Backpropagation

Puteaux, Fall/Winter 2020-2021

```
################################
##                            ##
##  Deep Learning in Python  ##
##                            ##
################################
```

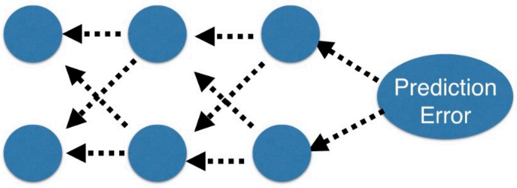
§1 Introduction to Deep Learning in Python

§1.2 Optimizing a neural network with backward propagation

§1.2.3 Backpropagation

1. What is the aim of backpropagation?

Backpropagation aims to allow gradient descent to update all weights in the neural network by getting gradients for all weights. This idea comes from the chain rule of calculus. It is important to understand the process, but generally, a library is available that implements this.



2. What is the process of backpropagation?

- Try to estimate the slope of the loss function with respect to each weight.
- Do forward propagation to calculate predictions and errors.
- Go back one layer at a time.
- Gradients for weight is the product of:
    1. node value feeding into that weight
    2. slope of the loss function with respect to node it feeds into
    3. slope of activation function at the node it feeds into
- It is also necessary to keep track of the slopes of the loss function with respect to node values.
- The slope of node values is the sum of the slopes for all weights that come out of them.

3. Practice question for the relationship between the forward and backward propagation:

- If gone through 4 iterations of calculating slopes (using backward propagation) and then updated weights, how many times must be done forward propagation?

  ☐ 0.

  ☐ 1.

  ☒ 4.

  ☐ 8.

4. Practice question for thinking about backward propagation:

- If the predictions were all exactly right, and the errors were all exactly 0, the slope of the loss function with respect to the predictions would also be 0. In that circumstance, which of the following statements would be correct?

  ☒ The updates to all weights in the network would also be 0.

  ☐ The updates to all weights in the network would be dependent on the activation functions.

  ☐ The updates to all weights in the network would be proportional to values from the input data.