Calculus for Machine Learning

Kevin Fu

November 2019

1 Introduction

Usually, when people say "machine learning," they're thinking of neural networks. Though decision trees, SVMs, and KNNs are all forms of machine learning, more complex ML systems that classify images or translate languages are based on neural networks. To properly understand neural networks, we'll spend three lectures on the topic, and have a problem set and Kaggle competition. This material is difficult if you don't have the requisite multivariable calculus knowledge (and is difficult even if you do).

For that reason, this lecture is designed to allow you to grapple with the mathematics behind neural networks before you have to grapple with the mechanics of one. In a few weeks, when you're creating your own neural networks from scratch for the Kaggle competition, you'll be glad you grappled now.

2 Derivatives

The derivative of a function gives us the rate of change at any point on that function. To understand how that's defined mathematically, it's helpful to look at how the derivative relates to a familiar concept: slope.

2.1 Definition

Slope is simply rise over run. For a function like

$$f(x) = 2x$$

we know the slope is 2, because the equation is in the form $y = mx + b$ ($b = 0$). We can check this by plugging in points, like $x_0 = 1$ and $x_1 = 2$, and computing the ratio of the change in rise, or Δy, to the change in run, or Δx, like this:

$$slope = \frac{rise}{run} = \frac{\Delta y}{\Delta x} = \frac{y_1 - y_0}{x_1 - x_0} = \frac{f(2) - f(1)}{2 - 1} = \frac{4 - 2}{2 - 1} = 2$$

This works for any points x_0 and x_1. But what if our function is non-linear? For example, the slope of a parabola, or $f(x) = x^2$, changes depending on what

x is chosen. It will be negative for negative x values, positive for positive x's, and zero at $x = 0$.

We can still approximate the slope between two points the same way we did for the linear function. The slope at $x = 1$, if we use a difference in x of 1 again, is approximately:

$$\frac{\Delta y}{\Delta x} \approx \frac{f(2) - f(1)}{2 - 1} = \frac{4 - 1}{2 - 1} = 3$$

If we use smaller differences of x, we'll get different results. Again starting at $x = 1$, with the numerator rewritten to use Δx for simplicity, we get this formula:

$$\frac{\Delta y}{\Delta x} \approx \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x}$$

which we can apply like this:

Δx	$\frac{\Delta y}{\Delta x}$	**Slope**
1	$\frac{2^2-1^2}{2-1}$	3
0.5	$\frac{1.5^2-1^2}{2-1}$	2.5
0.25	$\frac{1.25^2-1^2}{2-1}$	2.25
0.1	$\frac{1.1^2-1^2}{2-1}$	2.1
...	...	...

Graphically, this can be seen as taking secant lines, only with smaller and smaller differences between the two points forming our line. (See Figure 1.)

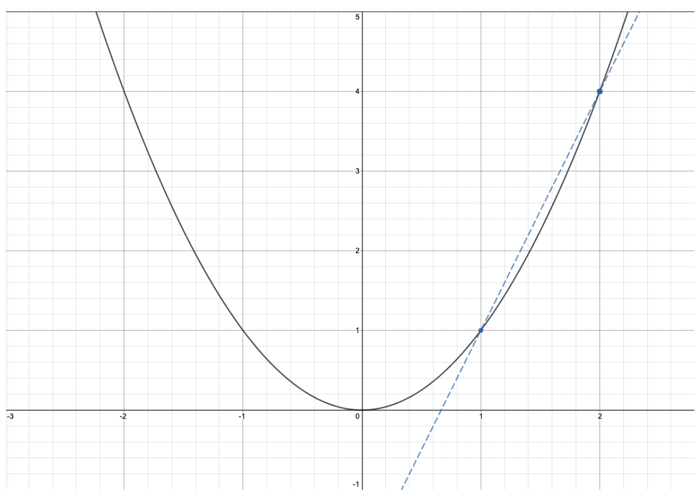

Figure 1: A secant of x^2.

What if we wanted the slope of the line tangent to the curve? (See Figure 2.) A tangent line, by definition, is the straight line that touches the curve at only one point, so we can't use the same Δx approximations we've been using, since those require two points to create a Δx from. You might be able to guess the answer is 2, given the table, but is there a more mathematically rigorous way to prove it?

