

# Project 01 – Math

Arthur J. Redfern

[arthur.redfern@utdallas.edu](mailto:arthur.redfern@utdallas.edu)

Feb 22, 2021

---

## 1 Logistics

- Assigned Feb 22, 2021 and due Mar 12, 2021
- This is an individual project, no help from others is allowed
- The use of any and all online resources is allowed but mildly discouraged; everything you need is available on the class GitHub page and it's likely that you'll find this project easier if you start there
- As mentioned in the instructions and repeated here for emphasis, the use of xNN packages is not allowed (i.e., no PyTorch, no TensorFlow, ...)

## 2 Goals

- The math section of the course provides the theory for understanding
  - Using layers to map input data to features to classes
  - Computing an error by generating a class pdf and comparing it with the true pdf
  - Back propagating the sensitivity of the error with respect to feature maps
  - Computing the sensitivity of the error with respect to parameters
  - Updating the parameters to minimize the error
- In this project you will put theory to practice and write Python (with NumPy) code to mimic a simple / lite / reduced functionality PyTorch library and demonstrate its use in the design and training of a simple neural network for MNIST image classification
- Optionally, you will add software enhancements that mimic those commonly found in professional xNN libraries and demonstrate their use in the design and training of the same simple neural network for MNIST image classification
- Optionally, you will add additional layers to support CNNs and demonstrate their use in the design and training of a simple CNN for MNIST image classification

## 3 Project

Follow the instructions in the 3 comment blocks at the top of the provided skeleton code.

## 4 What To Turn In Via eLearning

Follow the instructions in the 3 comment blocks at the top of the provided skeleton code for the requested Python files to turn in. Note: no zip files, no Jupyter / iPython notebooks, ... I should be able to cut and paste the Python code you submit in the .py file(s) into Google Colab and reproduce the results you provided at the top of your Python file(s).