Calculus for Data Science & Machine Learning

Lazy Programmer
https://lazyprogrammer.me

Version 1.0 — March 3, 2023

1 Limits

The notation for limits is as follows:

$$\lim_{x \to a} f(x) = L \tag{1}$$

It means that the limit of the function $f(x)$ as x approaches a is L.

We note that it is not always true that $L = f(a)$ although in some cases, it is.

Sometimes, more difficult limits require rigorous proofs. For intuition, we may use plotting and plugging in x values close to a in order to determine the limit.

Limits involving infinities and asymptotes: Sometimes we want to know the limit as $x \to \infty$, instead of the limit as x approaches some number a (note: ∞ is not a number). The actual limit itself may be infinite.

Indeterminate forms: If, when we plug in $f(a)$ directly and we arrive at an answer like $0/0$ or ∞/∞, this is called an indeterminate form. Further algebraic manipulation or proofs are needed to evaluate these limits.

2 Derivatives

In this section, we use the limit as a tool to define the derivative. The derivative is a function that tells us the slope of a given function at each point.

$$f'(a) = \lim_{x \to a} \frac{f(x) - f(a)}{x - a} \tag{2}$$

We learned about another, equivalent form:

$$f'(x) = \lim_{h \to 0} \frac{f(x + h) - f(x)}{h} \tag{3}$$

We noted that these forms can also be used as approximations for derivative checking, e.g. by plugging in a value for h that is very close to 0. i.e.

$$f'(x) \approx \frac{f(x + h) - f(x)}{h} \tag{4}$$

We learned that one can also use the 2-sided limit for derivative checking for improved accuracy:

$$f'(x) \approx \frac{f(x + h) - f(x - h)}{2h} \tag{5}$$

We learned about alternative notations for the derivative:

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

And we learned about higher order derivatives:

$$f''(x) = \frac{d^2y}{dx^2} \tag{7}$$

And in general:

$$f^{(n)}(x) = \frac{d^ny}{dx^n} \tag{8}$$

In this section, we derived a few derivatives using "first principles", but this is not easy. As a shortcut, it is better to use derivative rules (also derived using "first principles").

3 Derivative Rules

Power rule (for any real r):

$$\frac{d}{dx}x^r = rx^{r-1} \tag{9}$$

Constant multiple rule:

$$\frac{d}{dx}cf(x) = c\frac{d}{dx}f(x) \tag{10}$$

Addition rule:

$$\frac{d}{dx}[f(x) + g(x)] = \frac{d}{dx}f(x) + \frac{d}{dx}g(x) \tag{11}$$

Exponent rule:

$$\frac{d}{dx}e^x = e^x \tag{12}$$

Note: $e = 2.718281828459045...$ is a special constant in mathematics because of this property. In general:

$$\frac{d}{dx}b^x = \ln(b)b^x \tag{13}$$

Chain rule:

$$\frac{d}{dx}f(g(x)) = f'(g(x))g'(x) \tag{14}$$

Product rule:

$$\frac{d}{dx}[f(x)g(x)] = f'(x)g(x) + f(x)g'(x) \tag{15}$$

Quotient rule:

$$\frac{d}{dx}\frac{f(x)}{g(x)} = \frac{f'(x)g(x) - f(x)g'(x)}{[g(x)]^2} \tag{16}$$

Logarithm rule:

$$\frac{d}{dx}\ln(x) = \frac{1}{x} \tag{17}$$

In general:

$$\frac{d}{dx}\log_b(x) = \frac{1}{x\ln b} \tag{18}$$

Handy tricks for complicated derivatives: implicit differentiation and logarithmic differentiation.
Trigonometric rules:

$$\frac{d}{dx}\sin(x) = \cos(x) \tag{19}$$

$$\frac{d}{dx}\cos(x) = -\sin(x) \tag{20}$$

$$\frac{d}{dx}\tan(x) = \sec^2(x) \tag{21}$$

Inverse trigonometric rules:

$$\frac{d}{dx}\sin^{-1}(x) = \frac{1}{\sqrt{1-x^2}} \tag{22}$$

$$\frac{d}{dx}\cos^{-1}(x) = -\frac{1}{\sqrt{1-x^2}} \tag{23}$$

$$\frac{d}{dx}\tan^{-1}(x) = \frac{1}{1+x^2} \tag{24}$$

4 Applications of Differentiation

In this section, we looked at 3 applications of differentiation.

1) Finding the maximum or minimum of a function. To do this, we find the point(s) where $f'(x) = 0$.

In order to determine whether such points are minima or maxima, we apply the first derivative test or second derivative test.

The first derivative test looks at the sign of the derivative to the left and to the right of where $f'(x) = 0$. If we go from + to -, we have a maximum, and if we go from - to +, we have a minimum.

The second derivative test requires us to find $f''(x)$. If $f''(x) > 0$, we have a minimum, and if we have $f''(x) < 0$, we have a maximum. Otherwise, the test is inconclusive (we can have a min, max, or neither).

2) l'Hopital's rule for finding limits with indeterminate forms. Specifically:

$$\lim_{x\to a}\frac{f(x)}{g(x)} = \lim_{x\to a}\frac{f'(x)}{g'(x)} \tag{25}$$

3) Newton's method. Newton's method is an iterative algorithm which can be used to find the zeros of a function or to find stationary points (where $f'(x) = 0$).
For finding zeros:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \tag{26}$$

For finding stationary points:

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)} \tag{27}$$

5 Integration

The first fundamental theorem of calculus states that if the function F is defined by:

$$F(x) = \int_a^x f(t)dt \tag{28}$$

Then:

$$F'(x) = f(x) \tag{29}$$

In other words, integration is the opposite of differentiation.

