Suggested Teaching Guidelines for

Deep Neural Networks

PG-DAI February 2025

Duration: 40 hours Theory + 50 hours Lab

Objective: Deep Neural Network

Prerequisites: Knowledge of fundamentals of programming and basic mathematics & statistical knowledge.

Evaluation method: Theory exam– 40% weightage
Lab exam – 40% weightage
Internal exam – 20% weightage

<u>List of Books / Other training material</u>

Courseware:

Deep Learning using Python, by S Lovelyn Rose, L Ashok Kumar , D Karthika Renuka, Wiley India

Reference Book:

1. Deep Learning with Python by Francis Chollet
2. Deep Learning, Ian Goodfellow, Yoshua Bengio, Aaron Courville
3. Neural Networks and Learning Machines, Simon Haykin
4. Pattern Recognition and Machine Learning, Christopher M. Bishop
5. Hands-On Machine Learning with Scikit-Learn and TensorFlow
6. TensorFlow Deep Learning Cookbook
7. Reinforcement Learning with TensorFlow: A Beginner's Guide to Designing Self-learningSystems with TensorFlow and OpenAI Gym SayonDutta
8. Hands-On Reinforcement Learning with Python: Master Reinforcement and Deep Reinforcement Learning Using OpenAI Gym and TensorFlow Sudharsan Ravichandiran
9. Deep Reinforcement Learning Hands-On: Apply Modern RL Methods, with Deep Qnetworks, Value Iteration, Policy Gradients, TRPO, AlphaGo Zero and More MaximLapan

Session 1:

Introduction to Deep Neural Network

- Neural Network and its applications ,Single layer neural Network
- Activation Functions: Sigmoid, Hyperbolic Tangent, ReLu ,Overview of Backpropagation of errors

Assignment –Lab:

- Implement the different activation functions on the Dataset in the Jupyter notebook.

Session 2:

- Introduction to Tensorflow
- Introduction to Pytorch
- Comparison of Tensorflow and Pytorch

Assignment –Lab:

- Explore Tensorflow ,Pytorch and their libraries.

Session 3 & 4:

- Introduction Deep Learning and Neural Networks
- Practical Application of Neural Networks
- "Hello World" of Neural Network (Logistic Regression)
- Cost Function
- Activation
- Gradient Descent for Logistic Regression
- Back propagation

Assignment –Lab:

- Implement the main steps of a Shallow Neural Network
  - Understand the dataset
  - Implement your first Forward and Backward propagation
  - Implement activation function, gradient descent
  - Build Neural Network Model
  - Test and optimize the model
  - Make Predictions

  (to be done in Jupyter notebook. Need Jupyter-notebook and libraries installed)

Session 5: (2T + 4L)

- Sigmoid Model
- Sigmoid Loss function
- Introduction to Learning Algorithm
- Deriving the Gradient Descent Update Rule
- Sigmoid Evaluation

Assignment –Lab:

- Plotting Sigmoid 2D
- Plotting Sigmoid 3D
- Contour Plot
- Plotting Loss
- Standardization
- Test/Train split

Session 6: (2T + 4L)

- Shallow Neural Networks
- Hidden Units and Hidden Layers
- Activation function- Tan-h and ReLu
- Forward and Backward propagation with a hidden layer
- Deep Learning notations and Neural Network Representations

Assignment –Lab:

- Implement a two-class neural network with a hidden layer
- Implement forward and backward propagation
- Compute the cross-entropy loss

Session 7 & 8:

- Parameters vs Hyper Parameters
- Regularization
- L1-L2 normalization
- Frobenius norm

Assignment –Lab:

- Implement L1-L2 Regularization
- Improve performance of the learning algorithms

Session 9:

- Dropouts, Early Stopping
- Data Augmentation

Assignment –Lab:

- Implement Dropouts, Early Stopping and Data Augmentation.

Session 10:

- Vanishing Gradient and Exploding Gradient problems
- Gradient Checking

Assignment –Lab:

- Implement Gradient problems

Session 11: (2T + 4L)

- Batch Normalization and other methods for data normalization

Assignment –Lab:

- Programming exercise for Normalization

Session 12:

- Optimization Algorithms, ADAM
- Mini batch gradient descent

Assignment –Lab:

- Implement ADAM in python and Tensor flow

Session 13:

- RMSProp, Momentum and other gradient descent algorithms

Assignment –Lab:

- Implement gradient descent algorithms

Session 14 & 15:

- Tensor flow data structures and Library

Assignment –Lab:

- Fashion MNIST and digits MNIST exercises using Tensor flow.

Session 16: (2T + 4L)

- Sequential Modelling, Introduction
- Basic building blocks, applications
- Introduction to RNNs

Assignment –Lab:

- Implement RNN in Jupyter Notebook .

Session 17 & 18:

CNN - Convolutional Neural Networks

- Introduction, basic building blocks,
- Convolutional Concept
- Inception Network
- Transfer Learning
- Data Augmentation
- Padding, strides
- Pooling layers
- Fully connected
- Applications and Use Cases

Assignment –Lab:

- Using tensor flow and python implement CNN for ASL (American Sign Language)

Session 19: (2T + 4L)

- GANs - introductions, basic building blocks
- Implementing GAN

Assignment –Lab:

- Implement GAN using python and tensor flow.

Session 20:

Tuning Deep Learning Models

- Deciding Number of Layers
- Deciding Number of Neurons
- CNN tuning for better performance
- RNN/LSTM for Time Series Prediction

Trends in Deep Learning

- Echo State Networks / Reservoir Computing
- Auto Encoders
- Convolutional Auto Encoder
- Extreme Learning

Deep Learning Case Studies

- Whale Identification (5k Classes)
- Iris Detection
- Devnagari Digit Detection
- Flood prediction in Dam
- Heart Disease

Assignment –Lab:

- Implement CNN using python and tensor flow
- Implement RNN/LSTM using python and tensor flow

Self-Study:

Reinforcement Learning: Markov Decision Processes, Dynamic Programming, Monte Carlo,Temporal Difference Learning, Approximation Methods