

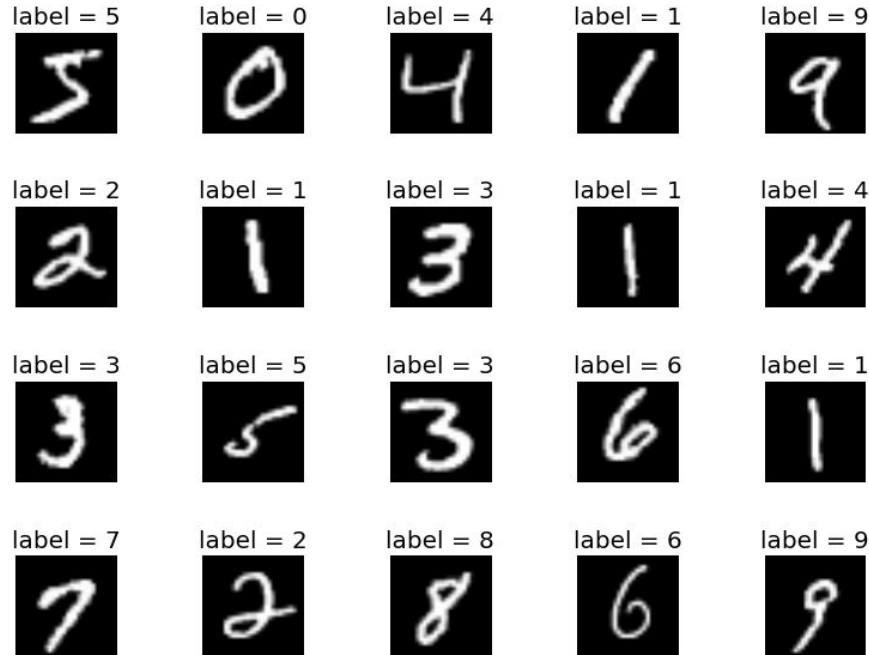
OpenACC

Lab3



Deep Neural Network(DNN)

- MNIST hand written digits classification Inference



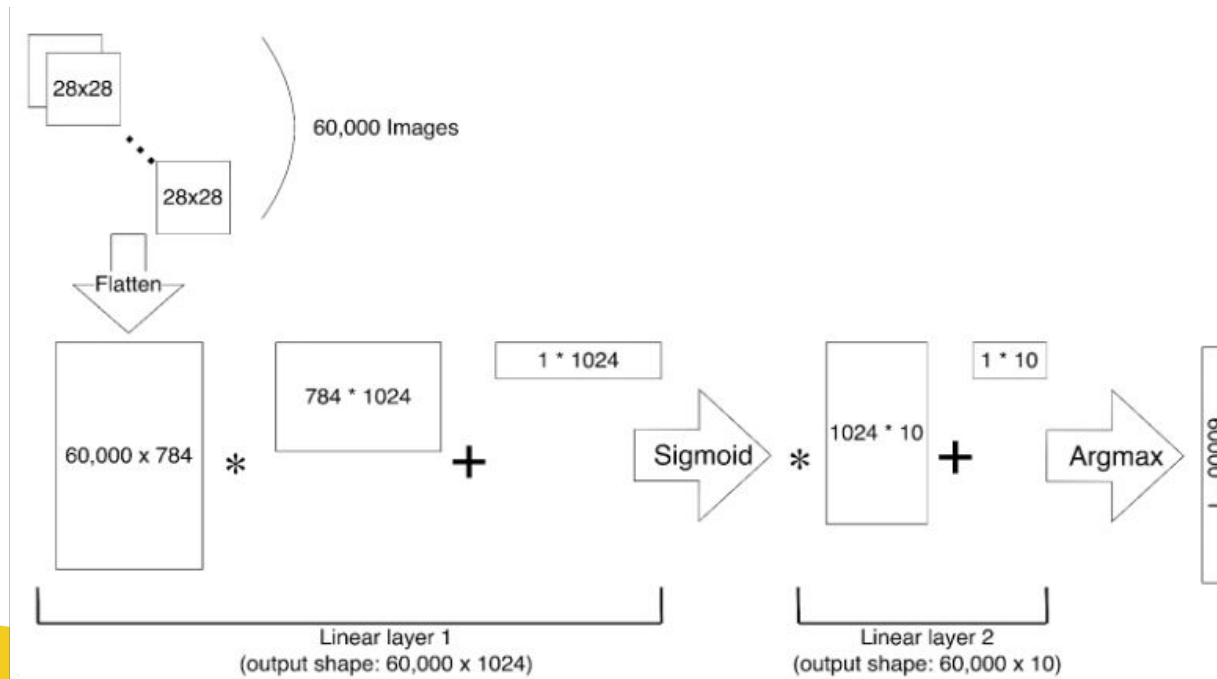
Deep Neural Network(DNN)

