Missouri University of Science & Technology Department of Computer Science
Spring 2022 **CS 6406: Machine Learning for Computer Vision (Sec: 101/102)**

Homework 1: Learning

Instructor: *Sid Nadendla* **Due:** *February 27, 2022*

Goals and Directions:

- The main goal of this assignment is to implement perceptrons and neural networks from scratch, and train them on any given dataset
- Comprehend the impact of hyperparameters and learn to tune them effectively.
- You are **not** allowed to use neural network libraries like PyTorch, Tensorflow and Keras.
- You are also **not** allowed to add, move, or remove any files, or even modify their names.

Problem 1 Neural Network Components *5 points*

- **Linear Features (1 points):** Implement a linear function in hw1/mlcvlab/nn/linear.py
- **Activation Functions (2 points):** Implement four activation functions, namely step, ReLU, Sigmoid, Softmax and Tanh function in hw1/mlcvlab/nn/activations.py.
- **Loss Functions (2 points):** Implement two loss functions, namely mean squared error (MSE) and binary cross entropy in hw1/mlcvlab/nn/losses.py.

Problem 2 Optimization Algorithms *5 points*

- **SGD (2 points):** Implement SGD in hw1/mlcvlab/optim/sgd.py
- **AdaM (3 points):** Implement AdaM in hw1/mlcvlab/optim/adam.py

Problem 3 Regularization *5 points*

- **1D Batch Normalization (3 points):** Implement one-dimensional Batch Normalization in hw1/mlcvlab/nn/batchnorm.py
- **Dropout (2 points):** Implement Dropout in hw1/mlcvlab/nn/dropout.py

Problem 4 Models *5 points*

Using library functions defined in hw1/mlcvlab/nn/*, do the following:

- **1-layer Neural Network (1 points):** Implement a one-layer NN in hw1/mlcvlab/models/nn1.py
- **2-layer Neural Network (1 points):** Implement a two-layer NN in hw1/mlcvlab/models/nn2.py
- **3-layer Neural Network (1 points):** Implement a three-layer NN in hw1/mlcvlab/models/nn3.py
- **4-layer Neural Network (1 points):** Implement a four-layer NN in in hw1/mlcvlab/models/nn4.py
- **4-layer Neural Network with Batch Normalization and Dropout (1 points):** Implement a four-layer NN with batch normalization and dropout features in hw1/mlcvlab/models/nn4_reg.py

Problem 5 Classification on MNIST Data *5 points*

- **Training on MNIST (2 points):** Train all the models in hw1/mlcvlab/models/* on MNIST data in the Jupyter noteboo, labeled as hw1/HW1_5.ipynb
- **K-fold Cross Validation (3 points):** In the same Jupyter notebook (i.e. hw1/HW1_5.ipynb), validate each of the five models using K-fold cross validation algorithm, with $K = 5$ folds.