Lab 1

Task 1

The first run with 10 iterations and a single unit does not well predict the target, this is due to the important loss value and the fact that there is a single neuron. By running the code several times the values are not exactly the same, this is due to the random initialization of the weights. As the number of iterations increases, the loss value decrease meaning that the accuracy of the prediction is increasing. As the number of iterations increases, the loss value decrease meaning that the accuracy of the prediction is increasing

Task 2

For a same number of unit and iterations, the XOR has a higher loss value than the AND. This is due because the AND operator requires a linear separator while a XOR need a more complex one. The iteration is not enough.

Task 3

Cause for only one neuron, it is hard to predict the nonlinear boundary. With the increase of number of neuron, it is easier to achieve the ability of prediction of nonlinear boundary. But here the reason if the accuracy didn’t improve is the number of iteration is not enough for so many neurons to train. I mean the epochs=2000 is not enough because now the function is actually much complex than that with only one neuron. If we change the epochs from 2000 to 20000, we can get a very good result with 10 neurons.

Task 4

With the increase of learning rate, the iteration step size of weights also increases.

With the initial parameters, the final loss value is 0.35.

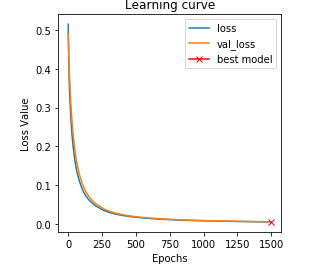
Keep the number of epochs constant, change the lr to 0.1, the result is very good. The final loss value almost close to 0.

Increase the number of epochs to 150 with lr 0.0001, the final loss value is 0.1. No

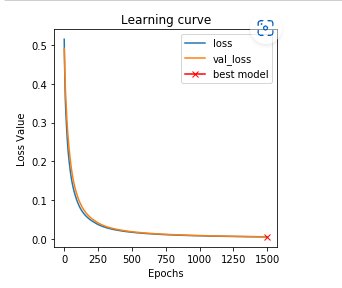
Increase the number of base dense to 256, the results didn’t change a lot. But if we increase the number of epoch, it will get better! I think with the complexity of the MLP, more iterations do we need!

Task 5

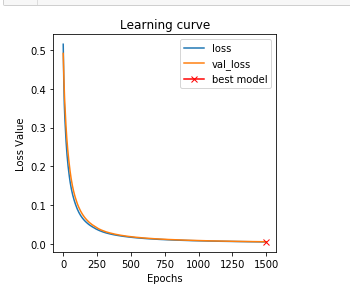
A



B Same



C For generalization power, the accuracy of validation group should be enough high at the same time of keeping high accuracy on training group.



D Convolutional layer: Extracting features. The feature of "incomplete connection, parameter sharing" greatly reduces network parameters, ensures network sparsity, and prevents overfitting. "Parameter sharing" is possible because of the locally correlated nature of the sample.

E Generally from the last layer after convolution to the output, need use the fully connected layer because those represents high-level features in the data and the fully connected layeer is the cheap way of leanring non linear combinations of these features.

F 1. Compared with MLP, LeNet uses fewer parameters and obtains better results.

2. Maxpool is designed to extract features

G If the problem is a 2 classification(label) problem, I think one neuron is enough and sigmoid is a good function to achieve two labels classification.