Deep Learning Programming Assignment

Ex. No.	**Date** (Tentative)	**Topic**
1	20/01/25	Feed Forward & Back-Propagation Learning Algorithm
2	27/01/25	ANN for MNIST digit Classification
3	05/02/25	CNN for MNIST digit Classification
4	10/02/25	ResNet-152 for Binary Classification of Skin Lesions
5	17/02/25	Autoencoder for dimensionality reduction Autoencoder for anomaly detection
6	03/03/25	VAE for fashionMNIST (Synthetic Data Generation)
7	10/03/25	Deep fake generation using GAN
8	17/03/25	RNN+LSTM
9	24/03/25	UNet model for segmentation / YOLO for object detection
10	31/03/25	Transformer model and attention mechanism
11	07/04/25	Vision Transformer
12	12/04/25	Self supervised Learning - Contrastive Learning
13	21/04/25	**Project submission**

Lab - 01 **Feed Forward & Back-Propagation Learning Algorithm**

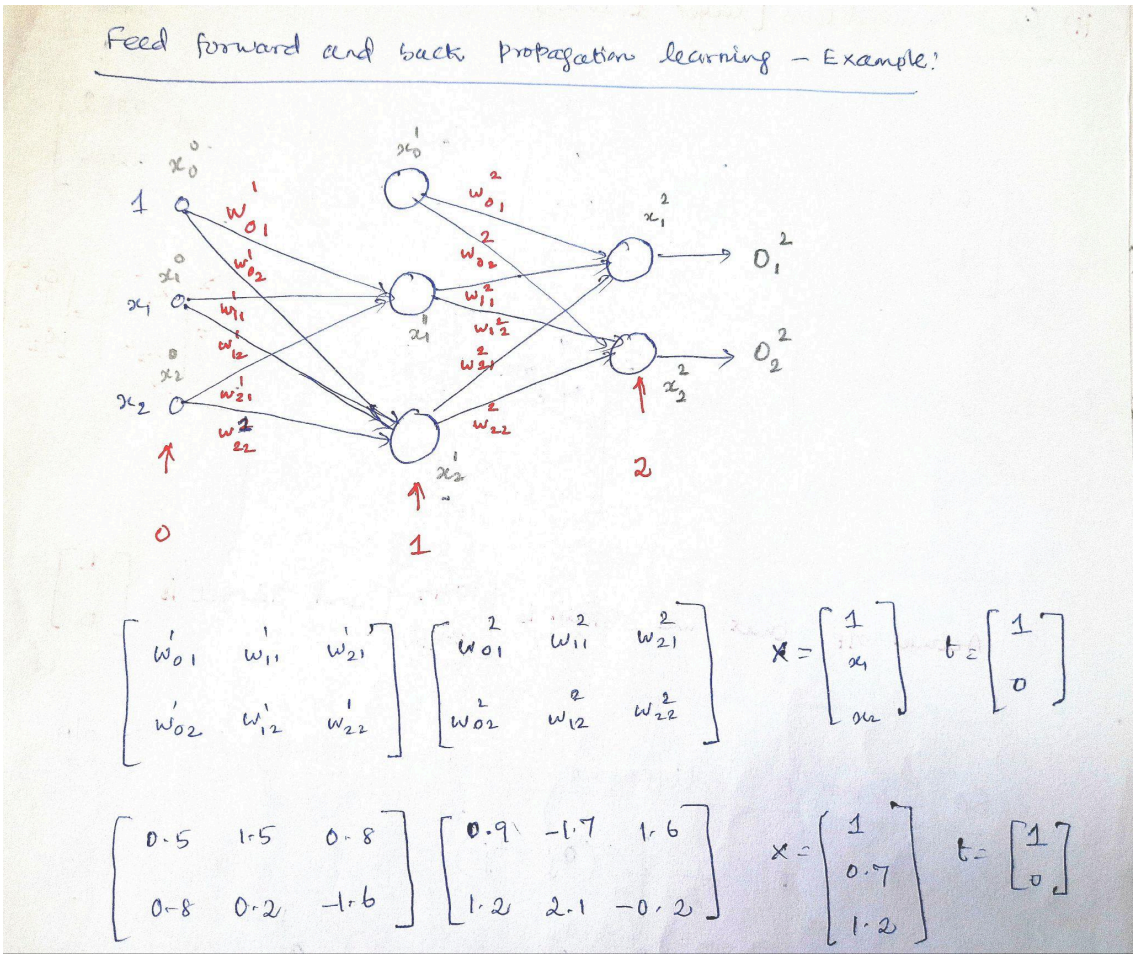
Date: 20-01-2025

NOTE - Do not use inbuilt functions for perceptron.

1. Implement the simple neural network algorithm from scratch in Python.
    - Initialize the weights with [0 0 0] and a learning rate of 0.0001.
    - For each iteration, calculate the output of the simple neural network for each input in the training set.
    - Use MSE to computer the error for all samples
    - Update the weights using the gradient descent procedure.
    - Repeat the above steps until the simple neural network converges or a maximum number of iterations is reached.
    - Test the trained simple neural network on a separate test set, explain how you came up with the test set.
    - Use the step function as an activation function in the output layer and sigmoid function for other layers.