The second fundamental theorem of calculus states that if we want to find the area under the curve of $f(x)$ from $x = a$ to $x = b$, then it can be computed by using the anti-derivative:

$$\int_a^b f(x)dx = F(b) - F(a) \tag{30}$$

Where $F'(x) = f(x)$.

We learned how to compute definite integrals (areas) and indefinite integrals (finding anti-derivatives).

We learned how to compute improper integrals (integrals that are really limits).

We learned about numerical integration: summing up rectangles or trapezoids to approximate a definite integral.

6 Vector Calculus

Partial differentiation: just like regular differentiation except we differentiate with respect to only a single variable at a time.

$$\frac{\partial f(x,y,z)}{\partial x} = \lim_{h\to 0} \frac{f(x+h,y,z) - f(x,y,z)}{h} \tag{31}$$

The gradient is just the vector of partial derivatives.

$$\nabla f(x,y,z) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}\right) \tag{32}$$

Note: in this course, we aren't really dealing with vectors and matrices (unless necessary). As such, the gradient can be applied to a function of any number of multiple variables, regardless of how those variables happen to be organized (a vector, a matrix, a tensor, etc.). At the end of the day, we're just collecting all possible partial derivatives. However, it usually "makes sense" to organize the gradient to have the same shape as the input variables. So if your input variable is a vector, then your gradient will be a vector of the same shape. If your input variable is a matrix, then your gradient will be a matrix of the shape shape.

The Jacobian is a matrix of partial derivatives that results when we have a vector function of a vector input argument. For example, if we have a function **f(x)** where **f** is m-dimensional and **x** is n-dimensional, then:

$$\mathbf{J} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \tag{33}$$

Since the gradient of a scalar function is a vector function, taking all possible derivatives again would yield a Jacobian. This is a special matrix called the Hessian.

Differentials in multiple dimensions (given $f(x,y)$):

$$df = \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy \tag{34}$$

And the corresponding chain rule (assuming that x and y are functions of t):

$$\frac{df}{dt} = \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt} \tag{35}$$

If x and y are functions of several variables (e.g. s and t), then this can be written as:

$$\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial t} \tag{36}$$

Important fact: the gradient is a vector that tells us the direction of steepest ascent. We demonstrated this by first defining the directional derivative:

$$D_{\mathbf{u}} f = \nabla f \cdot \mathbf{u} \tag{37}$$

Since $\nabla f \cdot \mathbf{u} = \|\nabla f\| \cos\theta$, we should have $\theta = 0$ to maximize this derivative (change in f), hence showing that ∇f is in fact the direction of steepest ascent (greatest change in f).

6.1 Optimization and Lagrange Multipliers

For unconstrained optimization, the process is analogous to the single variable case. Given a function $f(\mathbf{x})$, the point where:

$$\nabla f = 0 \tag{38}$$

is the point where f is either a minimum or a maximum. Typically in machine learning, our objective functions are set up such that we already know if we are looking for a minimum or a maximum, so there's no need to check.

However, if you did want to check, you could use the second derivative test, using the Hessian $\mathbf{H}$. If $\mathbf{H} \succ 0$ (is positive definite), then the point is a minimum. If $\mathbf{H} \prec 0$ (is negative definite), then the point is a maximum. Checking positive and negative definiteness requires knowledge of linear algebra.

For constrained optimization, we considered first a function with a single constraint.

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & g(\mathbf{x}) = 0 \end{aligned} \tag{39}$$

To solve this problem, we introduce a scalar called the Lagrange multiplier λ, and form the Lagrangian:

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) - \lambda g(\mathbf{x}) \tag{40}$$

The solution is found by setting $\nabla_{\mathbf{x},\lambda} L = 0$.

This is equivalent to solving the system of equations:

$$\begin{aligned} \nabla f &= \lambda \nabla g \\ g(\mathbf{x}) &= 0 \end{aligned} \tag{41}$$

When we have multiple constraints, we create a Lagrange multiplier for each constraint, and solve (assuming there are M constraints):

$$\begin{aligned} \nabla f &= \sum_{i=1}^{M} \lambda_i \nabla g_i \\ g_i(\mathbf{x}) &= 0, i = 1, 2, ..., M \end{aligned} \tag{42}$$

7 Taylor Series and Taylor Expansion

Nice, infinitely differentiable functions have Taylor expansions, which are infinite sums of polynomials.

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n \tag{43}$$

This is a Taylor expansion centered at x_0. If we use less than an infinite number of terms, x_0 is where the approximation is most accurate. If $x_0 = 0$, it is also called a Maclaurin series.

In multiple dimensions, the first-order Taylor expansion looks like this (this assumes that $\mathbf{x}$ is a column vector):

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \nabla f^T (\mathbf{x} - \mathbf{x}_0) \tag{44}$$

If we plug in $\mathbf{x}_0 = \mathbf{x}$ and $\mathbf{x} = \mathbf{x} + \Delta\mathbf{x}$, then we get the differential as $\Delta\mathbf{x} \to 0$, since we'd get:

$$f(\mathbf{x} + \Delta\mathbf{x}) \approx f(\mathbf{x}) + \nabla f^T (\mathbf{x} + \Delta\mathbf{x} - \mathbf{x}) \tag{45}$$

Which reduces to:

$$\Delta f \approx \nabla f^T \Delta\mathbf{x} \tag{46}$$

And the second-order Taylor expansion looks like this:

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \nabla f^T(\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T \mathbf{H}(\mathbf{x} - \mathbf{x}_0) \tag{47}$$

Looking ahead: this will be useful when we discuss Newton's method for optimization in multiple dimensions!