Binary classification using logistic regression

Practice#3-2

Report by: 완(2019007901)

Changelog for practice#3-2

- Changed the number of units in first hidden layer from 1 to 2. This is done by changing the dimensions of parameter W and b as below.

```
W1 = np.random.rand(2,1)
B1 = np.random.rand(2,1)
W2 = np.random.rand(1,2)
B2 = np.random.rand(1,1)
```

Estimated unknown function W & b

For our experiments, we will set the number of samples according to the assignment specifications (M=10000, N = 1000) and the number of iterations K will be set to 5000. Even after increasing the number of unit is the first hidden layer, we saw an increasing accuracy by only ~2% compared to previous assignment. Therefore, we conclude the best W and b parameters are as below.

```
[w1, b1, w2, b2] = [[[0.59945082]
 [0.52713867]], [[  5.83291433]
 [-46.90032418]], [[ 1.03895801 -2.2238745 ]], [[0.62344956]]]
Cost: 0.228928
Cost with n test samples =  0.30353332328061894
Accuracy for 'm' train samples: 75.09%
Accuracy for 'n' test samples: 75.3%
```

Empirically determined (best) hyper parameter, α

We will use the same learning rate from the previous assignment, which is 0.01. When we tried values lower that 0.01, we observed that the gradient descent has lower acceleration and if we use value higher than 0.01, the cost keeps oscillation between iterations. Therefore, we conclude that 0.01 is the best learning rate parameter for this program.

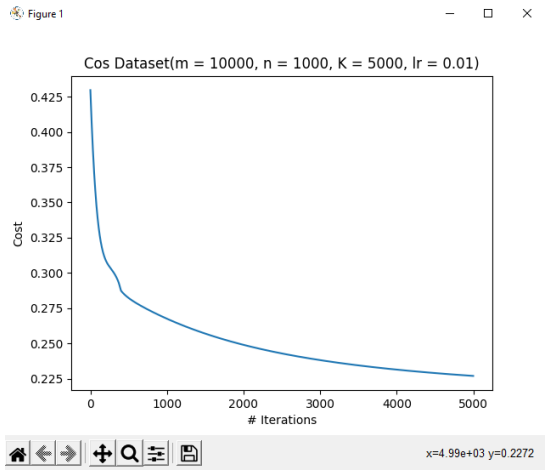
Accuracy

	m=10, n=1000, K=5000	m=100, n=1000, K=5000	m=10000, n=1000, K=5000
Accuracy (‘m’ train samples)	70.0%	72.0%	74.88%
Accuracy (‘n’ test samples)	52.2%	73.0%	77.2%

	m=10000, n=1000, K=10	m=10000, n=1000, K=100	m=1000, n=1000, K=5000
Accuracy (‘m’ train set)	49.21%	50.17%	73.3%
Accuracy (‘n’ test samples)	50.9%	52.2%	75.3%

Discussion

From the accuracy table above, even after adding the number of units in the first hidden layer, we only have a very little increase in accuracy compared to the previous assignment where the number of units in all hidden layer are only 1. The loss graph for this assignment is as below.



Based on the graph above, we still see the early saturation in loss reduction after 5000 iterations. We also tried to increase the number of iterations and also increased the learning rate, but it still gave the same accuracy. Therefore, we conclude that, adding only one additional unit in the first hidden layer is not enough for the network to reduce the loss further. Perhaps, by adding the layer might give a better accuracy result.