1 point

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?

- ( ) Try averaging the cost over a larger number of examples (say 1000 examples instead of 500) in the plot.
- (•) Try using a smaller learning rate α.
- ( ) Try using a larger learning rate α.
- ( ) This is not an issue, as we expect this to occur with stochastic gradient descent.

1 point

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?

- (•) Try halving (decreasing) the learning rate α, and see if that causes the cost to now consistently go down; and if not, keep halving it until it does.
- ( ) This is not possible with stochastic gradient descent, as it is guaranteed to converge to the optimal parameters θ.
- ( ) Try averaging the cost over a smaller number of examples (say 250 examples instead of 500) in the plot.
- ( ) Use fewer examples from your training set.

1 point

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

- [x] In the approach to online learning discussed in the lecture video, we repeatedly get a single training example, take one step of stochastic gradient descent using that example, and then move on to the next example.
- [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.
- [ ] 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.
- [ ] One of the advantages of online learning is that there is no need to pick a learning rate α.

1
point

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.

- [ ] An online learning setting, where you repeatedly get a single example (x, y), and want to learn from that single example before moving on.
- [x] A neural network trained using batch gradient descent.
- [x] Linear regression trained using batch gradient descent.
- [ ] Logistic regression trained using stochastic gradient descent.

1
point

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] 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.
- [x] Logistic regression trained using batch gradient descent.
- [ ] Linear regression trained using stochastic gradient descent.

1
point

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.

- [ ] Linear regression trained using stochastic gradient descent.
- [ ] Logistic regression trained using stochastic gradient descent.
- [x] Logistic 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).

1 point

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

- [x] If you have only 1 computer with 1 computing core, then map-reduce is unlikely to help.
- [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.
- [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.

✓

1 point

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

- [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.
- [x] 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.
- [ ] Running map-reduce over N computers requires that we split the training set into N^2 pieces.
- [x] If you have just 1 computer, but your computer has multiple CPUs or multiple cores, then map-reduce might be a viable way to parallelize your learning algorithm.

✓