Use the IRIS Dataset for the above, considering all four features: sepal length, sepal width, petal length, and petal width, but only two classes - **Setosa, and Versicolor**. Drop the feature vectors of the other class.
Please find the dataset here - Iris Dataset

2. Implement the feedforward and backpropagation learning algorithm for multi layer perceptrons in Python for the question provided in the attached image.
    a. Use the weights and biases as given.
    b. Implement the forward pass.
    c. Compute the loss between the predicted output and the actual output using an appropriate loss function (MSE).
    d. Compute the gradients of the loss function with respect to the weights and biases using the chain rule.
    e. Update the weights and biases.
    f. Iterate over multiple times (epochs), performing forward propagation, loss calculation, backpropagation, and parameter updates in each iteration till convergence (the actual output is the same as the target output).



Lab - 02 **ANN for MNIST digit Classification**

Date- 27-01-2025

Q1. Classify MNIST digits using Fully Connected Neural network.
Dataset : download from internet source

- Plot few samples from dataset
- Train the network
- Test on the test dataset
- Calculate test accuracy on test set

Lab - 03 **CNN for MNIST digit Classification**

Date- 05-02-2025

Q1. Classify MNIST digits using CNN.
Dataset : download from keras/pytorch/standard source.

- Plot few samples from dataset
- Train the network
- Test on the test dataset
- Calculate test accuracy on test set

Report:

Make a comparative report by analysing Fully Connected ANN and CNN based NN for the task of MNIST digit classification. For comparison use indicators like no. of model parameters, accuracy and others. Include architecture diagrams.

Submission Files:

1. CS22Bxxxx_lab03.ipynb
2. CS22Bxxxx_lab03_report.pdf (max 2 page)

Lab - 04 **Binary Classification of Skin Lesions**

Date: 12 Feb, 2025

1. In this lab assignment, implement the **ResNet-152(or ResNet 101)** model, a deep convolutional neural network, for classifying skin lesions into benign or malignant (binary classification).
    a. Use the dataset provided, write **a custom dataset class** to load the dataset properly.
    b. **Do not use prebuilt ResNet architecture** that comes with tensorflow/pytorch frameworks, instead **create architecture using basic layers and functions from the framework** like Conv2d, Linear, Relu, Pooling and others required.
    c. Print model parameters and architecture to verify it matches the standard one.
    d. **Experiment with different values of hyperparameters, describe the observation at each step of experimentation.** (*important* - must be documented in the submission file)
    e. After training the model, evaluate the model's performance on the test dataset provided.
    f. Calculate classification metrics such as **accuracy, precision, recall, and F1-score.**
    g. Finally, summarize your observations/learnings after completing this assignment task.

Try appropriate batch size to train your model using Colab, Kaggle platform.

Dataset

- **explore more about dataset at https://challenge.isic-archive.com/data/#2020**
- dataset link(to be used by students) https://drive.google.com/file/d/1Ntd1TaSsax7VZCsfI7LbAgfPjvkmNAEU/view?usp=sharing
- create a separate dirs for **melanoma(malignant)** and **non-melanoma(benign)** from the train set based on labels.
- Use only relevant columns for this classification task.

Lab - 05 **AutoEncoders**

Date: 22 Feb, 2025

1. MNIST dataset compression and reconstruction using PCA and AutoEncoder.
    a. Train a PCA model to compress the digits to 4 dimensional latent vectors.
    b. Train an AutoEncoder model to compress the digits to 4 dimensional latent vectors.
    c. Visualize the latent vectors using TSNE(https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html) for both the cases.
    d. Show reconstruction for 20 latent samples using both models.
    e. What is the compression ratio achieved?

2. Train an autoencoder model for anomaly classification
    a. Use the dataset assigned for your group.
    b. Train model only on normal set.
    c. Reconstruct on the test set and show the reconstruction error.
    d. What is the latent dimension?
    e. Plot TSNE for the test set.
    f. What is the threshold value of reconstruction error to classify as normal or anomalous ?

Group1:

Dataset: Melanoma/Non Melanoma dataset
Normal set: Non Melanoma set
Link: Use the split created in the last assignment from the train set.
Students: [CS22B1003 : CS22B2031]

Group2:

Dataset: Normal/Cataract disease dataset
Normal set: Normal
Link: normal_cataract.zip
Students: [CS22B2032 : all others]

Lab - 06 **Variational AutoEncoders for Synthetic Data Generation**

Date: 3 March, 2025

3. Train a VAE model for generating new data similar to training set.
    a. Train the model on fashionMNIST dataset.
    b. Generate 30 new samples using the learned model and plot them.

