

# Progetto di Laboratorio di Ottimizzazione, Intelligenza Artificiale e Machine Learning

Anno Scolastico 2024/25

Matteo Aloè – Elia Strazzella

---

## Classificazione Automatica delle Malattie delle Piante con MobileNetV2 e PyTorch

### Introduzione

Il progetto è stato realizzato utilizzando PyTorch e sfrutta tecniche di transfer learning, valutazione automatica e visualizzazione con TensorBoard.

In questo progetto viene affrontato il problema della classificazione automatica delle malattie delle piante utilizzando immagini fogliari provenienti dal dataset PlantVillage. L'obiettivo è costruire un modello di rete neurale in grado di identificare correttamente la malattia (o la salute) della pianta a partire dalla sola immagine.

### Dataset e Preprocessing





Il codice ha gestito automaticamente la suddivisione in train/val/test, la rinomina delle classi e la riduzione delle classi maggioritarie tramite undersampling.

Il dataset PlantVillage è stato preprocessato in diverse fasi:

- Riduzione classi troppo rappresentate tramite undersampling.
- Rinomina classi per pulizia.
- Normalizzazione delle immagini secondo i parametri di MobileNetV2 pretrained su ImageNet.
- Suddivisione train/val/test (70/15/15).
- Data augmentation con RandomResizedCrop, rotazioni, flip orizzontale e ColorJitter.

### Esempi Visivi dal Dataset

Di seguito sono riportati alcuni esempi di immagini fogliari estratte dal dataset PlantVillage. Ogni immagine rappresenta una diversa classe del problema di classificazione, includendo sia foglie sane che affette da varie patologie. Queste immagini costituiscono il punto di partenza per l'addestramento del modello MobileNetV2.

<p>Potato - Healthy</p>	
<p>Pepper Bell Healthy</p>	
<p>Tomato Mosaic Virus</p>	
<p>Tomato Leaf Mold</p>	

## Architettura

È stato implementato un modello basato su MobileNetV2 pre-addestrato su ImageNet, utilizzato come estrattore di feature. Gli ultimi strati sono stati adattati tramite fine-tuning per permettere al modello di specializzarsi meglio sul nostro dataset. La testa del modello consiste in un classificatore composto da un Dropout seguito da un layer fully connected (Linear) adattato al numero di classi, che assegna l'immagine a una delle 15 classi.

## Training

Durante il training è stato possibile riprendere l'addestramento da checkpoint salvati grazie alla funzionalità di resume automatica. Inoltre, è stato integrato il logging dei valori di accuratezza e loss tramite TensorBoard.

L'addestramento è stato effettuato utilizzando la tecnica di transfer learning con un modello pre-addestrato MobileNetV2. Il modello è stato addestrato fino a 10 epoche con early stopping, ottimizzando i parametri adattandoli al dataset specifico.

Sono state utilizzate le callback EarlyStopping con pazienza personalizzabile e salvataggio dei checkpoint (model\_best.pth, model\_last.pth) per migliorare la generalizzazione e ottimizzare il processo di addestramento.

## Valutazione

Il modello è stato valutato attraverso:

- Accuracy e loss su validation set, grafici di andamento, matrice di confusione e classification report.
- Matrice di confusione.
- Classification report (precision, recall, f1-score).
- Analisi visiva dei risultati con grafici di training/validation.

## Conclusione

Il progetto ha dimostrato come l'uso del transfer learning con MobileNetV2 possa portare ottimi risultati anche con dataset medicali agricoli a 15 classi. I risultati ottenuti sono soddisfacenti, con un'accuratezza valida sul validation set. Ulteriori sviluppi possono includere la sperimentazione con altre architetture o l'uso di tecniche di bilanciamento più avanzate.

**Requirements:**

- *pytorch*
- *cuda*
- *numpy*
- *matplotlib*
- *seaborn*
- *scikit-learn*
- *opencv-python*
- *Pillow*
- *tensorboard*