Your grade: 100%

Your latest: **100%** • Your highest: **100%** • To pass you need at least 80%. We keep your highest score.

Next item →

1. What does the backward() function in PyTorch do? 1 / 1 point

- (•) Computes the gradient of a tensor
- ( ) Performs matrix multiplication
- ( ) Saves the computational graph
- ( ) Initializes model parameters

Correct
Correct! The backward() function is used to compute the gradient of a tensor, which is crucial for backpropagation and training neural networks.

2. What is the role of weights in a neural network during the training process? 1 / 1 point

- (•) Weights are adjusted to minimize the loss function, helping the model make more accurate predictions.
- ( ) Weights are only used during forward passes and not during backward passes.
- ( ) Weights determine the structure of the neural network and do not change once the network is defined.
- ( ) Weights remain constant and do not change during the training process.

Correct
Correct! During training, weights are updated to minimize the loss function and improve the model's predictions.

3. What is the role of the torch.save function in PyTorch? 1 / 1 point

- ( ) To train the model
- (•) To save the state dictionary of a model for future use
- ( ) To define the model architecture
- ( ) To initialize model parameters

Correct
Correct! The torch.save function is used to save the state dictionary of a model so it can be loaded and used later.

4. What is the primary purpose of partitioning data into batches during model training? 1 / 1 point

- (•) To efficiently use computational resources and stabilize gradient updates
- ( ) To ensure different data distributions are used in training
- ( ) To increase the complexity of the model
- ( ) To simplify the implementation of the model

Correct
Correct! Partitioning data into batches helps efficiently use computational resources and stabilizes gradient updates.

5. Which component in a PyTorch model is responsible for updating the model parameters based on the calculated gradients? 1 / 1 point

- ( ) Activation function
- ( ) Loss function
- ( ) DataLoader
- (•) Optimizer

Correct
Correct! The optimizer updates the model parameters based on the calculated gradients and the learning rate.