Aim:

In this project, you will develop and test deep networks to classify natural. You will mainly be working with the MNIST dataset to gain intuition. The smaller MNIST dataset allows you to develop code and test it quicker. Once you have gained enough intuition with MNIST, you should extend it to the **CIFAR-10 dataset**. The dataset and its description can be found at https://www.cs.toronto.edu/~kriz/cifar.html

We expect one can start from simple baseline network architectures and training strategy, and gradually add in more advanced or even state-of-the-art techniques, including high-end architectures, data augmentation and network regularization, to improve the classification accuracy. https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html is a proper starting point.

Platform and Deep Learning Library:

You are encouraged to use

- Google Colab: which offers Free GPU resources with popular deep learning libraries (Pytorch, Tensorflow) installed.
- Pytorch: Easy to use.

1. Baseline Network on MNIST dataset (25 points)

The goal of this component is to extend the code that is provided to you on ICON to implement a different network structure.

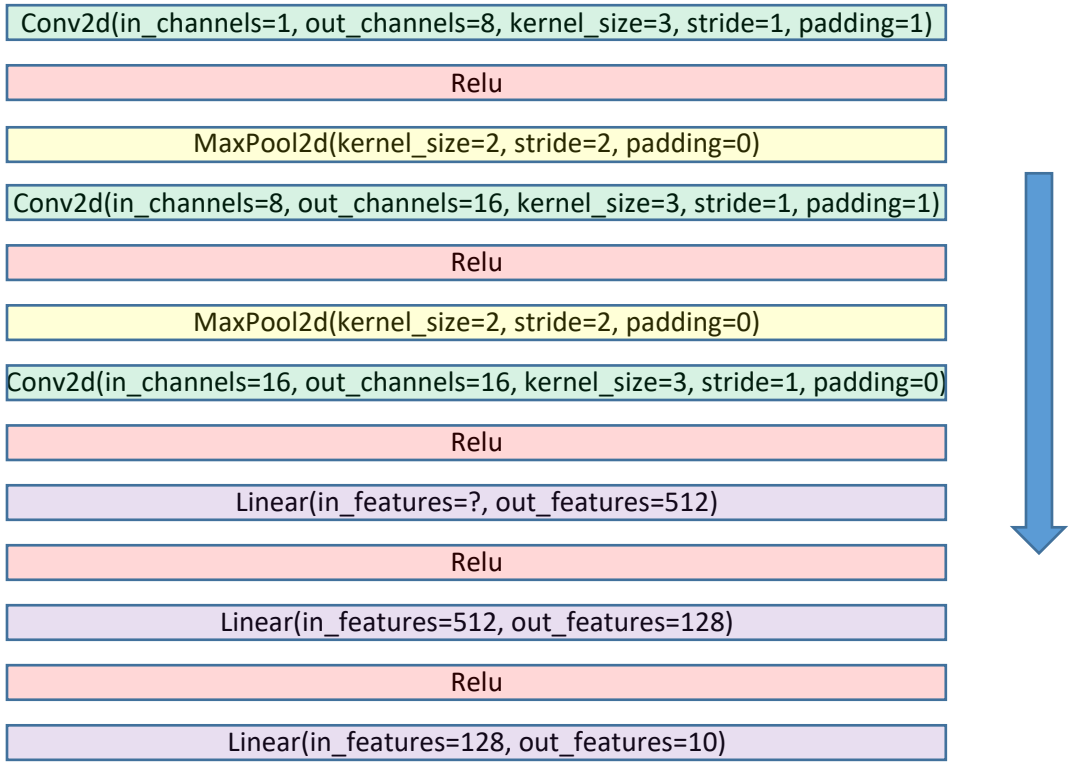

Fig. 1

1. Create the network as shown in Figure 1.
2. Find a proper learning rate and plot the training loss vs epoch.
3. Compare the test performance with the baseline network provided to you on the MNIST dataset.

See https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html for an example

2. Model Exploration (50 points):

The goal of this section is to understand the impact of the following hyperparameters and algorithmic choices on the performance of the system.

1. Learning rate (LR) and Optimizer: Adam or SGD (10 points)
    a. Reading Material:
        i. https://medium.com/octavian-ai/which-optimizer-and-learning-rate-should-i-use-for-deep-learning-5acb418f9b2
        ii. https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c
        iii. https://shaoanlu.wordpress.com/2017/05/29/sgd-all-which-one-is-the-best-optimizer-dogs-vs-cats-toy-experiment/
    b. Find proper LR (lr_best) for Adam and SGD, plot training loss vs training epoch number, and compare the convergence speed of the two optimizers and their respective test classification accuracies. Plot training loss vs training epoch number when setting learning rate to be 10 x lr_best and 0.1 x lr_best.
    c. Describe the lessons you learn from the experiments. Specifically compare the training convergence for the three learning rates (0.1 x lr_best, lr_best and 10 x lr_best).
2. Activation functions
    a. Reading Material:
        i. https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/
    b. Train two networks with Sigmoid and Relu as respective activation functions
    c. Test and compare the training convergence speeds and classification accuracies on the test dataset. Give your observation.
3. Early stopping strategy (10 points)
    a. Reading Material:
        i. https://machinelearningmastery.com/early-stopping-to-avoid-overtraining-neural-network-models/
        ii. https://towardsdatascience.com/preventing-deep-neural-network-from-overfitting-953458db800a
    b. Develop your early stopping strategy. Plot the training and validation loss vs training epoch number. Is there over-fitting problem?
    c. Report the classification accuracies with or without early stopping (generally overfitted model).
4. Data augmentation (10 points)
    a. Reading Material:

i. https://nanonets.com/blog/data-augmentation-how-to-use-deep-learning-when-you-have-limited-data-part-2/
ii. https://www.aiworkbox.com/lessons/augment-the-cifar10-dataset-using-the-randomhorizontalflip-and-randomcrop-transforms

b. Augment the training data and train the network
c. Report the classification accuracy and compare it to that without using augmentation

5. Network depth vs network width (10 points)
   a. Design two networks with different depths (e.g. 3 layers vs 5 layer), but similar total number of parameters.
   b. Report the classification accuracy and give your observation. Specifically is it true the deeper the better?

3. Extension to CIFAR-10 dataset (25 points):

The goal of this component is to extend the above model to CIFAR 10 dataset and report the testing performance. Note that the CIFAR10 dataset requires more training time. It may be difficult to vary the parameters and test their impact. You may look at

https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html for examples.

4. Extra credit (20 points): Optimize the model to improve performance

Based on the intuition gained in Section 2, adapt the model (depth, number of layers), parameters (learning rate), data augmentation, and early stopping to improve the performance of the model. Report the test accuracy on the CIFAR-10 dataset.

Plagiarism is not acceptable. It is also plagiarism to train your networks on the test dataset. If recognized, minus 10 points will be given. You have to submit your complete codes that generate you best results reported.

Grading: The scores for the extra-credit will be based on classification accuracy during testing.

Top 5 students: 20 points

Next 5 students: 15 points

Next 5 students 10 points

Next 5 students 5 points

Rubric:

a. The project will be scored mainly based on how much designing space you have explored. Achieving better test accuracy is a bonus.
b. Plagiarism is not acceptable.
   a. Please don't copy from your colleagues

b. Please don’t’ copy from the web
c. 0 points for project and report to Dean if copying is detected !!