# Machine Learning 3

Class 02

https://tinyurl.com/nhahmeaw

**Rifat Jahan Azad**

Consultant Technical Lead of Data science and ML (Brain Station 23)
Chief technology officer(ktinformatik)
contact(whatsapp): +8801775568572
linkedin: https://www.linkedin.com/in/rifat-jahan-azad/

# Overview

Session 26 : DNN:

- MLP and Backpropagation
- Regression MLP
- Implementing MLP with Keras
- Fine tuning NN hyperparameters
- Activation function
- Batch normalization
- Monte Carlo droupout
- Tensor Flow's API
- Dataset with Keras
- Deep Computer Vision using CNN

- MLP Feedforward only
- Backpropagation
- Logistic vs Regressive MLP
- MLP in Tensorflow
- MLP in Keras
- MLP in Torch
- Fine Tuning Hyperparameters
- Activation Functions, Skip connection, Pooling and Strides
- Normalization vs Batch Normalization
- Dropout vs MC Dropout
- TF API
- Datasets
- CNN
- RNN
- Realtime Training Visualization
- Working with Different Types of data
  - Text
  - Audio
  - Image
- Different Architectures
  - Single Neuron
  - CNN
  - RNN
  - Transformers

# Live Weight Update Demo

CNN MNIST: https://github.com/apssouza22/neuralnet-browser?tab=readme-ov-file

Live Training: https://playground.tensorflow.org

# Builtin Datasets

https://keras.io/api/datasets/

https://www.tensorflow.org/datasets/catalog/overview

Grandchallenge

Pdb protein related

pytorch

# Without Neuron

## And Gate

```python
def and_gate(input1, input2):
    """
    Simulates a logical AND gate.
    :param input1: Boolean value (True or False) for the first input.
    :param input2: Boolean value (True or False) for the second input.
    :return: Result of AND operation (True if both inputs are True, otherwise False).
    """
    return input1 and input2

# Test the AND gate
print(and_gate(True, True))   # Output: True
print(and_gate(True, False)) # Output: False
print(and_gate(False, True)) # Output: False
print(and_gate(False, False)) # Output: False
```

```python
def and_gate_binary(input1, input2):
    """
    Simulates a logical AND gate with binary inputs.
    :param input1: Integer value (0 or 1) for the first input.
    :param input2: Integer value (0 or 1) for the second input.
    :return: 1 if both inputs are 1, otherwise 0.
    """
    return input1 & input2

# Test the binary AND gate
print(and_gate_binary(1, 1))  # Output: 1
print(and_gate_binary(1, 0))  # Output: 0
print(and_gate_binary(0, 1))  # Output: 0
print(and_gate_binary(0, 0))  # Output: 0
```

# Colab

https://colab.research.google.com/drive/1oWvyQSVsGWdG_irgOhvrkWXRqZGy65EF?usp=sharing