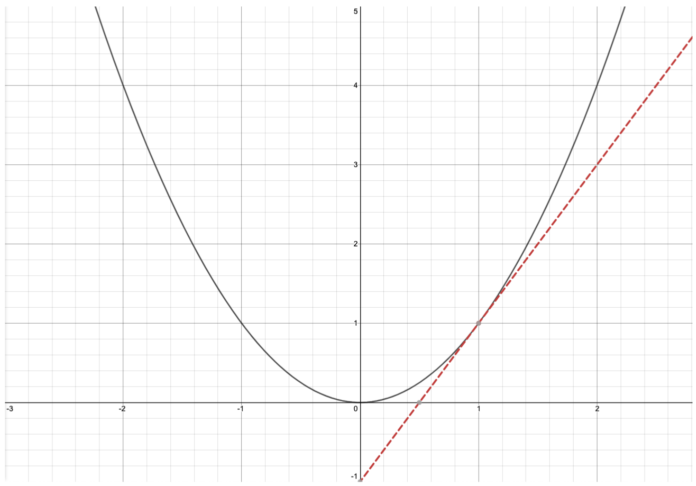

Figure 2: The line tangent to x^2 at (1, 1).

Yes, using limits, we can. A tangent line is a secant line where $\Delta x = 0$, but if we plugged that into our formula, we'd get this:

$$\frac{\Delta y}{\Delta x} \approx \frac{f(x_0 + 0) - f(x_0)}{0} = \frac{0}{0}$$

which is a nonreal answer. So instead, we take the limit of Δx as it approaches 0:

$$\frac{\Delta y}{\Delta x} = \lim_{\Delta x \to 0} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} \tag{1}$$

In other words, we're taking the slope of the secant line, but our secant line is formed from the smallest possible distance between two points. This allows us to approximate, as closely as mathematically possible, the slope of the line tangent to any point on the function $f(x) = x^2$.

The limit expression in (1) is actually the formal definition of the derivative. To notate that, we can write:

$$\frac{dy}{dx} = \frac{df}{dx} = \frac{d}{dx} f(x) = f'(x)$$

which are all understood to mean the derivative of f with respect to x. δ is replaced with d to show that we're taking the limit as δ approaches 0.

This tangent line is not just arbitrary; it has real-world applications. For instance, the tangent line on a distance vs. time graph for a car would give the car's velocity. A tangent line on a car's velocity vs. time graph would give the car's acceleration.

2.2 Derivative Rules

One function you'll want to figure out how to differentiate (take the derivative of) is the sigmoid function, because it's commonly used as the activation function in neural networks, which you'll learn next week.

$$S(x) = \frac{1}{1+e^{-x}} \tag{2}$$

However, simplifying the limit expression for the sigmoid function is annoying. Luckily, we have plenty of derivative rules that allow us to skip the limit expression altogether.

The derivative of any constant is 0. This makes sense if you think about the slope of the function $f(x) = c$—it will just be a flat horizontal line, so the slope at any point is 0. Similarly, the derivative of a linear function is just the coefficient before x. Think of $y = mx + b$. The m coefficient is the slope of the line at any point.

The derivative of the sum of two functions is just the sum of their derivatives. Mathematically, that's:

$$\frac{d}{dx}(f(x) + g(x)) = \frac{d}{dx}f(x) + \frac{d}{dx}g(x) \tag{3}$$

The power rule, used to differentiate functions with a constant exponent n, is:

$$\frac{d}{dx}x^n = nx^{n-1} \tag{4}$$

For example, the slope of the tangent line on any point of the parabola, or the closest approximation for rate of change at a point, is easily calculable using the power rule. If $f(x) = x^2$, then:

$$f'(x) = \frac{d}{dx}x^2 = 2x^{2-1} = 2x$$

So the rate of change at x = 1 is simply:

$$f'(1) = 2(1) = 2$$

Also, exponential functions, or functions with an variable in the exponent, can be differentiated like so:

$$\frac{d}{dx}a^x = a^x ln(a)$$

Since the only exponential part of the sigmoid function is e^x, just remember that:

