Wednesday, January 14, 2015
4:08 PM

Index Notation

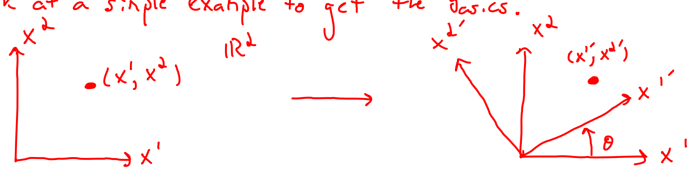
Before we go any further, we need to develop a better way to write things. So far we have made use of matrices to represent vectors, the metric and transformations. But...

3 reasons why matrices suck:

1. They are big (or can be) and writing them out explicitly can make you tired (unless you're Kevin).
2. Matrices do not commute, so when we write expressions we have to be careful about order.
3. Most importantly, we will soon encounter objects and operations that cannot be represented by matrices or matrix-multiplication.

Enter... index notation. This will re-produce all of our matrix goodness and more!

Let's look at a simple example to get the basics.



In matrix terms: $\begin{pmatrix} x^1 \\ x^2 \end{pmatrix} \to \begin{pmatrix} x^{1'} \\ x^{2'} \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x^1 \\ x^2 \end{pmatrix}$

Let's call $\begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} \Lambda^{1'}{}_1 & \Lambda^{1'}{}_2 \\ \Lambda^{2'}{}_1 & \Lambda^{2'}{}_2 \end{pmatrix}$

then we can write: $x^i \to x^{i'} = \Lambda^{i'}{}_j x^j$ where we evaluate this using the Einstein Summation Convention (repeated indices are summed).

Explicitly: $x^i \to x^{i'} = \Lambda^{i'}{}_1 x^1 + \Lambda^{i'}{}_2 x^2 \Rightarrow \begin{cases} x^{1'} = \Lambda^{1'}{}_1 x^1 + \Lambda^{1'}{}_2 x^2 = \cos\theta\, x^1 + \sin\theta\, x^2 \\ x^{2'} = \Lambda^{2'}{}_1 x^1 + \Lambda^{2'}{}_2 x^2 = -\sin\theta\, x^1 + \cos\theta\, x^2 \end{cases}$

BAM!!!

Wednesday, January 14, 2015
4:23 PM

Several important things to note:

1. Even in 38 dimensions, this expression would still be $x^i \to x^{i'} = \Lambda^{i'}{}_j x^j$ (size doesn't matter!)
2. We could have also written $x^i \to x^{i'} = x^j R^{i'}{}_j = x^1 R^{i'}{}_1 + x^2 R^{i'}{}_2 \Rightarrow \begin{cases} x^{1'} = x^1\cos\theta + x^2\sin\theta \\ x^{2'} = -x^1\sin\theta + x^2\cos\theta \end{cases}$

   Order doesn't matter! We get the same thing w/ $x^j R^{i'}{}_j$ or $R^{i'}{}_j x^j$. This is not the case w/ matrices!
3. You can immediately evaluate something like $M_{ijk}N^{ijk}$ given the elements of M_{ijk} and N^{ijk} even though you cannot represent this in terms of matrix multiplication (or matrices at all!)
4. You may have noticed that anytime an index is repeated it comes in an "upper" and "lower" pair, e.g. $\Lambda^{i'}{}_j x^j$. You will eventually understand what this means and why.
5. In 4D we use greek indices μ, ν, λ, etc. that take values $0, 1, 2, 3$.
   (ct, x, y, z)
6. In some cases you will see objects with one primed and one unprimed index. This is a transformation from one coordinate system to another, e.g. $x^{\mu'} = \Lambda^{\mu'}{}_\nu x^\nu$. In fact $\Lambda^{\mu'}{}_\nu$ will be the only object we encounter like this. Any other object should have all primed or all unprimed indices, e.g. $M^{\mu\nu}{}_\lambda$, $H_{\lambda'}{}^{\mu'}{}_{\nu'}$.
7. In some cases you will want to revert back to matrix expressions (when you can!). To do so we identify the row and column of a two index object as left (row) and right (column). Up or down does not matter, but to get the order right (which matters for matrices) just make sure repeated indices are immediately adjacent, e.g. $\Lambda^{\mu'}{}_\nu x^\nu$ works, but $x^\nu \Lambda^{\mu'}{}_\nu$ does not!!

   $$\Lambda^{\mu'}{}_\nu x^\nu \Downarrow \Lambda x \qquad x^\nu \Lambda^{\mu'}{}_\nu \Downarrow x\Lambda$$

Sunday, January 25, 2015
5:04 PM

Tensors

Represent physical quantities that are invariant but when given an explicit coordinate representation will typically have components that transform.

