Try again once you are ready

Grade received 70% | Latest Submission Grade 70% | To pass 80% or higher

Try again

1. Which notation would you use to denote the 4th layer's activations when the input is the 7th example from the 3rd mini-batch? **1 / 1 point**

- $a^{[7]\{3\}(4)}$
- $a^{[4]\{3\}(7)}$
- $a^{[3]\{7\}(4)}$

Expand

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

2. Which of these statements about mini-batch gradient descent do you agree with? **1 / 1 point**

- When the mini-batch size is the same as the training size, mini-batch gradient descent is equivalent to batch gradient descent.
- You should implement mini-batch gradient descent without an explicit for-loop over different mini-batches so that the algorithm processes all mini-batches at the same time (vectorization).
- Training one epoch (one pass through the training set) using mini-batch gradient descent is faster than training one epoch using batch gradient descent.

Expand

Correct
Correct. Batch gradient descent uses all the examples at each iteration, this is equivalent to having only one mini-batch of the size of the complete training set in mini-batch gradient descent.

3. Why is the best mini-batch size usually not 1 and not m, but instead something in-between? Check all that are true. **1 / 1 point**

- [x] If the mini-batch size is m, you end up with batch gradient descent, which has to process the whole training set before making progress.

  Correct

- [ ] If the mini-batch size is m, you end up with stochastic gradient descent, which is usually slower than mini-batch gradient descent.
- [x] If the mini-batch size is 1, you lose the benefits of vectorization across examples in the mini-batch.

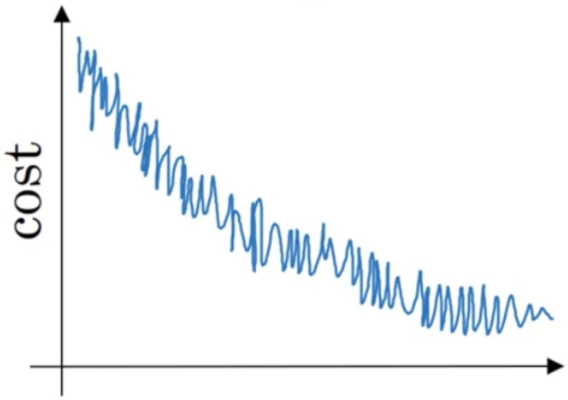
  Correct

- [ ] If the mini-batch size is 1, you end up having to process the entire training set before making any progress.

Expand

Correct
Great, you got all the right answers.

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

Expand

Correct

5. Suppose the temperature in Casablanca over the first two days of March are the following: **0 / 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?

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

Expand

Incorrect

Incorrect. $$v_2 = \beta v_{t-1} + (1- \beta) \, \theta_t$$ 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 the following is true about learning rate decay? **1 / 1 point**

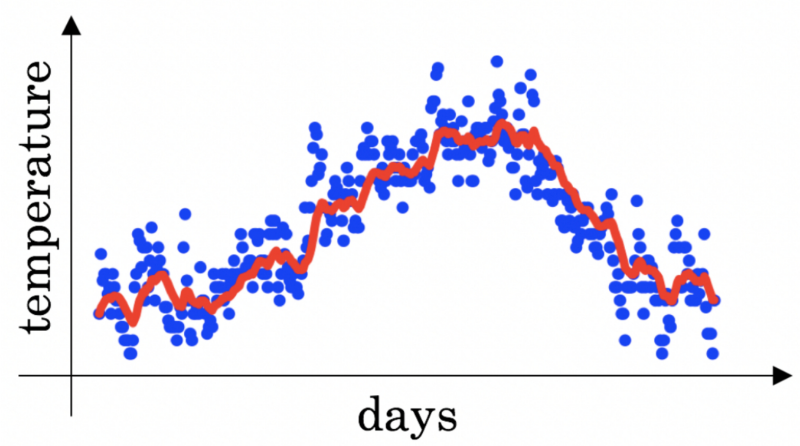
- The intuition behind it is that for later epochs our parameters are closer to a minimum thus it is more convenient to take smaller steps to prevent large oscillations.
- We use it to increase the size of the steps taken in each mini-batch iteration.
- It helps to reduce the variance of a model.
- The intuition behind it is that for later epochs our parameters are closer to a minimum thus it is more convenient to take larger steps to accelerate the convergence.

Expand

Correct
Correct. Reducing the learning rate with time reduces the oscillation around a minimum.

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.

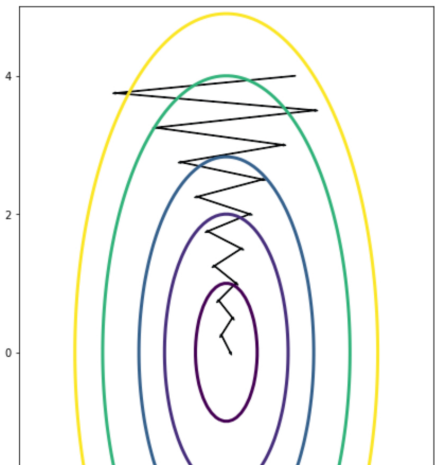
- [ ] Increasing β will create more oscillations within the red line.

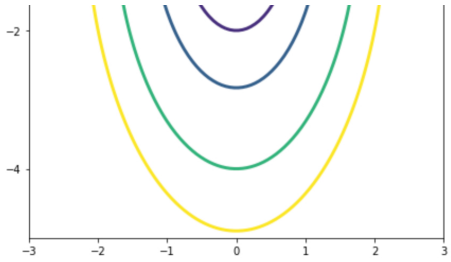
Expand

Correct
Great, you got all the right answers.

8. Consider the figure:

0 / 1 point





Suppose this plot was generated with gradient descent with momentum $\beta = 0.01$. What happens if we increase the value of β to 0.1?

- The gradient descent process starts oscillating in the vertical direction.
- The gradient descent process moves more in the horizontal and the vertical axis.
- The gradient descent process moves less in the horizontal direction and more in the vertical direction.
- The gradient descent process starts moving more in the horizontal direction and less in the vertical.

Expand

Incorrect
No. The use of a greater value of $$\beta$$ causes a more efficient process thus reducing the oscillation in the horizontal direction and moving the steps more in the vertical direction.

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 $\mathcal{J}(W^{[1]}, b^{[1]}, ..., W^{[L]}, b^{[L]})$. Which of the following techniques could help find parameter values that attain a small value for $\mathcal{J}$? (Check all that apply) **0 / 1 point**

- [x] Try mini-batch gradient descent.

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

- [ ] 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.

- [ ] Normalize the input data.

Expand

Incorrect
You didn't select all the correct answers

10. Which of the following are true about Adam? **1 / 1 point**

- Adam can only be used with batch gradient descent and not with mini-batch gradient descent.
- Adam automatically tunes the hyperparameter α.
- Adam combines the advantages of RMSProp and momentum.
- The most important hyperparameter on Adam is ϵ and should be carefully tuned.

Expand

Correct
True. Precisely Adam combines the features of RMSProp and momentum that is why we use two-parameter $$\beta_1$$ and $$\beta_2$$, besides $$\epsilon$$.