Congratulations! You passed!

Grade received 100% **Latest Submission Grade** 100% **To pass** 80% or higher

Go to next item

1. Using the notation for mini-batch gradient descent. To what of the following does $a^{[2]\{4\}(3)}$ correspond? **1 / 1 point**

- The activation of the third layer when the input is the fourth example of the second mini-batch.
- (selected) The activation of the second layer when the input is the third example of the fourth mini-batch.
- The activation of the fourth layer when the input is the second example of the third mini-batch.
- The activation of the second layer when the input is the fourth example of the third mini-batch.

Expand

Correct
Yes. In general $a^{[l]\{t\}(k)}$ denotes the activation of the layer l when the input is the example k from the mini-batch t.

2. Suppose you don't face any memory-related problems. Which of the following make more use of vectorization. **1 / 1 point**

- Mini-Batch Gradient Descent with mini-batch size $m/2$.
- Stochastic Gradient Descent
- (selected) Batch Gradient Descent
- Stochastic Gradient Descent, Batch Gradient Descent, and Mini-Batch Gradient Descent all make equal use of vectorization.

Expand

Correct
Yes. If no memory problem is faced, batch gradient descent processes all of the training set in one pass, maximizing the use of vectorization.

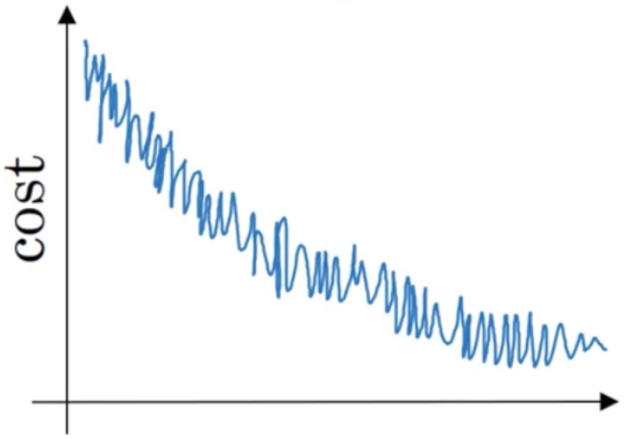
3. Which of the following is true about batch gradient descent? **1 / 1 point**

- It is the same as stochastic gradient descent, but we don't use random elements.
- (selected) It is the same as the mini-batch gradient descent when the mini-batch size is the same as the size of the training set.
- It has as many mini-batches as examples in the training set.

Expand

Correct
Correct. When using batch gradient descent there is only one mini-batch thus it is equivalent to batch gradient descent.

4. Suppose your learning algorithm's cost J, plotted as a function of the number of iterations, looks like this: **1 / 1 point**



Which of the following do you agree with?

- (●) If you're using mini-batch gradient descent, this looks acceptable. But if you're using batch gradient descent, something is wrong.
- ( ) Whether you're using batch gradient descent or mini-batch gradient descent, this looks acceptable.
- ( ) Whether you're using batch gradient descent or mini-batch gradient descent, something is wrong.
- ( ) If you're using mini-batch gradient descent, something is wrong. But if you're using batch gradient descent, this looks acceptable.

Expand

Correct

5. Suppose the temperature in Casablanca over the first two days of March are the following: **1 / 1 point**

March 1st: $\theta_1 = 30^\circ \text{ C}$

March 2nd: $\theta_2 = 15^\circ \text{ C}$

Say you use an exponentially weighted average with $\beta = 0.5$ to track the temperature: $v_0 = 0, v_t = \beta v_{t-1} + (1-\beta)\,\theta_t$. If v_2 is the value computed after day 2 without bias correction, and $v_2^{\text{corrected}}$ is the value you compute with bias correction. What are these values?

- ( ) $v_2 = 20, v_2^{\text{corrected}} = 20.$
- ( ) $v_2 = 20, v_2^{\text{corrected}} = 15.$
- ( ) $v_2 = 15, v_2^{\text{corrected}} = 15.$
- (●) $v_2 = 15, v_2^{\text{corrected}} = 20.$

Expand

Correct

Correct. $v_2 = \beta v_{t-1} + (1-\beta)\,\theta_t$ thus $v_1 = 15, v_2 = 15$. Using the bias correction $\frac{v_t}{1-\beta^t}$ we get $\frac{15}{1-(0.5)^2} = 20$.

6. Which of these is NOT a good learning rate decay scheme? Here, t is the epoch number. **1 / 1 point**

- ( ) $\alpha = e^{-0.01\,t}\alpha_0.$
- (●) $\alpha = 1.01^t\,\alpha_0$
- ( ) $\alpha = \frac{\alpha_0}{1+2t}$