- Two fully-connected layer

```
simpleNN(  
    (nn): Sequential(  
        (0): Flatten(start_dim=1, end_dim=-1)  
        (1): Linear(in_features=784, out_features=1024, bias=True)  
        (2): Sigmoid()  
        (3): Linear(in_features=1024, out_features=10, bias=True)  
        (4): Softmax(dim=None)  
    )  
)
```

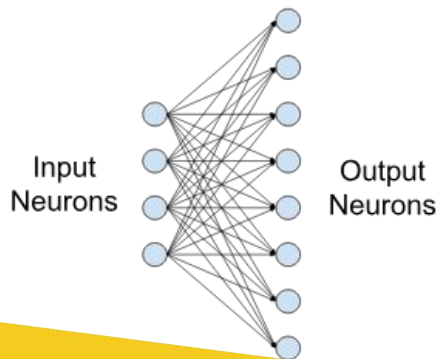
Deep Neural Network(DNN)

- Two fully-connected layer

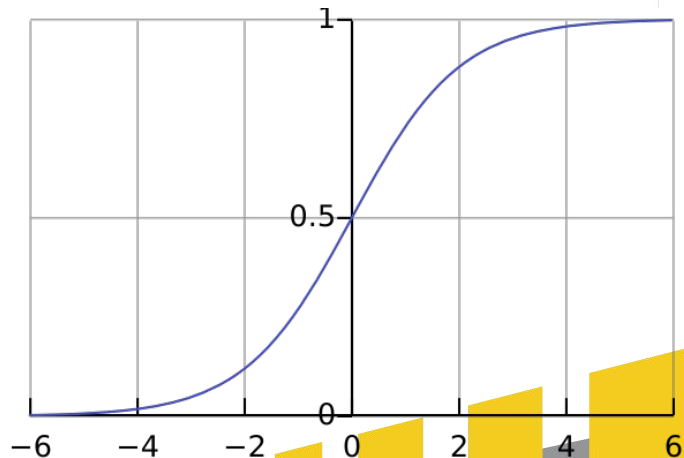


Deep Neural Network(DNN)

- Two fully-connected layer
 - First Linear Layer + Sigmoid Activation Function
 - Second Linear Layer + Argmax



$$\text{Sigmoid}(x) = \sigma(x) = \frac{1}{1 + \exp(-x)}$$



Task

- Given the sequential version of DNN code that runs on CPU.
- Try to parallel the matrix computation in the neural network.
- Using OpenACC
- The sequential code requires about 30~40 seconds

Task

- Provided files:
 - Sequential code
 - pretrained model weights(since the task is inference)
 - Makefile
- Please only modify the functions with TODO label.

Workflow

1. `cp -r /home/pp24/share/lab-mnist ~/lab_mnist`
2. `cd ~/lab_mnist`
3. `module load nvhpc-nompi/24.9`
4. Parallel the TODOs
5. compile the program : `make`
6. run the DNN program : `srun --gres=gpu:1 ./mnist`
7. judge: `lab-mnist-judge`
8. Scoreboard: [mnist](#)

Result

- Inference accuracy should be **97.8183%** (the parallelization should not affect the accuracy)

Result

Inference accuracy: 97.8183%

```
-----  STATS  -----  
Time for initializing CUDA device:      473 m.s.  
Time for reading MNIST data & weights: 375 m.s.  
Time for inferencing                    : 366 m.s.  
Time for calculating accuracy           : 7 m.s.  
----- END OF STATS -----
```

Judge Result

MNIST	350.00	accepted
ExtMNIST	1395.00	accepted

Submission

Submit your code and Makefile (optional) to eeclass before **10/31 23:59**

- mnist.cpp
- Makefile (Optional)