Feedback — XVII. Large Scale Machine Learning

Help

You submitted this quiz on **Mon 19 May 2014 10:22 PM PDT**. You got a score of **5.00** out of **5.00**.

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?

Your Answer		Score	Explanation
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.	✔	1.00	Such a plot indicates that the algorithm is diverging. Decreasing the learning rate α means that each iteration of stochastic gradient descent will take a smaller step, thus it will likely converge instead of diverging.
This is not an issue, as we expect this to occur with stochastic gradient descent.			
This is not possible with stochastic gradient descent, as it is guaranteed to converge to the optimal parameters θ.			
Use fewer examples from your training set.			

Total	1.00 / 1.00

Question 2

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

Your Answer		Score	Explanation
☑ 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 $\frac{\partial}{\partial \theta_j} cost(\theta, (x^{(i)}, y^{(i)}))$.)	✔	0.25	Just as with batch gradient descent, you can compute the derivative numerically and compare it to your computed value to check for correctness.
☐ One of the advantages of stochastic gradient descent is that it uses parallelization and thus runs much faster than batch gradient descent.	✔	0.25	Stochastic gradient descent still runs in series, one example at a time.
☑ If you have a huge training set, then stochastic gradient descent may be much faster than batch gradient descent.	✔	0.25	Because stochastic gradient descent can make progress after only a few examples, it can converge much more quickly than batch gradient descent.
☐ Suppose you are using stochastic gradient descent to train a linear regression classifier. The cost function $J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$ is guaranteed to decrease after every iteration of the stochastic gradient descent algorithm.	✔	0.25	Since each iteration of stochastic gradient descent takes into account only one training example, it is not guaranteed that every update lowers the cost function over the entire training set.
Total		1.00 / 1.00	

Question 3

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

Your Answer		Score	Explanation
☐ 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.	✔	0.25	Since online learning algorithms do not save old examples, they can be very efficent in terms of computer memory and disk space.
☐ When using online learning, you must save every new training example you get, as you will need to reuse past examples to re-train the model even after you get new training examples in the future.	✔	0.25	Online learning algorithms throw away old examples, incorporating them only once when they are first seen.
☑ 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.	✔	0.25	This is essentially the definition of online learning.
☑ One of the advantages of online learning is that if the function we're modeling changes over time (such as if	✔	0.25	Online learning algorithms move toward correctly classifying the most recent examples, so as user tastes change and we receive new, different data, the algorithm will automatically take those into account.

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.		
Total	1.00 / 1.00	

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.

Your Answer		Score	Explanation
☑ Linear regression trained using batch gradient descent.	✔	0.25	You can split the dataset into N smaller batches, compute the gradient for each smaller batch on one of N separate computers, and then average those gradients on a central computer to use for the gradient update.
☑ Logistic regression trained using batch gradient descent.	✔	0.25	You can split the dataset into N smaller batches, compute the gradient for each smaller batch on one of N separate computers, and then average those gradients on a central computer to use for the gradient update.
☐ Linear regression trained using stochastic gradient descent.	✔	0.25	Since stochastic gradient descent processes one example at a time and updates the parameter values after each, it cannot be easily parallelized.
☐ An online learning setting, where you repeatedly get a	✔	0.25	Since you process one example at a time, this algorithm cannot be easily parallelized.

single example (x, y), and want to learn from that single example before moving on.	
Total	1.00 / 1.00

Question 5

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

Your Answer		Score	Explanation
☑ In order to parellelize 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.	✔	0.25	In the reduce step of map-reduce, we sum together the results computed by many computers on the training data.
☑ 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.	✔	0.25	The maximum speedup possible is N-fold, and it is unlikely you will get an N-fold speedup because of the overhead.
☑ If you have only 1 computer with 1 computing core, then map-reduce is unlikely to help.	✔	0.25	Map-reduce is a useful model for parallel computation.
☐ If we run map-reduce using N computers, then we will always get at least an N-fold speedup compared to using 1 computer.	✔	0.25	The maximum speedup possible is N-fold, and it is unlikely you will get an N-fold speedup because of the overhead.
Total		1.00 / 1.00	