In terms of a labelling scheme (upper, lower) we have seen:

$\underbrace{(0,0)}$ scalar

$\underbrace{(1,0)}$ vectors

$\underbrace{(0,1)}$ dual vectors

$\underbrace{(0,2)}$ the metric $g_{\mu\nu}$

$\underbrace{(2,0)}$ the inverse metric $g^{\mu\nu}$

In general you could have $(\overset{\text{upper}}{m}, \underset{\text{lower}}{n})$ mixed indices, e.g. $\underbrace{T^{\mu\nu\alpha}{}_{\lambda\beta\gamma\delta}}_{(3,4)\text{ tensor}}$

A popular definition: A tensor is something that transforms like a tensor.

A better definition: Tensors are multi-linear maps from the space of vectors and dual vectors into the reals, i.e. $T(V,V,V,\omega,\omega,\omega,\omega) \to \#$

The latter definition is born out by a "well-fed" tensor:

Well-fed : $T^{\mu\nu\alpha}{}_{\lambda\beta\gamma\delta} V^\lambda V^\beta V^\gamma V^\delta \omega_\mu \omega_\nu \omega_\alpha \in \mathbb{R}$ no "free" indices

contrast with

Starving : $T^{\mu\nu\alpha}{}_{\lambda\beta\gamma\delta} V^\lambda V^\beta V^\gamma \omega_\mu \omega_\nu = F^\alpha{}_\delta$

Over-fed: $T^{\mu\nu\alpha}{}_{\lambda\beta\gamma\delta} V^\lambda V^\beta V^\gamma V^\delta V^\eta \omega_\mu \omega_\nu \omega_\alpha = H^\eta$

The work we did on vectors and dual-vectors pays off big time now!

The transformation of tensor components is determined by their index structure:

$c \to c' = c$

$V^\mu \to V^{\mu'} = \Lambda^{\mu'}{}_\mu V^\mu$

$\omega_\mu \to \omega_{\mu'} = \Lambda^\mu{}_{\mu'} \omega_\mu$

$T^{\mu\nu} \to T^{\mu'\nu'} = \Lambda^{\mu'}{}_\mu \Lambda^{\nu'}{}_\nu T^{\mu\nu}$

$H^\mu{}_\nu \to H^{\mu'}{}_{\nu'} = \Lambda^{\mu'}{}_\mu \Lambda^\nu{}_{\nu'} H^\mu{}_\nu$

$G_{\mu\nu} \to G_{\mu'\nu'} = \Lambda^\mu{}_{\mu'} \Lambda^\nu{}_{\nu'} G_{\mu\nu}$

Remember if you want to use matrix multiplication you need to get contracted indices adjacent

$\Lambda \equiv \Lambda^{\mu'}{}_\mu$ or $\Lambda^{\nu'}{}_\nu$

$\Lambda T \Lambda^T = T'$

$\Lambda H \Lambda^{-1^T} = H'$

$\Lambda^{-1^T} G \Lambda^{-1} = G'$

$G_{\mu\nu} \to G_{\mu'\nu'} = \Lambda^{\mu}{}_{\mu'} \Lambda^{\nu}{}_{\nu'} G_{\mu\nu}$ to get repeated indices adjacent $\Lambda^{-1^T} G \Lambda^{-1} = G'$

etc.

using $\Lambda_{\nu}{}^{\nu'} = (\Lambda^{\nu'}{}_{\nu})^T$

and $\Lambda^{\nu}{}_{\nu'} = (\Lambda^{\nu'}{}_{\nu})^{-1}$

thus $\Lambda_{\nu'}{}^{\nu} = (\Lambda^{\nu'}{}_{\nu})^{-1^T}$