datasets and preprocessing steps
- Network architecture, implementation and training process
- Evaluation, insights, and conclusion

Presentation (10-15 minutes):

- Summarize project findings, methodology, and key results.

Deliverables

- Python script containing ANN implementation and analysis
- Final report (Word or PDF format)
- Presentation file (PowerPoint or PDF slides)
- Do NOT submit a zipped file