Test Person 2022-04-25

Email: testperson@example.com
Class: E1a
Teacher: Professor Teacher

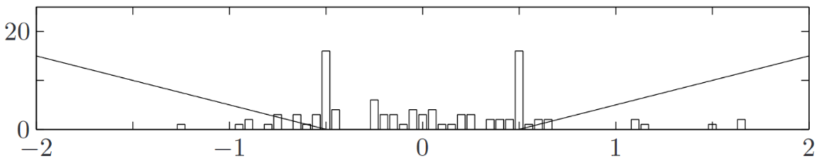
2022-05-03

DLAI (28 April 2022)

Word count: 660

Question 1 (5 points)

Consider the penalty function whose behavior can be described by the illustration below, where the bar plot is a histogram of the network weights after training, and the line plot illustrates the penalty function:



- Based on the figure above, describe its behavior in words including what kind of regularization is applied to the network weights.
- Write a mathematical expression for this penalty function.

Question 2 (7 points)

Your friend came up with an idea for a new training algorithm to replace SGD -- but doesn't have time to explain, and just sends you the code by quickly mentioning: "this is faster!". Here's the code:

```
grads_squared = 0
for _ in num_iterations:
    dw = compute_gradient(x, y)
    grad_squared += dw * dw
    w = w - (lr / np.sqrt(grad_squared)) * dw
```

In an effort to understand your friend's idea, describe what this code does, and in what way it leads to a better training algorithm. Feel free to describe failure cases, if you think there are. Do your best to prove your understanding.

Question 3 (3 points)

How many learnable parameters has a 3-layer MLP with bias (input → hidden1 → hidden2 → output) with 40-20-10-3 neurons, with batch normalization in the first hidden layer, and dropout (p=0.7) in the second hidden layer?

Question 4 (3 points)

Consider the roto-translation operator $A_{\theta,v}\, f(x) = f(R_\theta\, x + v)$, where $R_\theta = \begin{pmatrix} \sin\theta & \cos\theta \\ -\cos\theta & \sin\theta \end{pmatrix}$.

Prove explicitly that it is a linear operator. How would you phrase formally the statement "an operation T is rototranslation-equivariant"?

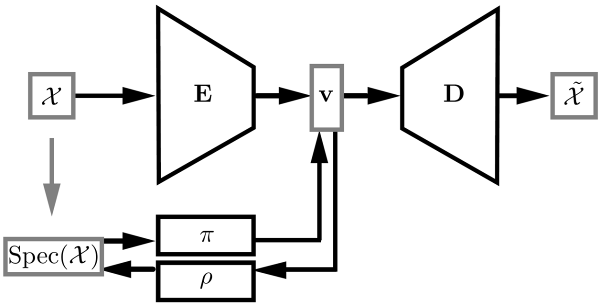
Question 5 (4 points)

Deep neural networks are provably powerful and can approximate arbitrarily complex functions. Yet, one or more of the following architectures are not even able to reasonably approximate the identity map. Which one(s), and why?

1. Deep ReLU without batchnorm
2. Deep ReLU with batchnorm
3. Any VAE
4. Any MLP with dropout and one linear layer at the output

Question 6 (8 points)

Consider the following architecture of a modified autoencoder:



Here, $\mathcal{X}$ is some input (e.g., a sound wave) and $\text{Spec}(\mathcal{X})$is another input computed from the former (e.g. the frequencies λ of the sound wave). The modified autoencoder is trained with the following loss:

$$\ell = \ell_{\mathcal{X}} + \alpha \ell_{\lambda}\,, \quad \text{with}$$
$$\ell_{\mathcal{X}} = \frac{1}{n} \| D(E(\mathbf{X})) - \mathbf{X} \|_F^2$$
$$\ell_{\lambda} = \frac{1}{k} (\| \pi(\boldsymbol{\lambda}) - E(\mathbf{X}) \|_2^2 + \| \rho(E(\mathbf{X})) - \boldsymbol{\lambda} \|_2^2)$$

What is the loss trying to enforce? Describe with the highest level of detail that you can afford, and provide at least *two* examples of usage of this model after it's been trained.

Question 7 (5 points)

You are training an object detector that sees supervised images containing one or more objects with associated class (just a scalar $y \in [0, .., C-1]$, where C is the number of classes) and bounding box (a vector $(x, y, w, h)$where x, y denote the lower left border and w, h denote the width and height of the box respectively). At inference time, your model should be able to detect objects from the classes seen during training and correctly guess their class and bounding box. After observing weak performances, you realize that the model is often tricked by features in the background. What could you do to improve the performance of your model?

Question 8 (5 points)

A particularly enterprising friend of yours founded a startup dealing with vocal assistants. He claims his strenght lies in the data, as he has access to a vast amount of human-annotated user queries that are updated month by month. He decided that he would train his query understanding models iteratively at the start of each month, each time starting from the weights of the previous month. Given the cost of the computing infrastructure, he also decided that each time the model would only be fine-tuned over the new set of queries, as it has already been trained over the older ones during previous iterations.

To his surprise, the model performance started deteriorating after a few months. After a preliminary investigation, they realized that most of the errors resided in queries that the model should have seen in the older months. Given your notorious expertise in deep learning, he turned to you to ask your opinion on this matter. Beware, his earnings directly depend on this stuff, so your analysis will have to be convincing and detailed. Assuming that you managed to convince him, how would you then mitigate the problem?