Notes on Calculus III

Aaron Pierce

1 Early Chapters

The beginning chapters aren't as noteworthy as the later ones, but they are still valuable. The following is a collection of the most important aspects

1.1 Dot Products

Dot products are a form of vector multiplication.

Definition: The dot product of two vectors $(\vec{A} \cdot \vec{B})$ is the projection of one onto the other times the length of the projection and the vector being projected on.

That's a mouthful, and I like pictures.

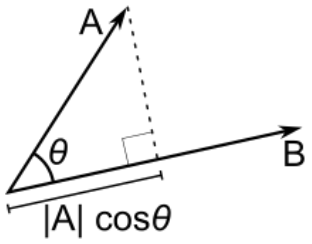

Figure 1: The dot product

Numerically, this projection is $|\vec{A}|cos(\theta)|\vec{B}|$, which corresponds to the component of A that lies on B, times their lengths.

This definition derives some very useful other expressions, such as

$$\cos(\theta) = \frac{\vec{A} \cdot \vec{B}}{|\vec{A}||\vec{B}|} \tag{1}$$

$$comp_{\vec{B}}\vec{A} = \frac{\vec{A} \cdot \vec{B}}{|\vec{B}|} = \vec{A}\cos(\theta) \tag{2}$$

Another way to define the dot product is the sum of the products of the components of the vectors. Another mouthful. The math is a easier to

understand

$$\text{Let}\vec{A} = (A_1, A_2, ..., A_n)$$
$$\text{Let}\vec{B} = (B_1, B_2, ..., B_n)$$
$$\vec{A} \cdot \vec{B} = A_1B_1 + A_2B_2 + ... + A_nB_n$$

Let's call this the algebraic definition, as opposed to the projection or geometric definition from earlier.
This was surprising to me. How is this equivalent to the projection definition from earlier? What helped me was realizing that this definition is equivalent to applying the geometric definition over the components of B.

$$\text{Let}\vec{A} = (A_1, A_2, A_3)$$
$$\text{Let}\vec{B} = (B_1, B_2, B_3)$$
$$\vec{B} = (B_1, 0, 0) + (0, B_2, 0) + (0, 0, B_3)$$
$$\vec{A} \cdot \vec{B} = \vec{A} \cdot (B_1, 0, 0) + \vec{A} \cdot (0, B_2, 0) + \vec{A} \cdot (0, 0, B_3)$$
$$= A_1B_1 + A_2B_2 + A_3B_3$$

This amounts to projecting A onto each component of B, which is just the corresponding component of A (projecting a vector onto a vertical vector is the same as taking the vertical component of the first vector and so on), and multiplying their lengths, which gives us exactly the algebraic definition of the dot product

1.2 Cross Products

Cross products are, in my head, the counterpoint to a dot product. Whereas you can think of the dot product as what two vectors have in common (the component of one on another), the cross product gives the opposite, what the vectors do not have in common

Yet again we have two definitions, a geometric definition (again preferred by me), and an algebraic one

Geometric Definition: The cross product of two vectors $(\vec{A} \times \vec{B})$ is a new vector that is orthogonal to both vectors, whose length is equal to the area of the parallelogram formed by the two vectors being crossed.

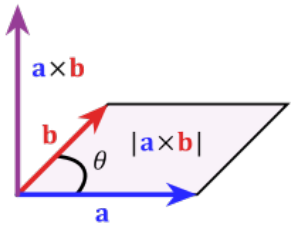

Figure 2: The cross product

This seems a little strange. Why this definition of all things? It becomes a little clearer when we consider what the area of a parallelogram actually is. This area is given by the base times the height, or for the vectors in the picture, $|\vec{A}||\vec{B}|\sin(\theta)$, because $|\vec{A}|$ is the length of the base and $|\vec{B}|\sin(\theta)$ gives the height

So if the length of the cross product is this area, then $|\vec{A}\times\vec{B}| = |\vec{A}||\vec{B}|\sin(\theta)$, and because we have introduced θ, this becomes a very useful formula.

One use is to find the distance from a point and a line, which can be computed by crossing two vectors to form a parallelogram between the line and the point, and dividing by the length of its base to find the point's distance off the line.

It's worth specifically noting that the cross product returning an orthogonal vector is particularly powerful and useful. The length is arguably less useful of the two properties.

Actually computing this cross product is a little strange and unexplained to me, but I'll leave it here

$$det\begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ A_1 & A_2 & A_3 \\ B_1 & B_2 & B_3 \end{vmatrix}$$

It's unclear to me why this produces an orthogonal vector, or why it is the length of the area of the parallelogram. Further research needed.

