Practical Deep Learning with Python:
Neural Networks, CNNs & Transfer Learning

Course Outline

Instructor: **Nazmus Sayadat Rumman**
Trainee Officer, Credit Risk Management, Bank Asia
B.Sc. Applied Mathematics, University of Rajshahi
M.Sc. Applied Statistics & Data Science, Jahangirnagar University

Content	
Chapter 1	**Foundations of Neural Networks** **1.1 The Building Blocks** ▸ Neuron & perceptron model — geometric intuition ▸ Multi-layer perceptron (MLP): layers, weights, biases ▸ Activation functions — ReLU, sigmoid, tanh, softmax **1.2 How a Network "Thinks"** ▸ Forward propagation — from input to prediction ▸ Hand-tracing a small network step by step **1.3 Kaggle Environment Setup** ▸ Notebooks, GPU/TPU runtime ▸ Dataset workflow on Kaggle
Chapter 2	**Training and Saving Neural Networks** **2.1 How Networks Learn** ▸ Loss functions ▸ Backpropagation & chain rule — intuition, not full derivation ▸ Gradient descent variants: SGD, Momentum, RMSprop, Adam **2.2 Keeping Training Stable** ▸ Vanishing/exploding gradients ▸ Weight initialization strategies ▸ Batch Normalization **2.3 Generalization** ▸ Regularization: L1/L2, Dropout **2.4 Saving Your Work** ▸ Saving a trained model (Keras model.save()/PyTorch state_dict) ▸ Reloading a saved model for later reuse
Chapter 3	**Structuring a Deep Learning Project (ML Strategy)** **3.1 Setting Up the Problem Correctly**

	▸ Train/dev/test split strategy ▸ Bias vs. variance diagnosis **3.2 Debugging a Model** ▸ Error analysis workflow ▸ Data mismatch pitfalls **3.3 Hyperparameter Tuning** ▸ Learning rate, batch size, epochs ▸ Callbacks: early stopping, checkpoints
Chapter 4	**Image Processing & Convolutional Neural Networks** **4.1 Working with Images** ▸ Digital image basics: pixels, channels, color spaces ▸ Preprocessing: resizing, normalization ▸ Data augmentation **4.2 How CNNs See** ▸ Convolution & pooling operations ▸ CNN architecture design **4.3 Build & Train** ▸ Building and training a CNN from scratch on a Kaggle image dataset
Chapter 5	**Transfer Learning with Pretrained Models** **5.1 Standing on Giants' Shoulders** ▸ Landmark architectures overview: LeNet, VGG, ResNet ▸ Loading a pretrained model **5.2 Adapting a Pretrained Model** ▸ Freezing/unfreezing layers ▸ Fine-tuning on a new dataset **5.3 Building a Reusable Pipeline** ▸ Saving/loading the full adapted model for later use
Chapter 6	**Deployment- FastAPI & Streamlit** **6.1 Serving a Model** ▸ FastAPI basics ▸ Loading a saved model into an inference endpoint **6.2 Building an Interface** ▸ Streamlit basics ▸ Connecting the Streamlit frontend to the FastAPI backend **6.3 Testing End-to-End** ▸ Sending requests, checking predictions, handling errors
Chapter 7	**Live Project Demo** **7.1 End-to-End Pipeline**

STATX Let's Think Deep

	▸ Select a Kaggle dataset ▸ Train/fine-tune a CNN ▸ Evaluate performance ▸ Save the model ▸ Deploy via FastAPI + Streamlit
Assignment Project	

STATX
Let's Think Deep