Question 1 (2 point)

- 1a. What kinds of experiments did you run in your attempts to pass the `test_add_dataset` and `test_saved_add_dataset` cases? What hyperparameters seemed to be the most important? How does this match or conflict with your intuition about which hyperparameters are generally important?

In order to pass the test_add_dataset I changed the model initialization method from torch.nn.init.uniform_() to torch.nn. init.xavier_uniform_() and I did nothing else. It turns out that under fixed number of epochs the initial state of parameters could be determining in deciding whether can pass the test.

To pass test_saved_add_dataset(), I changed the number of epoch from 10 to 1000 in the python file to train my model until the loss reaches below the critical value. Epochs is the most important hyperparameters. Beside this "brute-force" solution, I also finetuned the model structure by adding a ReLu layer, increasing the number of hidden layers. The improvement of model performance as the complexity of model improves matches with my intuition.

- 1b. What kinds of experiments did you run in your attempts to pass the `test_saved_multiply_dataset` cases? What hyperparameters seemed to be the most important? How does this match or conflict with your intuition about which hyperparameters are generally important?

As above, I increased the epochs from 10 to 1000. I also added a ReLu layer and increase the number_of_hidden_layers to 3 to increase the complexity of neural network. I think the number of epochs is still the most important factor. The positive correlation between the improvement of model complexity and the improvement of model ability matches my intuition.