Large Scale Machine Learning

Question 1

Suppose you are training a logistic regression classifier using stochastic gradient descent. You find that the cost (say, $cost(\theta, (x^{(i)}, y^{(i)}))$, 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?

- [ ] Use fewer examples from your training set.
- [ ] This is not an issue, as we expect this to occur with stochastic gradient descent.
- [ ] Try averaging the cost over a smaller number of examples (say 250 examples instead of 500) in the plot.
- [x] **Try using a smaller learning rate α.**

Question 2

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

- [ ] Stochastic gradient descent is particularly well suited to problems with small training set sizes; in these problems, stochastic gradient descent is often preferred to batch gradient descent.
- [ ] In order to make sure stochastic gradient descent is converging, we typically compute $J_{train}(\theta)$ after each iteration (and plot it) in order to make sure that the cost function is generally decreasing.
- [x] **In each iteration of stochastic gradient descent, the algorithm needs to examine/use only one training example.**
- [x] **You can use the method of numerical gradient checking to verify that your stochastic gradient descent implementation is bug-free. (One step of stochastic gradient descent computes the partial derivative $\frac{\partial}{\partial \theta_j} cost(\theta, (x^{(i)}, y^{(i)}))$.)**

Question 3

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

- [x] **When using online learning, in each step we get a new example (x, y), perform one step of (essentially stochastic gradient descent) learning on that example, and then discard that example and move on to the next.**

- [ ] Online learning algorithms are most appropriate when we have a fixed training set of size ***m*** that we want to train on.
- [ ] One of the **disadvantages** of online learning is that it requires a large amount of computer memory/disk space to store all the training examples we have seen.
- [x] **One of the advantages of online learning is that if the function we're modeling changes over time (such as if we are modeling the probability of users clicking on different URLs, and user tastes/preferences are changing over time), the online learning algorithm will automatically adapt to these changes.**

Question 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.

- [x] **Linear regression trained using batch gradient descent.**
- [x] **Computing the average of all the features in your training set $\mu = \frac{1}{m} \sum_{i=1}^{m} x^{(i)}$ (say in order to perform mean normalization).**
- [ ] Logistic regression trained using *stochastic gradient descent*.
- [ ] An online learning setting, where you repeatedly get a single example *(x, y)*, and want to learn from that single example before moving on.

Question 5

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

- [x] **Because of network latency and other overhead associated with map-reduce, if we run map-reduce using N computers, we might get less than an N-fold speedup compared to using 1 computer.**
- [ ] If we run map-reduce using N computers, then we will always get at least an N-fold speedup compared to using 1 computer.
- [x] **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.**
- [ ] Running map-reduce over N computers requires that we split the training set into N^2 pieces.