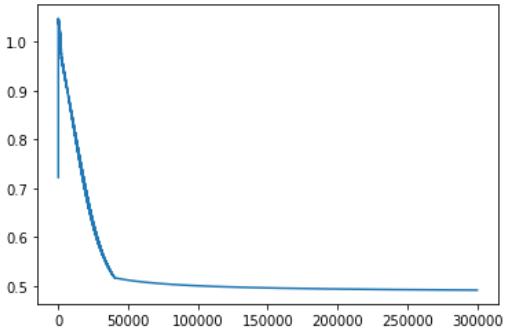
Binary classification using 2-layerd network #3-2

컴퓨터소프트웨어학과 2017029470 김종원

1. Accuracy

- $W^{[1]} = [\,[-0.5], [-0.5]\,],\ W^{[2]} = [\,[0.5, -0.5]\,]$
- $b1 = \big[[0.8], [0.6]\,\big],\ \ b2 = \big[\,[0.5]\big],\ \ \alpha = 0.001$

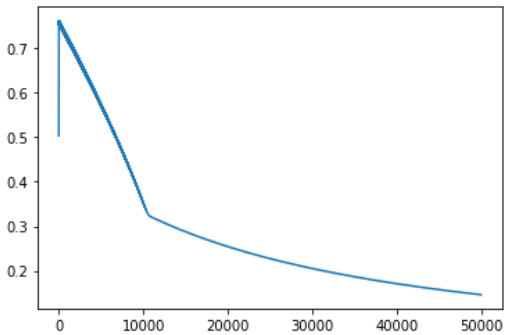
	m = 10, n = 1000, k = 5000	m = 100, n = 1000, k = 5000	m = 10000, n = 1000, k = 5000
Accuracy (m train samples)	**60.0**	**53.0**	**50.2**
Accuracy (n test samples)	**52.8**	**49.2**	**50.0**
	m = 10000, n = 1000, k = 10	**m = 10000, n = 1000, k = 100**	**m = 1000, n = 1000, k = 5000**
Accuracy (m train samples)	**50.5**	**50.62**	**56.1**
Accuracy (n test samples)	**50.9**	**48.5**	**55.0**



Even though the unit was increased by one more in the first layer, the accuracy did not improve. The graph on the left shows the cost function value as the number of iterations increases. It can be seen that the value of the cost function hardly changed after k = 50,000.

2. Estimated unknown function parameters W & b

The initial values of W & b were set differently, and a few additional tests were performed.
W1, W2, b1, b2 are initialized with a random real number between -0.5 and 0.5 using the random function in the numpy library. Most of it appeared around 50%, and there was no difference from the existing accuracy above.



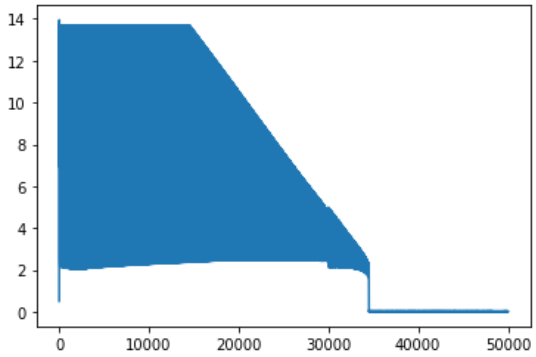
$W^{[1]} = [\,[-1.0], [-1.0]\,],\ W^{[2]} = [\,[1.0, -1.0]\,]$

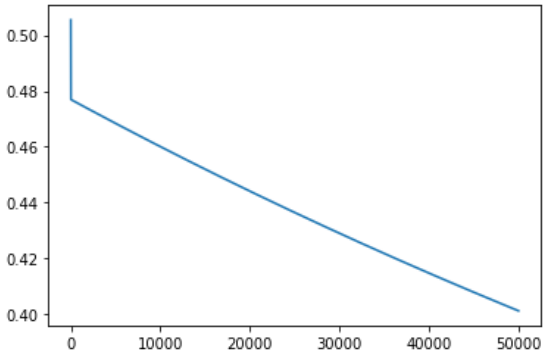
$b1 = [\,[90.0], [270.0]\,],\ \ b2 = \big[[0.]\big],\ \ \alpha = 0.001$

However, if I set W1, W2, b1, and b2 as above and find the accuracy, the cost was high at the beginning and the accuracy came out about 50%. But as the number of iterations increased, the value of the cost function continued to decrease, and in the end, the accuracy came out to 100%. By setting the initial value of the hyperparameter well, even when the cost function value was the largest, it was about 0.7, which was different from the previous testing results.

3. Empirically determined (best) hyper parameter, α

If $\alpha = 0.01$ is set, overflow occurs in the process of finding A1, causing the value of the cost function to become very large in the middle. Also, when a = 0.0001 is set, the rate at which the value of the cost function decreases is very slow, so the number of iterations must be increased.





4. Discussion

Why did overflow occur in the process of finding A1?

When the result of exp exceeds the maximus number representable by value's floating point data type format, an "overflow encountered in exp" warning occurred. It is just warning not an error, so I could just ignore it. However, I have not been able to figure out why such an error occurs only when $\alpha = 0.01$.

Correlation between the number of units and execution time

Even for a neural network with the same number of layers, the larger the number of units, the longer the execution time.