1.3 Vector/Parametric Forms of Lines

This is a short one. In $\mathbf{R}^2$, lines can be defined in point slope form. You start at a point, follow a slope, and you get your line. There's a similar concept in $\mathbf{R}^3$, with the vector form of a line. You start at a point, follow a vector, and you get your line.

The vector form of a line is as follows:

$$L = \{\vec{r}(t) = P_0 + t\vec{d}\}$$

Where P_0 is the point, and $\vec{d}$ is the direction vector of the line, the 3-space analog of a line's slope.

This can also be rewritten in terms of each component of the vector that is returned by $\vec{r}(t)$, known as the parametric form of the line

$$\begin{aligned} x &= P_{0x} + \vec{d}_x t \\ y &= P_{0y} + \vec{d}_y t \\ z &= P_{0z} + \vec{d}_z t \end{aligned}$$

To compare to the other forms, all of the following draw the same line

$$\begin{aligned} y &= x \\ y - 0 &= 1(x - 0) \\ L &= \{\vec{r}(t) = (0,0) + (1,1)t\} \\ \vec{r}(t) = (x,y) &: \begin{cases} x = 0 + 1t \\ y = 0 + 1t \end{cases} \end{aligned}$$

1.4 Vector and 3D Functions

Vector functions are pretty quick.

$$\vec{v}(t) = (x(t), y(t), \dots)$$

Their derivatives are the derivatives of the components of their vectors, and the usual derivative rules (product, quotient, etc.) work the way you would expect.

3D (or higher) functions are just as quick. Some function of x and y returns a z value, essentially taking the xy plane and raising it up along the z axis at each point (x, y) to the value $f(x, y)$

If you want to take a derivative of these, you have to take some care, because there are now two axises in which you can move, so the notion of a derivative becomes a bit fuzzy. Enter the partial derivative

Definition: The partial derivative of a function is the derivative of a multidimensional function along a single axis. You take a slice of the surface created by the function, and the resulting single dimension of movement is the respect of the derivative. All other variables are treated as constants.

$$\frac{\partial}{\partial x}(f(x) = xy) = (1x^0)y$$
$$\frac{\partial}{\partial y}(f(x) = xy) = x(1y^0)$$

One of the uses of the derivative in 2 dimensions was to approximate a function. The tangent line is pretty close to the function at very small steps. In 3d, we need a tangent plane, to represent the two dimensions of movement we have.

To find a tangent line in $\mathbf{R}^2$, we used a point and a slope $y-y_0 = f'(x)(x-x_0)$. In $\mathbf{R}^3$, we need two slopes, so

$$z - z_0 = \frac{\partial f}{\partial x}(x - x_0) + \frac{\partial f}{\partial y}(y - y_0)$$

represents the tangent plane.

A little manipulation of that later and we get a way to linearize the function, which is pretty similar

$$f(x, y) \approx f(a, b) + \frac{\partial f(a, b)}{\partial x}(x - a) + \frac{\partial f(a, b)}{\partial y}(y - b)$$

Where (a, b) is the point where you start, and (x, y) is where you end up. Which amounts to starting at (a, b), and walking a bit in the x multiplied by $\frac{\partial f}{\partial x}$, which is the slope (rise / run) in the x times the run, so you get the z that you should go up by, and you do the same in the y axis

1.4.1 Directional Derivatives

What if instead of nudging a 3D function by x or y, we nudge it along some vector? Pick some vector $\vec{v} = (a, b)$ to nudge along. This is like nudging by a in the x direction, and b in the y direction. The directional derivative is

$$\nabla_{\vec{v}} f(x, y) = a\frac{\partial f}{\partial x} + b\frac{\partial f}{\partial y} = \vec{v} \cdot \nabla f(x, y)$$

Note the subscript of ∇. Without it it's the gradient, as on the right, and with it it's the directional derivative. Similar to a partial derivative, this corresponds to a slice of the graph of a 3D function from some plane that isn't necessarily parallel to either axis x or y. That slice gives us some 2d function, where some step along that vector corresponds to a change in z, and some people write the directional vector as $\frac{\partial f}{\partial \vec{v}}$ to denote this.

1.4.2 Contour Maps

Before we leave 3D functions it's worth mentioning contour maps because they're cool.

The contour map of a 3d function is created by taking horizontal slices of the function at constant z values. This creates a neat way of visualizing these functions and will be useful for gradient fields later.

2 The Multidimensional Chain Rule

What came before was all pretty clear the first time through. After this, though, was when the lectures started to get muddy, which inspired me to take some more detailed notes.

The multidimension chain rule is a weird one. I couldn't ever find a satisfactory answer, intuition, or explanation for why it made sense.

