Module 1: Getting Started

1. Introduction to Artificial Intelligence
   - History of AI
   - Applications of AI
   - AI in Computer Vision
   - AI Terminology
   - Why Deep Learning so popular

2. NumPy Refresher
   - Introduction to NumPy
   - Why do we need a special Library for Maths and DL
   - NumPy Basic Operations
   - Mathematical Functions
   - Reshape & Combine Array
   - Element-wise Operations
   - Linear Algebra
   - Array Statistics

3. Introduction PyTorch
   - Why PyTorch
   - Introduction to PyTorch
   - PyTorch Basics

4. What is inside an ML Algorithm
• Machine Learning Pipeline
• Solving ML Problems
• Gradient Descent
• Gradient Descent for Optimization

**Assignment1**: Implement ReLU, Softmax and Neuron using PyTorch

**Assignment2**: Implement Gradient Descent for two variables

**Module 2 : Neural Networks**

1. Feature Vectors 1-D to N-D
   • Feature Vectors and Normalization

2. Neural Network Basics
   • What is Neural Network
   • Loss Functions for Regression
   • Loss Functions for Classification
   • Types of Activation Functions
   • How does the network learn
   • Demystifying Neural Networks

3. Binary Classification using Perceptrons
   • Binary Classification using a Perceptron

4. PyTorch NN Module
   • Introduction to pyTorch NN Module
   • PyTorch NN Module
   • MLP using Functional API
   • MLP using Sequential API

5. Image Classification using Multilayer Perceptron
Assignment 3: Implement MSE and MAE

Module 3: Convolutional Neural Network

1. Convolution Operation
   - What is Convolution Operation
   - CNN Building Blocks
   - Layers in CNN

2. How to implement LeNet using PyTorch
   - How to implement LeNet
   - Implementing LeNet using PyTorch
   - LeNet with BatchNorm
   - Effects of Batch Normalization

3. Evaluation of Classification Performance
   - Performance Metrics for Classification
   - How to Implement Classification Metrics

4. Introduction to Torchvision
   - TorchVision Overview
   - Datasets
   - What are the different Transforms used to Train a Network
   - Different Models in TorchVision
   - Utils: Utility Functions in TorchVision
   - IO Operations in TorchVision
   - Ops module in TorchVision

5. Important CNN architectures
• Different CNN Architecture
• Pre-trained Models in Torchvision
• Pre-trained Classification Models in TorchVision

**Assignment4:** Implement CNN for Image Classification on CIFAR10 Dataset

---

**Module 4 : Deep Neural Networks**

1. Optimization

   • What are Optimizers
   • Learning Rate Decay Methods
   • LR Scheduler

2. Training Deep Neural Networks

   • Step1: Data understanding
   • Step2: Data Preparation
   • Step3: Check Training Pipeline
   • Step4: Train the Model
   • Step5: Improve the Model
   • Check Training Pipeline

3. How to add Robustness to a model

   • Bias variance Tradeoff
   • How to prevent Overfitting
   • Training with Regularization

4. Data Loader with Image Folder

   • How to load Custom Datasets in PyTorch

5. GPU access on Azure
• How to get Microsoft Azure Pass
• Redeem Azure Pass
• Create an Instance on Azure
• Run Jupyter Notebooks on Azure
• Login to your instance using SSH
• How to stop your instance

Assignment5: Implement Adam Optimizer

Project1: Implement an Image Classifier from scratch

Module 5 : Best Practices in Deep Learning

1. Troubleshooting Training with Tensorboard
   • Tensorboard Overview
   • Tensorboard Dashboard
   • Logging using Tensorboard
   • Sharing Tensorboard Logs

2. Leveraging Pre-Trained models
   • CNN Architectures(Recap)
   • Fine-Tuning and Transfer Learning
   • Fine-Tuning using ResNet

3. How to structure your project for scale
   • Introduction to py_modules Package
   • Motivation of Trainer Pipeline
   • Hands-on Trainer Pipeline

4. PyTorch Lightning
   • Introduction to PyTorch Lightning
Module 6: Object Detection

1. Object Detection overview
   - Introduction to Object Detection
2. Evaluation Metrics
   - Evaluation Metrics for Object detection
   - Compute Evaluation Metrics
3. Traditional Algorithms in Object Detection
   - Different Traditional Algorithms
   - Implement Non-Maximum Suppression
4. Two stage Object Detectors
   - Introduction to Two Stage Object Detectors
   - Faster RCNN using TorchVision
   - Understanding Faster RCNN
   - Faster RCNN Fine-tuning
   - Faster RCNN Fine-Tuning Training

Module 7: Single Stage Object Detectors

1. YOLO
   - Introduction to YOLO
2. Single Stage Multibox Detector(SSD)
   - Introduction to SSD
   - SSD with PyTorch Hub
3. RetinaNet
   - Introduction to RetinaNet
   - RetinaNet with Detectron2
4. How to create Custom Single Stage Detector

Project 2: Kaggle Competition on Image Classification
Module 8: Segmentation

1. Semantic Segmentation Architecture
   - Semantic Segmentation Architectures
   - Dilated Convolution
   - Transposed Convolution
   - Fully Convolution Network (FCN)
   - U-Net
   - SegNet
   - Deeolab
2. Evaluation Metrics for Semantic Segmentation
   - Dice Coefficient Metrics
3. LinkNet Architecture
   - Introduction to LinkNet Architecture
4. Soft-Dice Loss
   - Introduction to Soft-Dice Loss
5. FCN and DeepLab using TorchVision
   - FCN and DeepLabV3 using Torchvision
6. U-Net for MRI Abnormality Segmentation
7. Train your Model from scratch
8. Instance Segmentation
   - Instance Segmentation using Mask RCNN

Assignment7: LinkNet Architecture with VGG16

Project4: Kaggle competition on Semantic Segmentation
Module 9 : Pose Estimation

1. Dense Pose
   - Introduction to DensePose
   - DensePose Inference
   - DensePose Training
2. Create your own Gym Trainer
   - Squat Checker

Project5: Create an App of your choice

Module 10 : Azure Deployment and Cognitive Services

1. How to your App on Azure Cloud Instance
   - Virtual Machine Creation for Deployment
   - Naive Deployment
   - Robust Deployment
   - Deployment using Azure App Service
2. Introduction to Azure Cognitive Services
   - Azure Cognitive Services

Project6: Deploy your App on Azure using Github repository

Module 11 : LibTorch

1. Introduction to TorchScript
2. Introduction to LibTorch
   - LibTorch Installation
   - Introduction to LibTorch
• From PyTorch to LibTorch
• Training with Custom Dataset

3. Introduction to ONNX