

1	2	3	4	5	Total
/20	/25	/20	/20	/15	

## Machine learning class mid-term report Ajou University

### -Instruction-

You must use PyTorch for your implementation. The code and screenshot of print output should be submitted in a single compressed file. (Zip, 7zip, tar, and so on).

\* All source code must be based on `midterm_baseline.ipynb` file. That is, in order to implement the activation function in problem 1, the inside of `class Net(nn.Module)` in the `midterm_baseline.ipynb` file must be modified. Please do not make your own baseline other than `midterm_baseline.ipynb`.

\* In the report, you just need to submit the modified source code part and the result (accuracy). You can paste the modified code and result as text or a screenshot in your report.

\* After modifying `midterm_baseline.ipynb` according to the description of each problem, you do not need to submit all source code, just submit the code text or screenshot of the modified part.

\* Please submit only the modified part of the code in the `midterm_baseline.ipynb` file. If you have modified the `batch_size`, learning rate, training code, or any part of the code in the `midterm_baseline.ipynb` file, you must submit the modified part.

\* Any code that is open to the public online is available for use. However, you should not get personal help from other students in our class.

\* You can freely ask questions about the homework format, submission method, etc. But I can't help you with direct error correction.

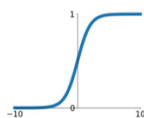
\* The report should not exceed 10 pages in total. The deadline for submission is November 17th at 7:00 PM Korean Standard Time.

1. [Activation function 1] Implement activation functions such as Sigmoid and tanh and compare the accuracy between ReLU and Sigmoid and tanh. **You can use PyTorch's built-in function.** You must modify the `midterm_baseline.ipynb` file to implement these activation functions.

In summary, we need to 1) implement Sigmoid and tanh and 2) compare the accuracy between Sigmoid and tanh with ReLU. [20 points]

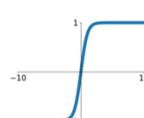
**Sigmoid**

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



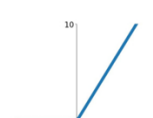
**tanh**

$$\tanh(x)$$



**ReLU**

$$\max(0, x)$$

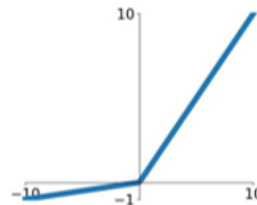


2. [Activation function 2] In this problem, please 1) implement Leaky ReLU by PyTorch NN Modules and 2) report the accuracy. **You can not use PyTorch's built-in function in this problem.** You must modify the midterm\_baseline.ipynb file to implement these activation functions.

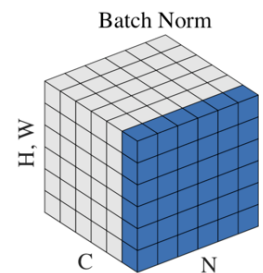
Reference code

[https://pytorch.org/tutorials/beginner/examples\\_nn/two\\_layer\\_net\\_module.html](https://pytorch.org/tutorials/beginner/examples_nn/two_layer_net_module.html)

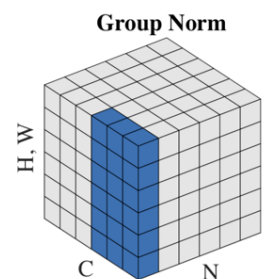
**Leaky ReLU**  
 $\max(0.1x, x)$



3. [Normalization 1] In this problem, 1) implement batch normalization and 2) compare the accuracy between with batch normalization neural network and with-out batch normalization neural network. **You can use PyTorch's built-in function.** You must modify the midterm\_baseline.ipynb file to implement these activation functions.



4. [Normalization 2] In this problem, 1) implement group normalization and 2) compare the accuracy between batch normalization neural network and group normalization neural network. **You can use PyTorch's built-in function.** You must modify the midterm\_baseline.ipynb file to implement these activation functions.



5. [Normalization 3] In this problem, 1) report and 2) discuss the results of how the performance of the batch normalization neural network and the group normalization neural network changes as the batch size changes. For two neural networks, batch normalization neural network and group normalization neural network, report the performance when the batch size is 2, 16, and 128, respectively, and if the performance (accuracy) of a specific neural network is changed according to the batch size, explain the reason.