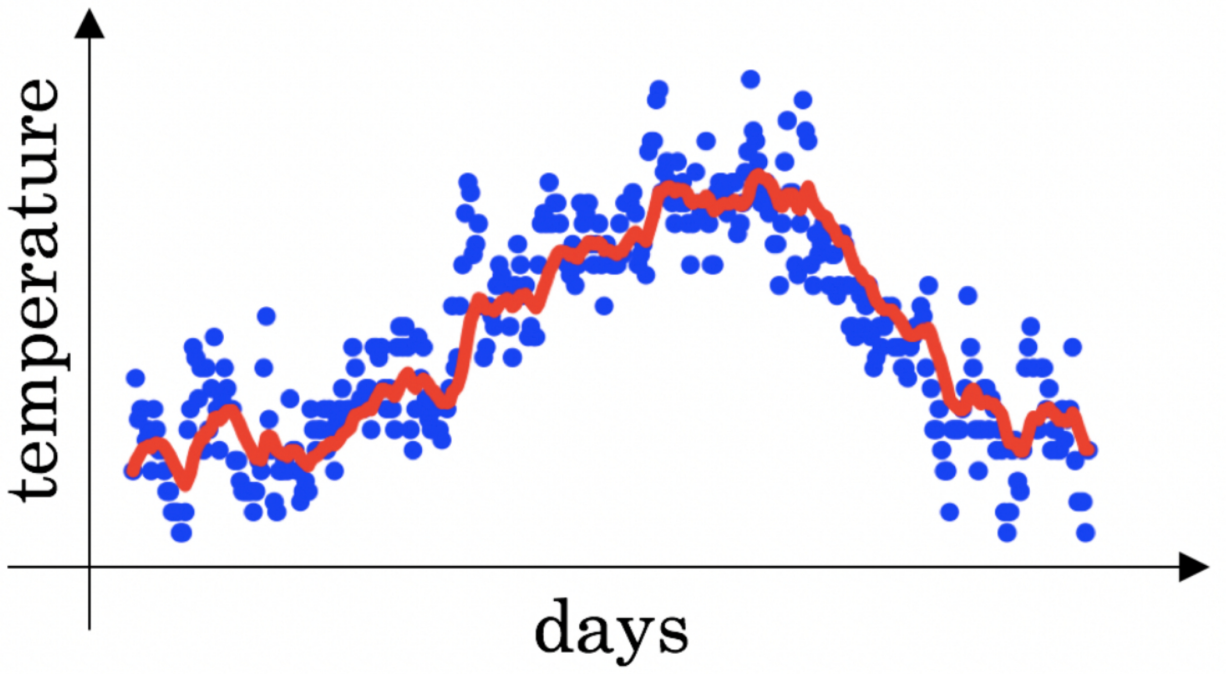
$1 + 3t$

- $\alpha = \frac{\alpha_0}{\sqrt{1+t}}$.

Expand

Correct
Correct. This is not a good learning rate decay since it is an increasing function of t.

7. You use an exponentially weighted average on the London temperature dataset. You use the following to track the temperature: $v_t = \beta v_{t-1} + (1-\beta)\theta_t$. The red line below was computed using $\beta = 0.9$. What would happen to your red curve as you vary β? (Check the two that apply) **1 / 1 point**



- [ ] Decreasing β will shift the red line slightly to the right.
- [x] Increasing β will shift the red line slightly to the right.

Correct
True, remember that the red line corresponds to $\beta = 0.9$. In the lecture we had a green line $\beta = 0.98$ that is slightly shifted to the right.

- [x] Decreasing β will create more oscillation within the red line.

Correct
True, remember that the red line corresponds to $\beta = 0.9$. In lecture we had a yellow line $\beta = 0.98$ that had a lot of oscillations.

- [ ] Increasing β will create more oscillations within the red line.

Expand

Correct
Great, you got all the right answers.

8. Which of the following are true about gradient descent with momentum? **1 / 1 point**

- [x] Increasing the hyperparameter β smooths out the process of gradient descent.

Correct
Correct. Gradient descent with momentum makes use of moving averages, which smooths out the gradient descent process.

- [ ] It decreases the learning rate as the number of epochs increases.
- [x] It generates faster learning by reducing the oscillation of the gradient descent process.

Correct
Correct. The use of momentum makes each step of the gradient descent more efficient by reducing oscillations.

- [x] Gradient descent with momentum makes use of moving averages.

Correct
Correct. Gradient descent with momentum makes use of moving averages, which smooths out the gradient descent process.

Expand

Correct
Great, you got all the right answers.

9. Suppose batch gradient descent in a deep network is taking excessively long to find a value of the parameters that achieves a small value for the cost function $\mathcal{J}(W^{[1]}, b^{[1]}, \ldots, W^{[L]}, b^{[L]})$. Which of the following techniques could help find parameter values that attain a small value for $\mathcal{J}$? (Check all that apply) **1 / 1 point**

- [x] Try mini-batch gradient descent.

Correct
Yes. Mini-batch gradient descent is faster than batch gradient descent.

- [x] Normalize the input data.

Correct
Yes. In some cases, if the scale of the features is very different, normalizing the input data will speed up the training process.

- [ ] Try initializing the weight at zero.
- [x] Try using Adam.

Correct
Yes. Adam combines the advantages of other methods to accelerate the convergence of the gradient descent.

Expand

Correct
Great, you got all the right answers.

10. In very high dimensional spaces it is most likely that the gradient descent process gives us a local minimum than a saddle point of the cost function. True/False? **1 / 1 point**

- ( ) True
- (•) False

Expand

Correct

Correct. Due to the high number of dimensions it is much more likely to reach a saddle point, than a local minimum.