Deep Learning & Applied AI

April 29, 2024

Your name:

Your matr. number:

Question 1 (5 points) Suppose you are writing your own GD procedure in pytorch, and reached this part:

```
with torch.no_grad():
        Z = Z - lr * Z.grad
        Z.grad = None
```

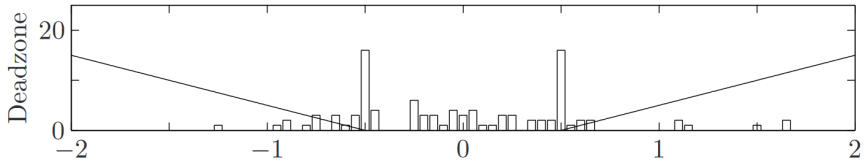
Some student walks by, glosses over your code, and casually says: "this is broken." – without giving an explanation. Alas, the student is right! What could be wrong in your code snippet? How would you fix it?

Question 2 (3 points) You are given a deep neural network taking as input a 3-channel image with 28×28 pixels and a convolutional layer using 3×3 convolutions with 7 output channels. How many weights do you have in the convolutional layer?

Question 3 (7 points) Suppose you want to construct an autoencoder for MLPs; that is, the AE should take as input the weights of an MLP, and reconstruct these weights in the output. The overall idea is that one could then sample the latent space to generate new neural networks.

Can it be done? Think about what could be a major issue with this idea, discuss its feasibility, and whether this would actually be useful or not.

Question 4 (4 points) The "deadzone" penalty function looks like this:



- Describe in words the illustration above, explaining how the penalty acts and how it regularizes the input.
- Write down a possible mathematical expression for this penalty.

Question 5 (4 points) Suppose you trained a classifier using the cross-entropy loss:

- In one case you only kept track of the training *accuracy*, getting 100% at convergence. Does this imply zero loss?
- In the other case, you only kept track of the training *loss* and observed it goes to zero. Does this imply 100% accuracy?

Question 6 (4 points) Consider an encoder-decoder model where each training image, before being fed to the encoder, is intentionally corrupted by setting a random subset of its pixels to zero. Then, the decoder is asked to reconstruct the image as it was *before* the corruption step. Would the resulting trained model be any useful in practice? Motivate your answer.

Question 7 (5 points) Let $\mathcal{S}_n = \{N_\Theta\,,\ |\Theta| = n\}$ be the space of "spherical deep nets", defined as the set of deep neural networks, parametrized by Θ, such that the set of weights of each network identifies a point on the n-dimensional unit sphere. How would you train a network M_Θ to ensure that $M_\Theta \in \mathcal{S}_n$?

Question 8 (8 points) How would you train an AE so that the latent codes produced by the encoder are as sparse as possible? (i) Give explicit mathematical expressions; (ii) explain your reasoning (iii) discuss about the guarantee (or lack thereof) that at test time you get sparse codes, and (iv) explain why having sparse latent codes might be useful.