$$\frac{d}{dx}e^x = e^x ln(e) = e^x \tag{5}$$

The last rule we'll need to differentiate the sigmoid function is the chain rule, which is:

$$\frac{d}{dx}f(g(x)) = f'(g(x))g'(x) \tag{6}$$

In words: if you have a function that contains another function, you need to take the derivative of the outer function, plug in the inner function, then multiply everything by the derivative of the inner function. It's a little like recursive calls, because you have to go one layer at a time. We have to take this extra step because the variable x affects both the inner and outer functions, so a tiny change dx will also affect both, which our derivative must show.

With these three rules, you should be able to calculate the derivative of the sigmoid function in (2). Try it yourself, then check the step-by-step solution below. Hint:

$$\frac{1}{1+e^{-x}} = (1+e^{-x})^{-1}$$

2.3 Solution

Since the sigmoid function has an e^{-x} nested inside it, we must apply the chain rule in (6). Thinking of the inner function $g(x)$ as $1+e^{-x}$, we get:

$$\frac{d}{dx}((1+e^{-x})^{-1}) = \frac{d}{dx}((g(x))^{-1})\frac{d}{dx}(g(x))$$

The left half is solvable with the power rule (4). The right half is a sum of two functions, so we'll apply (3). (The derivative of a constant is 0.)

$$= -1(1+e^{-x})^{-2}(\frac{d}{dx}1 + \frac{d}{dx}(e^{-x})) = -1(1+e^{-x})^{-2}\frac{d}{dx}(e^{-x})$$

The derivative on the right is also a case of the chain rule, where the "inner" function is $-x$ and the "outer" function is e^x. (The derivative of a linear function is the coefficient.)

$$= -1(1+e^{-x})^{-2}\frac{d}{dx}(e^{g(x)})\frac{d}{dx}(g(x)) = -1(1+e^{-x})^{-2}(e^{-x})(-1)$$

Simplifying, we get:

$$S'(x) = \frac{e^{-x}}{(1+e^{-x})^2} \tag{7}$$

If this was your solution, congrats! You've just successfully applied several new calculus rules. See Figure 3 for what that looks like visually.

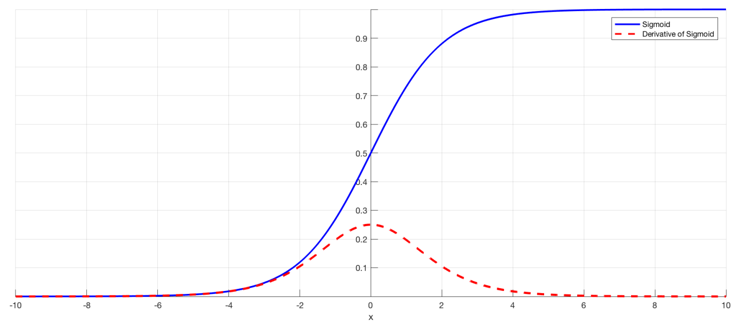

Figure 3: The sigmoid function graphed with its derivative.

But we can actually further simplify (7). With the equation for the sigmoid function $S(x)$ from (2) in mind, let's rewrite our answer to be:

$$(\frac{1}{1+e^{-x}})\frac{e^{-x}}{1+e^{-x}} = S(x)\frac{e^{-x}}{1+e^{-x}}$$

We can further split the right half like so:

$$= S(x)\frac{(1+e^{-x})-1}{1+e^{-x}} = S(x)(\frac{1+e^{-x}}{1+e^{-x}} - \frac{1}{1+e^{-x}})$$

Which gives us the most useful expression for the derivative of the sigmoid function:

$$S'(x) = S(x)(1-S(x)) \tag{8}$$

This means that to take the derivative of the sigmoid function with our custom neural networks, we only need to program in the sigmoid function itself. That's neat.

3 Gradients

The other place you'll require an understanding of calculus is in gradient descent, or how neural networks learn. In essence, neural nets are chains of formulas that input data passes through to get to an output prediction. The total error, or cost, of the neural network's prediction can be used to determine how we should change the formulas leading up to it.