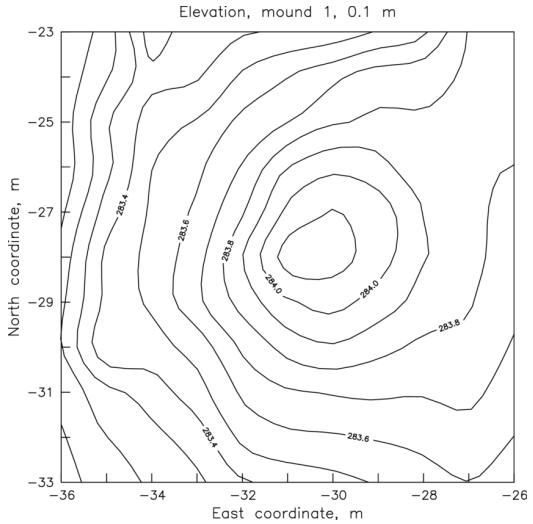

Figure 3: A contour map of some mound

It helped to first consider the single dimension chain rule.

$$\text{For a function} f(g(x)), \text{Let} u = g(x)$$
$$\frac{d}{dx} f(g(x)) = \frac{d}{dx} f(u) = f'(u) \frac{du}{dx}$$
$$\frac{du}{dx} = g'(x)$$
$$\frac{d}{dx} f(g(x)) = f'(g(x)) g'(x)$$

This is still pretty shallow for me. Seems more like a trick of notation than something that makes sense. If we go back and consider the derivative, though, it is definitionally the change in the value of the function when you change the input by a little bit.

$$f(x) = x^2$$
$$f(x + dx) = (x + dx)^2$$
$$= x^2 + 2xdx + dx^2$$

So when we nudge the function by dx, we get $x^2 + 2xdx + dx^2$. Before the nudge, the function was x^2, so the nudge changes the function by $2xdx + dx^2$. As $dx \to 0$, the value of dx^2 becomes very small and much less significant than $2xdx$. So when we nudge the function, its return value meaningfully changes by twice the value of the function at x, times the tiny nudge we made in the x, meaning that $f'(x) = 2x\ dx$. This is where the power rule comes from.

When we nudge $f(g(x))$, we first nudge $g(x)$ by something, and the new value then becomes the input of f.

Let's ignore $g(x)$ for a second. So long as $g(x)$ is differentiable, then a small change in x would feel no different than if we nudged $g(x)$ as far as f is concerned. Let's call $g(x)$ u instead. A small nudge in u, du, corresponds to a change of $f'(u)\ du$. The nudge du is itself the change in g when you change x, that nudge is $g'(x)\ dx$, which gives us the chain rule

$$\frac{d}{dx}f(g(x)) = f'(u)\ du$$
$$du = \frac{du}{dx}dx = g'(x)dx$$
$$\frac{d}{dx}f(g(x)) = f'(g(x))g'(x)\ dx$$

Okay, so with the single dimensional chain rule out of the way we still have a beast left. Let's look at something easy first.

$$\text{Let} f(a, b) = a + b$$
$$\text{What is } \frac{\partial}{\partial x}f(x^2, y^3).$$

The first thing to note is that a and b are private variables. The user who is feeding x's and y's into f has only indirect control over a and b. This means that taking a partial derivative with respect to a or b means nothing here, because the user inputting variables only knows what x and y are, and has never heard of a and b

Now, if you're just looking at this, you can just plug the functions in.

$$f(x^2, y^3) = x^2 + y^3$$
$$\frac{\partial f}{\partial x} = 2x$$
$$\frac{\partial f}{\partial y} = 3y^2$$

So you dont actually *need* the chain rule. You could plug all of the functions in and take a partial derivative as normal. However, if your functions get complex, say $f(u, v) = u^2 sin^2(v)$ and u and v are themselves complicated, the partial derivatives get annoying really fast, so having a way to pre-compute this with a formula would be nice.

Let's take that example. for $f(sin(x), cos(y))$ what happens when we nudge x? Let's again call sin(x) u.

$$f(u, \cos(y)) = u^2 \sin^2(\cos(y))$$
$$\frac{\partial f}{\partial u} = 2u(\sin^2(\cos(y))\ du$$
$$du = \frac{du}{dx}dx = \cos(x)dx$$
$$\frac{\partial f}{\partial u} = 2\sin(x)(\sin^2(\cos(y))\cos(x)$$

Okay, that's all fine and good I guess, but what does it actually mean? The first thing we did was ignore sin(x). We want a change in f with a change in x, but a change in x makes a change in u, so we considered the change in u first. The change caused by u was $\frac{\partial f}{\partial u}$, which is some term times du. So what is du? We really want a change with respect to x, so how can we find du in terms of dx? We can take $\frac{du}{dx}$, because u depends on the variable x, and then multiply that by dx, so that we're left with du.

