Neural Networks and Deep Learning

Week-2

In week 1, we have given a very basic definition of what a neural network is. Now, we take the 1st formal step into neural network by understanding how logistic regression works, we see it as a neural network with ‘0’ hidden layers to build a cat classifier.

We understand how to use sigmoid activation function, to determine whether a given image is cat or non-cat. We get the intuition behind cost function, which in the simplest terms mean penalising the algorithm heavily for every wrong decision it makes, to better fit the training data.

We understood how vectorization significantly improves the computation time instead of using loops.

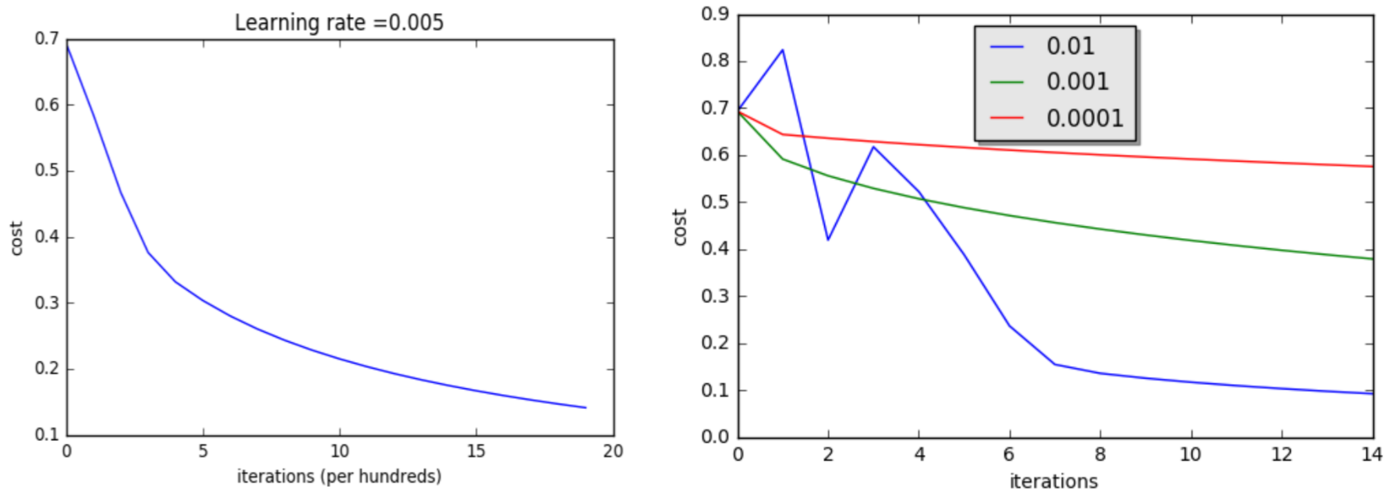
We have seen how Broadcasting works in python, which is helpful in most of the situations but at times it can introduce subtle undesired bugs.

After running logistic regression on the cat classifier, we got a training accuracy of 99% and Test accuracy of 70% indicating that we are over-fitting the data.

We have seen, how the results vary if we change the value of the parameter α (learning-rate).

Result:

Learning Rate	Train Accuracy	Test Accuracy
0.05	99.04 %	70 %
0.01	99.51 %	68 %
0.001	88.99 %	64 %
0.0001	68.42 %	36 %



Both the graphs are on Test Set only.

Conclusion:

1. The graph on the left (Cost Vs Number of Iterations) confirms that our algorithms work well, as cost is decreasing with number of iterations.

2. The graph on the right (Cost Vs Number of Iterations (for different learning rates))

- Different learning rates give different costs and thus different predictions results
- If the learning rate is too large (0.01), the cost may oscillate up and down. It may even diverge (though in this example, using 0.01 still eventually ends up at a good value for the cost).
- A lower cost doesn't mean a better model. You have to check if there is possibly overfitting. It happens when the training accuracy is a lot higher than the test accuracy.
- In deep learning, it is usually recommended that you:
    - Choose the learning rate that better minimizes the cost function.
    - If your model overfits, use other techniques to reduce overfitting.