Dataset: **fashionMNIST**

https://github.com/zalandoresearch/fashion-mnist?tab=readme-ov-file#get-the-data

Lab - 07 **Deep Fake Generation: GAN model on CelebA dataset**

Date: 10-03-2025

1. Train a GAN model on celebrity faces dataset.
    a. Use basic layers from the frameworks to write the discriminator and generator.
    b. Understand the flow of data from input to output while training and testing.
    c. Visualize samples from the dataset after loading the dataset.
    d. After each epoch, plot a few samples to track the training progress.
    e. Generate 50 samples from the trained model.

Dataset: CelebA

Use only images from the dataset to train the GAN model, resize the image to 64x64 for training to minimize computational requirements.

- **Main source of the dataset:** https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html
- **PyTorch:** https://pytorch.org/vision/stable/generated/torchvision.datasets.CelebA.html
- **Tensorflow:** https://www.tensorflow.org/datasets/catalog/celeb_a

Lab - 08 **Sentiment Classification using RNN, LSTM**

Date: 17-03-2025

1. Train a RNN based sentiment analysis model for classification of movie reviews.
    a. Explore and learn about the different preprocessing steps in the Natural Language Processing(NLP) domain.
    b. Apply suitable preprocessing steps for this sentiment analysis assignment.
    c. Build and train a RNN model using basic layers from the framework.
    d. Test model on the test set using suitable evaluation metrics.

2. Train a LSTM based model for the same sentiment analysis problem.
    a. Build and train a LSTM model using basic layers from the framework.
    b. Test model on the test set using suitable evaluation metrics.

Compare between the two approaches and highlight the improvements.

Dataset: Stanford Sentiment Treebank 2

Original dataset link: https://huggingface.co/datasets/stanfordnlp/sst2

Dataset Zip Link: https://drive.google.com/file/d/1TytoIgt7KI9Ep9bo8bs_X0HSSnBJX0oi/

Data fields in dataset: *Read the parquet file using pandas*

- `idx`: Monotonically increasing index ID.
- `sentence`: Complete sentence expressing an opinion about a film.
- `label`: Sentiment of the opinion, either "negative" (0) or positive (1).

Split the provided train dataset of 67349 rows into 5000 for testing and rest for training. Use the separately provided validation dataset (872 rows) file for validation.

Lab - 09 Segmentation using U-Net

Date: 24-03-2025

Brain Tumor Segmentation using U-Net on LGG MRI Dataset

1. Train a Simple U-Net Model for Brain Tumor Segmentation

   a. Explore and learn about different preprocessing steps in medical imaging, specifically for MRI scans.

   b. Apply suitable preprocessing steps such as normalization, resizing, and data augmentation for this segmentation task.

   c. Build and train a U-Net model using basic layers from the framework for brain tumor segmentation.

   d. Test the model on the test set using suitable evaluation metrics such as Dice coefficient and IoU.

Dataset: LGG MRI Dataset

Preprocess the dataset by normalizing and resizing the images. Split the dataset into training, validation, and testing sets (e.g., 80% training, 10% validation, 10% testing).

Dataset Link : https://www.kaggle.com/datasets/mateuszbuda/lgg-mri-segmentation

Dataset Zip Link:
https://drive.google.com/drive/folders/1F7pobjXSJk99EUtph3_gdq_b0ZgaG30Y?usp=sharing

Lab - 10 News Summarization using Transformers

Date: 07-04-2025

1. Fine tune a transformer based model to summarize descriptive news.
    a. Explore and learn about different variants of transformer based models and tasks for which they can be used.
    b. List out the models, max token inputs, no. of parameters.
    c. Select a suitable variant of **BART** model, which can be finetuned on free compute platforms like colab/kaggle.
    d. Use the dataset for summarizing the news articles, fine tune the model after loading pretrained a model from hugging face.
    e. Explore and learn about different metrics to evaluate the performance of the model for text summarization tasks.
    f. Use suitable metrics to evaluate the fine tuned model performance on the test set.

Dataset:

- **Indian Language Summarization Dataset**
- **https://huggingface.co/datasets/ILSUM/ILSUM-1.0**
- For this assignment use the "**English**" subset.
- Filter subset of dataset such that not much information is lost (because input token size is fixed) and no of training samples > 1000 for fine tuning.
- Use the same filtering criteria for test and validation set.

To get familiar with hugging face library, you can refer to this video tutorial:

Stanford CS224N NLP with Deep Learning | 2023 | Hugging Face Tutorial, E...

Lab - 11 Image Classification using ViT

Date: 07-04-2025

1. Classify chest Xray images using ViT in two classes.
    a. Build a vision transformer model using the basic layers from the framework.
    b. Resize images to 224x224.
    c. Apply suitable patching mechanism to convert images into patches to input to the ViT.
    d. Use ‘Finding Labels’ column to classify images in 2 classes: ‘No finding’ vs ‘With findings’ (every label combined other than no finding)
    e. Split the dataset in 70:10:20 ratio for training, validation and testing and make sure that the class distribution is maintained in all the splits.
    f. Train the model using the processed dataset obtained after applying above steps.
    g. Use all suitable classification metrics to test the model performance on the test set.

Dataset:
Chest Xray dataset: xray_dataset.zip