

Back-to-School savings now on! Purchase before September 30th, and get 25% off Socrative Pro. 

Remind me later

Enter coupon code "BTS20" at checkout. [Learn More](#)

CONTI6128

Save and Exit

## CAS Advanced - Day 5



Align Quiz to Standard

Enable Sharing  
SOC-47865305

1. Is it okay to initialize all the weights to the same value as long as that value is selected randomly using He initialization ?



- i** No, all the weights should be sampled independently. They should not all have the same initial value. One important goal of sampling weights randomly is to break symmetries. If all the weights have the same initial value, even if that value is not zero, then symmetry is not broken and backpropagation will be unable to break it. This means that all the neurons in any given layer will always have the same weights.



2. Is it okay to initialize the bias terms to 0 ?

- i** It is perfectly fine to initialize the bias terms to zero. Some people like to initialize them just like weights, and that is okay too. It does not make much difference.



### 3. Name three advantages of the ELU activation function over ReLU.

- i** Advantages are : First, it can take on negative values, so the average output of the neurons in any given layer is typically closer to 0 than when using the ReLU activation function. This helps alleviate the vanishing gradients problem. Second, it always has a nonzero derivative, which avoids the dying units issue that can affect ReLU units. Lastly, it is smooth everywhere, whereas the ReLU's slope abruptly jumps from 0 to 1 at  $z=0$ . Such an abrupt change can slow down gradient descent because it will bounce around  $z=0$ .



### 4. In which cases would you want to use each of the following activation functions: ELU, leaky ReLU, ReLU, tanh, logistic and softmax ?

- i** The ELU activation function is a good default. If you need the NN to be as fast as possible, you can use one of the leaky ReLU variants instead. The simplicity of the ReLU activation function makes it many people's preferred option, despite the fact that they are generally outperformed by the ELU and leaky ReLU. The tanh can be useful in the output layer if you need to output a number between -1 and 1, but also useful in the output layer when you need to estimate a probability, but is also rarely used in hidden layers. Finally, the softmax activation function is useful in the output layer to output probabilities for mutually exclusive classes, but other than that it is rarely used in hidden layers.



### 5. What may happen if you set the momentum hyperparameter too close to 1 (e.g. 0.99999) ?

- i** The algorithm will likely pick up a lot of speed, hopefully roughly toward the global minimum, but then it will shoot right past the minimum due to its momentum. Then it will slow down and come



minimum due to its momentum. Then, it will slow down and come back, accelerate again, overshoot again, and so on. It may oscillate this way many times before converging, so overall it will



take much longer to converge than with a smaller momentum value.

---

**6.** Name two ways you can produce a sparse model

- i** One way to produce a sparse model (with most weights equal to zero) is to train the model normally, then zero out tiny weights. For more sparsity, you can apply L1 regularization during training, which pushes the optimizer toward sparsity.



---

**7.** Does dropout slow down training ? Does it slow down inference ?

- i** Yes, dropout does slow down training, in general roughly by a factor of two. However, it has no impact on inference since it is only turned on during training.



---

Add a Question

**Multiple Choice**

**True / False**

**Short Answer**

Socrative  Get **PRO!** [Learn More](#)