**Assignment Write-up Structure**

Assignment No:

Aim:

Objectives:

Theory:

Algorithm (if applicable)

Dataset used and its attributes ( if applicable)

Implementation/Program/Code

Results/Visualization Graphs

Analysis of results ( for accuracy etc.)

Conclusion

**A. Machine Learning**

|  | Introduction to Keras and Tensorflow (Optional – Pytorch). Configure and use google colab and kaggle GPU  **Objectives**:  1. To configure anaconda and google colab, kaggle environment  2. To Explore TF/Keras/Pytorch libraries  3. To learn to use GPU/TPU  4. To learn and understand Git  **Theory to be included**  1. Keras/TF/Pytorch configuration steps  2. Google colab configuration steps and commands  3. kaggle configuration steps and commands  4. Github configuration and settings  5. Git commnads |
| --- | --- |
|  | Develop multi class classifier using deep multilayer perceptron (Keras/tensorflow/pytorch) for MNIST hand recognition dataset and CIFAR10. Fine the parameters for better accuracy.  Analyze the model accuracy and generate classification report. Plot accuracy and loss graph.   Develop application with GUI to upload input to the system.   Test the model.  **Objectives:**  1. Learn Deep Neural Network modeling  2. Learn to develop and deploy models  **Theory to be included**  1. Description of basic data preprocessing steps for images.( Data cleaning, Standardization, Normalization, data splitting)  2. Optimization algorithms  3. Activation functions  4. Loss functions |
|  | Develop classification model for cat-dogs dataset using CNN model. Analyze the model accuracy and generate classification report.   Develop an application and test the user given inputs.   Analyze the result with and without regularization/dropout  **Objectives:**  1. To learn about classification  2. To learn CNN  3. To demonstrate and analyse the results  **Theory to be included**  1. What is CNN (Description with block diagram and required equations)?  2. Batch normalization  3. Regularization  4. Padding, Strides |
|  | Develop face recognition system using CNN. Create a dataset of minimum 20 students from your class. Check and validate the accuracy of the model.   Apply dimensionality reduction on input image and plot the change in accuracy of system.  **Objectives**  1. To learn Data set creation  2. To learn data normalization  **Theory to be included:**  1. Dataset creation steps  2. Image Augmentation  3.Libraries for image augmentation  4. Fit\_generator, validate\_generator, predict\_generator |
|  | Write a program to demonstrate the change in accuracy/loss/convergence time with change in optimizers like stochastic gradient descent, adam, adagrad, RMSprop and Nadam for any suitable application  **Objectives:**  1. To learn optimization algorithms  2. To learn and understand hyperparameters  **Theory to be included:**  1. SGD, Adam, RMSprop, Nadam  2. Dataset exploration (used for demonstration) |
| 6. | Apply transfer learning with pre-trained VGG16 model on assignment 3 and analyze the result.  **Objectives:**  1. To learn pre-trained models  2. To learn transfer learning  **Theory to be included:**  1. Transfer Learning details, approaches  2. Pre-trained models and explanation (ResNet 50, MobileNet, Inception)  3. VGG 16 detailing  4. Significance of TL |
| 7 | Develop RNN model for Cryptocurrency pricing prediction or text sentiment analysis  **Objectives:**  1. To learn RNN  2. To learn and implement LSTM  **Theory to be included:**  1. RNN  2. LSTM  3. BPTT  4. GRU  5. Bidirectional RNN |
| 8 | Develop an autoencoder to encode and decode the image. Analyze the results.  a) Develop AE for MNIST dataset  b) Use output of AE as input to CNN  **Objectives**  1. To learn AE  2. To implement AE  **Theory to be included**  1. Autoencoders  2. Types of AE  3. Applications |

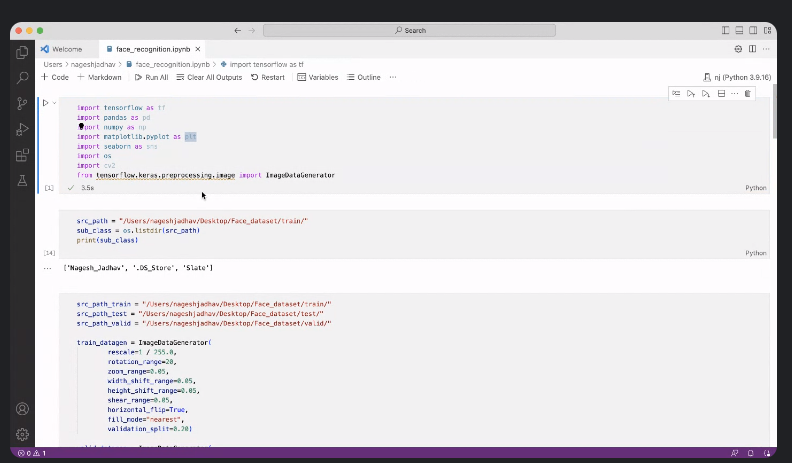
Assignment 4

05 09 2023

1. Image generator

A folder in C drive called Dataset with three folders

* Train Dataset
  + Folders of the different features Labelled
* Validation dataset
  + Same folders
  + State
    - Images must be different in this compared to train
* Test Dataset
  + Predict Folder
  + State



Passing one image at a time for testing