Chapter 1

Deep Learning

Deep learning is a subfield of machine learning that focuses on algorithms inspired by the structure and function of the brain called artificial neural networks. While you may have encountered simple, shallow neural networks, deep learning involves neural networks with many layers, hence they are often referred to as ”deep” neural networks.

1.1 Deep Learning

Deep learning models learn to represent data by training on a large number of examples. Unlike shallow neural networks that have one or two layers of hidden nodes, deep networks can have tens or even hundreds of layers of hidden nodes. Each layer in these networks performs a nonlinear transformation of its inputs and is trained to extract increasingly abstract features with each additional layer.

1.2 Chain Rule

In order to understand how these networks are trained, we need to revisit a fundamental concept from calculus, the chain rule. The chain rule is used for differentiating compositions of functions. It essentially says that the derivative of a composed function is the product of the derivatives of the composed functions.

Suppose we have a function $y = f(g(x))$, then the derivative of y with respect to x is:

$$\frac{dy}{dx} = f'(g(x)) \cdot g'(x) \tag{1.1}$$

This rule becomes indispensable when calculating the gradient of the loss function in a deep learning model with respect to the model parameters.

1.3 Backpropagation

Backpropagation is the method used to train deep learning models by calculating the gradient of the loss function with respect to each weight in the network. The name "backpropagation" comes from the fact that the calculation of the gradient proceeds backwards through the network, with the gradient of the final layer of weights being calculated first and the gradient of the first layer of weights being calculated last.

Mathematically, backpropagation uses the chain rule to efficiently compute these gradients. Starting from the final layer, the chain rule is repeatedly applied to propagate the gradient backwards through the network, storing intermediate results as it goes along. Once the gradient has been calculated, the weights are updated using a gradient descent step.

Appendix A

Answers to Exercises

INDEX