So if $\frac{\partial f}{\partial u}$ is something times du, and du is $\frac{du}{dx}dx$, then $\frac{\partial f}{\partial x} = \frac{\partial f}{\partial u}\frac{du}{dx}$ This makes some sense. If we nudge x and it doubles u, and if we nudge u and it doubles f, it makes sense that nudging x quadruples f (by multiplying, not squaring/exponentiating).

This is nice, but what if v also depends on x? Then we also have to think about v now! So when we nudge x it makes some change to u, which makes a change to f, but it also makes a change to v, which makes a change to f. What's the total change? Well u and v aren't really related. The function can do whatever it wants with u and v, but no matter what, when you take a partial derivative of one, the other is constant. Unless they're the same thing, but that's beside the point. Because they create two independent changes, you just add them. You change x, it changes u, which changes f. It also changes v, which changes f. u and v don't change each other, so you don't multiply them or compose them or do anything fancy. The partial derivative is just the sum of the changes.

This is all honestly shallow intuition. I still don't really get all of this. My saving grace is this tree

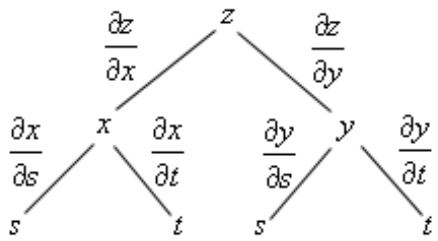

Figure 4: How to compute a multidimensional chain rule

If you want $\frac{\partial z}{\partial s}$ you follow all of the branches to s, multiply down the branch, and add the various branches. So,

$$\frac{\partial z}{\partial s} = \frac{\partial z}{\partial x}\frac{\partial x}{\partial s} + \frac{\partial z}{\partial y}\frac{\partial y}{\partial s}$$

This gives you what we had earlier. You keep unpacking various partial derivatives until you reach the variable you want, and then add all of the independent changes. I particularly like this tree becuase you could go 9 functions deep and it works the same way

My professor really likes emphasizing this derivative matrix $Df(x)$, which I think is supposed to be the Total Derivative but everything about this seems like we'll get to it later so I'll skip it for now and instead focus on

3 The Gradient

So first off, the gradient is, according to Grant Sanderson, weird. We'll start with the algebraic definition.

$$\nabla f(x, y, ...) = (\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \dots)$$

$$\begin{aligned}
\text{Let} f(x, y) &= x^2 sin(y) \\
\frac{\partial f}{\partial x} &= 2x \sin(y) \\
\frac{\partial f}{\partial y} &= x^2 \cos(y) \\
\nabla f(x, y) &= (\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}) \\
&= (2x \sin(y), x^2 \cos(y))
\end{aligned}$$

With this, it's important to note that the gradient is a **vector valued function**, which is a vector of the partial derivatives of f. (Grant called this a full derivative, as it contains all of the partial derivatives).

A helpful way to think of this is

$$\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \vdots \end{bmatrix}$$

And that $\nabla f(x)$ distributes $f(x)$ over that vector.

This is fine. It's a vector of partial derivatives, so what? What's neat about this is that it gives the direction of steepest ascent. That isn't at all intuitive at all, though. We just have a whole bunch of slopes in a whole bunch of directions. What gives?

Imagine you are at a point on a 3D graph. In order to climb the graph as fast as possible, you want to find the largest slope from a directional derivative along some unit vector evaluated at your point. (A unit vector because a $\vec{\infty}$ would always be the best choice)

To find this best vector, consider what the directional derivative actually is.

$$\nabla_{\vec{v}} f(a,b) = \nabla f(a,b) \cdot \vec{v}$$

If we want to maximize $\nabla_{\vec{v}} f(a,b)$, we really need to figure out what $\vec{v}$ dotted with $\nabla f(a,b)$ will give us the biggest slope (remember that $\vec{v}$ is a unit vector).

A dot product equals $|a||b|\cos(\theta)$. When the vectors lie on the same line ($\cos(\theta) = 1$) then we get a maximal dot product, assuming $|a|$ & $|b|$ are fixed, which they are in a directional derivative.

Therefore, the best unit $\vec{v}$ will be on the same line as $\nabla f(a,b)$. It needs to be a unit vector, so take $\frac{\nabla f(a,b)}{|\nabla f(a,b)|}$ and you get a unit vector that returns the highest possible slope from a directional derivative

The gradient itself, evaluated at (a,b), normalized, will then be the vector that, when traveled along, results in the greatest increase in altitude, because it is the highest slope, as we got from maximizing the directional derivative

Pretty cool stuff. It's the backbone of the Gradient Descent Algorithm and I implemented it for a Linear Regression Model