Feedback — XVII. Large Scale Machine Learning

Help

You submitted this quiz on **Tue 26 Aug 2014 8:40 AM 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 averaging the cost over a smaller number of examples (say 250 examples instead of 500) in the plot.			
Try using a larger learning rate α.			
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 possible with stochastic gradient descent, as it is guaranteed to converge to the			

optimal parameters θ.

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
☐ 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.	✔	0.25	Stochastic gradient descent is preferred when you have a large training set size; if the data set is small, then the summation over examples in batch gradient descent is not an issue.
☑ 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.
☑ Before running stochastic gradient descent, you should randomly shuffle (reorder) the training set.	✔	0.25	It is a good idea to shuffle your data so that gradient descent does not take a long sequence of steps based on a biased subset of the data (such as a long run of $y = 0$ examples in logistic regression).
☐ One of the advantages of stochastic gradient descent is that it uses parallelization and thus	✔	0.25	Stochastic gradient descent still runs in series, one example at a time.

runs much faster than batch gradient descent.			
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.
☐ One of the advantages of online learning is that there is no need to pick a learning rate α.	✔	0.25	One still must choose a learning rate to use online learning.
☑ 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.	✔	0.25	This is one good approach to online learning discussed in the lecture video.
☑ Online learning algorithms are usually best suited to problems	✔	0.25	Such a stream of data is well-suited to online learning because online learning does not save old training examples, but instead uses them once and

were we have a continuous/non-stop stream of data that we want to learn from.			then throws them out.
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 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.
☐ A neural network 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.
☑ A neural network 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.
☑ 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).	✔	0.25	You can split the dataset into N smaller batches, compute the feature average of each smaller batch on one of N separate computers, and then average those results on a central computer to get the final result.

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.
☑ 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.	✔	0.25	Such a setup allows us to use many computers to do the hard work of gradient computation while making the parameter update simple, as it occurs in one place.
☐ Linear regression and logistic regression can be parallelized using map-reduce, but not neural network training.	✔	0.25	All three can be parallelized using map-reduce.
☑ 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	✔	0.25	The maximum speedup possible is N-fold, and it is unlikely you will get an N-fold speedup because of the overhead.

speedup compared to using 1 computer.

Total	1.00 / 1.00