However, since these formulas all affect one another, dozens of variables end up affecting the overall cost function. So the change in variables can't be expressed with a derivative, which applies to the slope of a 2D graph, but requires the "slope" of an n-dimensional graph. This is best expressed as a vector, which we call the gradient, and write as ∇f.

Luckily, when you actually implement a neural network, you won't have to find the gradient manually, thanks to matrix algebra tricks. Instead, we'll gain a conceptual understanding of the gradient today with a simple 3D example.

The equation $z = x^2 + y^2$ looks like a parabola rotated around the z-axis. (See Figure 4.) Visualize a ball on this surface. It would naturally roll down to the origin, right? With the gradient, we can actually find the direction of travel of such a ball at any point. This is similar to how with the derivative of a parabola, we can find the direction of travel of a ball placed inside the parabola's curve at any point.

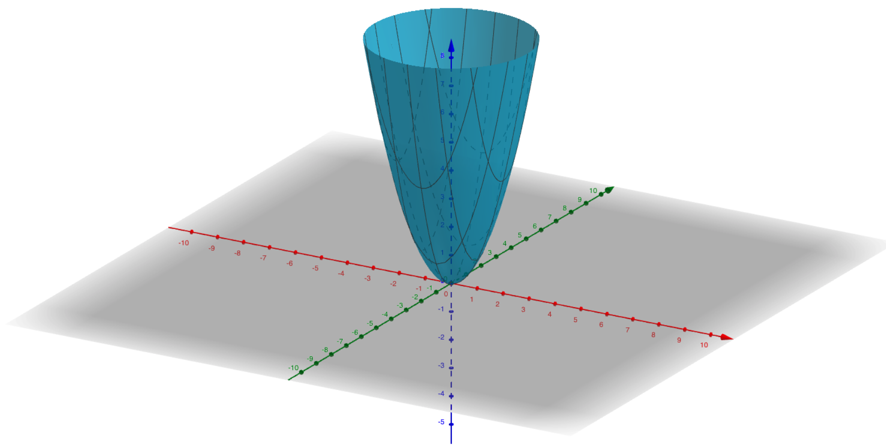

Figure 4: $z = x^2 + y^2$

And finding gradients doesn't require a new derivative rule. Instead, to find a gradient, take a derivative with respect to every variable in the function, which means we apply standard rules to one variable and treat all other variables as constants. For the function $z = x^2 + y^2$, we'll take two derivatives, one with respect to x, and another with respect to y. These derivatives are called partials and can be notated as:

$$\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}$$

Our final gradient will simply be the vector of these two partials:

$$\nabla z = \langle \frac{\partial z}{\partial x}, \frac{\partial z}{\partial y} \rangle$$

In this example, the partials are simple, since the derivative of the other variable's terms is zero for both partials (since we treat the other variable as a constant). So the final gradient is:

$$\nabla z = \langle \frac{\partial z}{\partial x}, \frac{\partial z}{\partial y} \rangle = \langle 2x, 2y \rangle$$

At the point (1,1), we'd get the vector $\langle 2, 2\rangle$. But wait, that vector points in the opposite direction from where the ball would roll to. In reality, the gradient points in the direction of steepest ascent. Taking the negative of the gradient will give the direction of steepest descent:

$$-\nabla z = -\left\langle 2x, 2y \right\rangle = \left\langle -2x, -2y \right\rangle$$

Which at the point (1,1), would give us a direction of $\left\langle -2, -2 \right\rangle$, matching the direction of travel of our imaginary ball as it goes to the origin. This formula works at any point, too; try (1,0) or (0,0).

4 References

4.1 Derivative Help

- Khan Academy: https://www.khanacademy.org/math/calculus-1/cs1-derivatives-definition-and-basic-rules
- 3b1b: https://www.youtube.com/playlist?list=PLZHQObOWTQDMsr9K-rj53DwVRMYO3t5Yr

4.2 Images

- Sigmoid: https://towardsdatascience.com/derivative-of-the-sigmoid-function-536880cf918e
- 2D graphs: https://www.desmos.com/calculator
- 3D graph: https://www.geogebra.org/3d?lang=en