

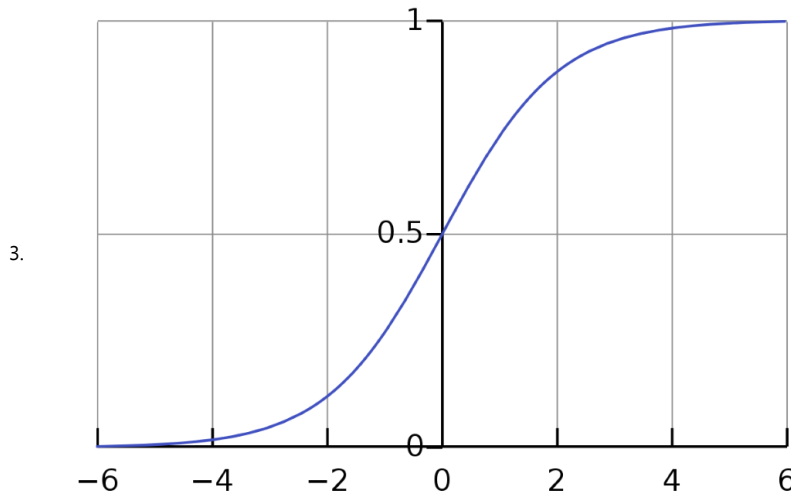
Fast AI chapter 4 questionnaire

01 July 2021 14:43

1. Why can't we use accuracy as a loss function?

Answer: the slight change in the parameters doesn't change the accuracy it will only change when there is big change in for using loss function we measure the probability and if it's right we add $1 - \text{proba}$ and if it's wrong we add proba to the loss of the func

2. Draw the sigmoid function. What is special about its shape?



Answer sigmoid = $1 / (1 + e^{-x})$

the special functionality about sigmoid function is it converges / converts the values from any range to range (0 to 1)

3. What is the difference between a loss function and a metric?

Answer : the loss function even measures the small change in the parameters while the metric just dependent on the result of the model

4. What is the function to calculate new weights using a learning rate?

Answer : optimizer

we use `loss.backward()` and `params.grad()` for calculating the weights

5. What does the DataLoader class do?

Answer : DataLoader helps fast ai to load the data

6. Write pseudocode showing the basic steps taken in each epoch for SGD.

Answer 1. declare learning rate

2. Calculate the grads

3. Making the grads 0

7. Create a function that, if passed two arguments [1,2,3,4] and 'abcd' , returns [(1, 'a'), (2, 'b'), (3, 'c'), (4, 'd')] . What is special about that output data structure?

Answer: `list(zip(x,y))` we can pass it to the dataloaders which will help us to train model and it's x is train data and y is the label

8. What does view do in PyTorch?

Answer : view does the reshaping of pixel images (since they don't use your memory they are memory efficient

9. What are the bias parameters in a neural network? Why do we need them?

Answer : bias parameter helps us when the all the pixels are zero so it doesn't allow neural network to get gradient as 0

10. What does the @ operator do in Python?

Answer : @ opearator used to do the matrix multiplication with python

11. What does the backward method do?

Answer : `loss.backward()` : computes the gradient of the loss

12. Why do we have to zero the gradients?

Answer cause every time we calculate the grads we have to zero it if we didn't do it then gradients get added after each epoch

13. What information do we have to pass to Learner ?

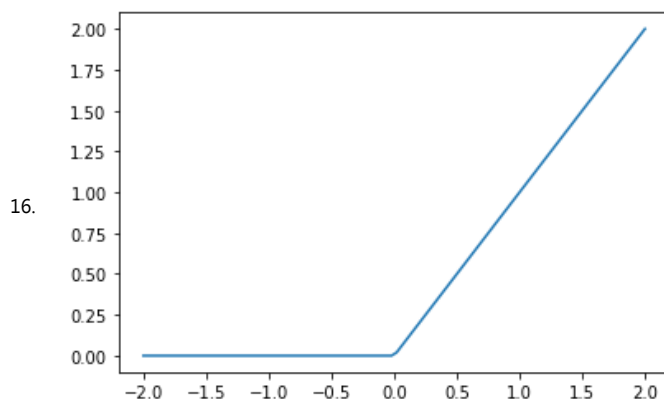
Answer : dataloader , basemodel/ architecture , optimizer , loss and metric

14. Show Python or pseudocode for the basic steps of a training loop.

Answer: `get_predictions()` ----> `calculateloss ()` ----> `calculate gradients ()` ----> `update the params ()`-----> repeat with get predictions (loop)

15. What is ReLU? Draw a plot of it for values from -2 to +2 .

Answer: ReLU it's the rectified linear unit , for negative values the function gives output as 0 if they are more than 0 then it will give output as value

**17. What is an activation function?**

Relu , sigmoid , etch are the activation functions which decides the output format which our outputs gonna be so relu is having no negative values , while the sigmoid is having all probabilities in between 0 and 1

18. What's the difference between F.relu and nn.ReLU ?

`F.relu()` takes the tensors as the input parameters and convert to 0 if params are -ve and +ve values if they are positive

`nn.ReLU()` does the same operation as the `F.relu` but it requires us to initilaize the method in the function parameter

19. The universal approximation theorem shows that any function can be approxi-mated as closely as needed using just one nonlinearity. So why do

we normally use more?

With using the non linearity it can help us to discover more patterns in the model which will in turn help us to make our predictions better. We can use a deeper model with less number of parameters, better performance, faster training, and less compute/memory requirements.