Week 10-1: Large Scale Machine Learning

1. Suppose you are training a logistic regression classifier using stochastic gradient descsent. You find that the cost (say, $cost\big[\theta, \big(x^{(i)}, y^{(i)}\big)\big]$, averaged over the last 500 examples), plotted as a function of the number of iterations, is slowly increasing over time. Which of the following changes are likely to help?

Try using a smaller learning rate α. The current α is too large to observe convergence.

2. Which of the following statements about stochastic gradient descent are true? Check all that apply.

(a) If you have a huge training set, then stochastic gradient descent may be much faster than batch gradient descent.
(b) Before running stochastic gradient descent, you should randomly shuffle (reorder) the training set.

3. Which of the following statements about online learning are true? Check all that apply.

(a) One of the advantages of online learning is that if the function we are modeling changes over time (such as if we are modeling the probability of users clicking on different URLs, and user taste/preferences are changing over time) the online learning algorithm will automatically adapt to these changes.
(b) Online learning algorithms are usually best suited to problems where we have continuous/non-stop stream of data that we want to learn from.

4. Assuming that you have a very large training set, which of the following algorithms do you think can be parallelized using map-reduce and splitting the training set across different machines? Check all that apply.

(a) Linear regression trained using batch gradient descent.
(b) A neural network trained using batch gradient descent.

5. Which of the following statements about map-reduce are true? Check all that apply.

(a) When using map-reduce with gradient descent, we usually use a single machine that accumulates the gradients from each of the map-reduce machines, in order to compute the parameter update for that iteration.
(b) In order to parallelize a learning algorithm using map-reduce, the first step is to figure out how to express the main work done by the algorithm as computing sums of functions of training examples.
(c) If you have just 1 computer, but your computer has multiple CPUs or multiple computing cores, then map-reduce might be a viable way to parallelize your learning algorithm.