AtomCamp Cohort 11 (Deep Learning Module)
Assignment 2: Optimizing Deep Learning Models for Real-World Data

1 DATASET: UCI FOREST COVERTYPE

(*This dataset has already been introduced and a baseline model demonstrated during Session 4.*)

2 OBJECTIVE

This assignment is designed to help you improve upon a baseline neural network model using practical techniques in architecture design, regularization, optimization, and evaluation. You will also compare the performance of your neural network with ensemble-based machine learning models and reflect on their relative strengths. The goal is to equip you with skills and intuition expected of a deep learning engineer working with structured data.

Note: These tasks have already been demonstrated in class using the FashionMNIST dataset. By reviewing Sessions 3 and 4, you should be in a position to reproduce and adapt these techniques to achieve a **test accuracy of around 94%** on this dataset with proper tuning.

3 TASKS

3.1 NEURAL NETWORK ARCHITECTURE

- Modify the baseline MLP architecture by exploring deeper or wider configurations.
- Experiment with different activation functions (e.g., ReLU, LeakyReLU, SELU).
- Apply Batch Normalization where appropriate and explain your design choices.

3.2 REGULARIZATION TECHNIQUES

- Apply Dropout to prevent overfitting and experiment with different dropout rates.
- Add L2 regularization to Dense layers and observe its impact on learning dynamics.

3.3 OPTIMIZER AND LEARNING RATE STRATEGY

- Try multiple optimizers (e.g., Adam, RMSprop, SGD with momentum).
- Use learning rate scheduling strategies like ReduceLROnPlateau.
- Reflect on how different configurations affect training stability and generalization.

3.4 TRAINING MANAGEMENT

- Integrate EarlyStopping to halt training when validation performance stops improving.
- Log training and validation performance per epoch.
- Visualize training curves (accuracy and loss).

3.5 MODEL EVALUATION

Note: Due to class imbalance in the Forest CoverType dataset, **accuracy alone is not a sufficient evaluation metric.**

You are required to report the following:

- **Accuracy**
- **Precision**, **Recall**, and **F1-score** (both macro and weighted)
- **Confusion Matrix** for detailed error analysis
- Visualized **learning curves** for training and validation

3.6 COMPARISON WITH ENSEMBLE METHODS

- Train and evaluate either a RandomForestClassifier or XGBoostClassifier using the same dataset.
- Report the same metrics as above for fair comparison.
- Reflect briefly (3–5 lines): *Why do tree-based ensemble methods often outperform MLPs on structured/tabular datasets?* Base your reasoning on your results and what you've learned in class.

4 DELIVERABLES

Submit a Jupyter Notebook containing:

- Complete and readable code with outputs
- Visualizations and model summaries
- Metric-based evaluation of each model
- Commentary explaining your tuning and design decisions
- Final comparative analysis between MLP and ensemble method