Vanilla Neural Networks

Shamel Bhimani

August 2025

Contents

1 Precursors to Backpropagation

test

1.1 Chain Rule

1.1.1 Core Principle of the Chain Rule

The chain rule is a formula for finding the *derivative* of a **composite function**, that is, a function that is formed by the composition of two or more other functions. It is an indispensable tool in science, engineering, and statistics, particularly for optimization problems where functions are dependent on a chain of intermediate variables. In the context of neural networks, the backpropagation algorithm is, at its core, a highly efficient and systematic application of the chain rule.

1.1.2 The Single-Variable Chain Rule

Let y be a function of u, denoted as $y = f(u)$, and let u in turn be a function of x, denoted as $u = g(x)$. The composition of these two functions forms a new function, $y = F(x) = f(g(x))$.

The **Chain Rule Theorem** states that if both f and g are differentiable functions, then the derivative of the composition function $F(x)$ with respect to x is the product of the derivative of the outer function f with respect to its input u, and the derivative of the inner function g with respect to its input x.

Theorem 1.1 (**Chain Rule**). *If $y. = f(u)$ and $u = g(x)$ are differentiable functions, then the derivative of y with respect to x is given by:*

$$\frac{dy}{dx} = \frac{dy}{du} \cdot \frac{du}{dx}$$

An equivalent notation, often used for simplicity, is the prime notation:

$$F'(x) = f'(g(x)) \cdot g'(x)$$

Example 1.1.1.
Let $y = (x^2+1)^3$. We can identify this as a composite function. Let the inner function be $u = g(x) = x^2+1$. Let the outer function be $y = f(u) = u^3$.

First, we find the individual derivatives:

$$\frac{dy}{du} = \frac{d}{du}(u^3) = 3u^2$$
$$\frac{du}{dx} = \frac{d}{dx}(x^2+1) = 2x$$

Now, we apply chain rule:

$$\frac{dy}{dx} = \frac{dy}{du} \cdot \frac{du}{dx} = (3u^2) \cdot (2x)$$

Finally, we substitute the expression for u back into the result:

$$\frac{dy}{dx} = 3(x^2+1)^2 \cdot (2x) = 6x(x^2+1)^2$$

1.1.3 The Multi-Variable Chain Rule

The concept of the chai rule extends to functions of multiple variables using partial derivatives. This form is particularly relevant in fields like optimization and machine learning, where a function's value depends on a multitude of intermediate parameters.

Consider a function z that depends on two intermediate variables x and y, such that $z = f(x, y)$. In turn, both x and y are functions of a third variable, t, such that $x = g(t)$ and $y = h(t)$. To find the total rate of change of z with respect to t, we must sum the contributions from each path of dependency.

Theorem 1.2 (Multi-Variable Chain Rule)**.** *If $z = f(x, y)$ is a differentiable function of x and y, and $x = g(t)$ and $y = h(t)$ are differentiable functions of t, then the total derivative of z with respect to t is:*

$$\frac{\mathrm{d}z}{\mathrm{d}t} = \frac{\partial z}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}t} + \frac{\partial z}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}t}$$

The partial derivative $(\frac{\partial z}{\partial x'} \frac{\partial z}{\partial y})$ account for how z changes with respect to its immediate inputs, while the derivatives $(\frac{\mathrm{d}x}{\mathrm{d}t'} \frac{\mathrm{d}y}{\mathrm{d}t})$ account for how those inputs change with respect to the ultimate variable, t. The sum captures the total effect.

Generalization: This rules extends to any number of intermediate variables. If $z = f(x_1, x_2, \ldots, x_n)$, and each x_i is a function of t, then:

$$\frac{\mathrm{d}t}{\mathrm{d}z} = \sum_{i=1}^{n} \frac{\partial z}{\partial x_i}\frac{\mathrm{d}x_i}{\mathrm{d}t}$$

Example 1.2.1.
Let $z = x^2y^3$, where $x = \sin t$ and $y = \cos(t)$. We want to find $\frac{\mathrm{d}z}{\mathrm{d}t}$. First, we compute the partial derivatives of z with respect to its inputs x and y:

$$\frac{\partial z}{\partial x} = \frac{\mathrm{d}}{\mathrm{d}x}(x^2y^3) = 2xy^3 \tag{1}$$

$$\frac{\partial z}{\partial y} = \frac{\mathrm{d}}{\mathrm{d}y}(x^2y^3) = 3x^2y^2 \tag{2}$$

Next, we compute the derivatives of the intermediate variables with respect to t:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}(\sin t) = \cos t \tag{3}$$

$$\frac{\mathrm{d}y}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}(\sin t) = -\sin t \tag{4}$$

Finally, we apply the multi-variable chain rule formula:

$$\frac{\mathrm{d}z}{\mathrm{d}t} = \frac{\partial z}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}t} + \frac{\partial z}{\partial y}\frac{\mathrm{d}y}{\mathrm{d}t} \tag{5}$$

$$\frac{\mathrm{d}z}{\mathrm{d}t} = (2xy^3)(\cos t) + (3x^2y^2)(-\sin t) \tag{6}$$

$$\tag{7}$$

To obtain the final expression solely in terms of t. we substitute $x = \sin t$ and $y = \cos t$ back into the equation:

$$\frac{\mathrm{d}z}{\mathrm{d}t} = 2\sin t\cos^4 t - 3\sin^3 t\cos^2 t) \tag{8}$$

1.1.4 A Short Note on Application

The architecture of a neural network, from the inputs to the final loss function, is a complex chain of composite functions. Backpropagation is a sophisticated algorithm that applies the multi-variable chain rule to compute the gradient of the loss with respect to every weight and bias in the network. It systematically calculates the partial derivatives of each function in the chain, propagating the error signal backward from the output layer to the input layer to determine how each indvidual parameter should be adjusted to minimize the loss.