Einstein Summation Notation

Aditya Sengupta

January 29, 2019

1 Introduction

Many years ago, someone decided to drop the summation sign $\sum$ from expressions that would normally need it, and trust that everyone would be smart enough to recognize there would have to be a summation there without writing it. That man's name? Albert Einstein. (No, really.)

2 Usage

Einstein notation essentially drops the summation sign from a sum over a dummy variable:

$$\sum_{i=1}^{n} a_i \to a_i \tag{1}$$

We consider the summation to be implicit based on knowing that i has no external meaning, and knowing that n is the dimension of the vector a. This is useful when representing the elements of a vector coupled with the unit vectors in each direction,

$$\vec{r} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = x_1\hat{e_1} + x_2\hat{e_2} + \cdots + x_n\hat{e_n} = \sum_{i=1}^{n} x_i\hat{e_i} = x_i\hat{e_i} \tag{2}$$

This notation allows us to represent the idea of different coordinate systems compactly:

$$\vec{r} = x_i\hat{e_i} = x_i'\hat{e_i'} \tag{3}$$

In conventional notation, we would represent the transformation $\hat{e_i} \to \hat{e_i'}$ by a transformation matrix,

$$\vec{r'} = \Lambda\vec{r} \tag{4}$$

$$\begin{bmatrix} x_1' \\ x_2' \\ \vdots \\ x_n' \end{bmatrix} = \begin{bmatrix} \lambda_{11} & \lambda_{12} & \dots & \\ \lambda_{21} & \dots & & \\ \vdots & \dots & & \\ & & & \lambda_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{5}$$

The matrix is a compact way of representing this transformation, but given how regular the component naming is, it could be made much more so. We can do this by considering that the matrix is just encoding a series of linear combinations of variables:

$$x_1' = \lambda_{11}x_1 + \lambda_{12}x_2 + \cdots + \lambda_{1n}x_n \tag{6}$$
$$x_2' = \lambda_{21}x_1 + \lambda_{22}x_2 + \cdots + \lambda_{2n}x_n \tag{7}$$

$$\vdots$$

$$x_n' = \lambda_{n1}x_1 + \lambda_{n2}x_2 + \cdots + \lambda_{nn}x_n \tag{8}$$

We can write each of these as a summation, and then drop the summation sign,

$$x_1' = \sum_{i=1}^{n} \lambda_{1i}x_i = \lambda_{1i}x_i \tag{9}$$
$$x_2' = \sum_{i=1} \lambda_{2i}x_i = \lambda_{2i}x_i \tag{10}$$

$$\vdots$$

$$x_n' = \sum_{i=1}^{n} \lambda_{ni}x_i = \lambda_{ni}x_i \tag{11}$$

which we can write all as one expression by denoting the arbitrary index on the new variable by j,

$$x_j' = \lambda_{ji}x_i \tag{12}$$

Since we drop the summation signs that make some aspects more explicit, we often rely on the indices to provide both information about the sum and to keep track of whether the sum is valid. It is typically required, for example, that each index appears twice in a summation expression; if you do not see each index twice, it became invalid somewhere.

3 Dot and Cross Products

The logical jump in using Einstein notation is not really in dropping the sum. It is in representing with a summation what would otherwise be represented with vector-specific notation. For example, the dot product of two vectors is usually written as a property of vectors, $\vec{a} \cdot \vec{b}$, and switching only to the summation notation to represent dot products feels like a stretch, doubly so without the summation sign itself. In Einstein notation, this would become

$$\vec{a} \cdot \vec{b} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \cdot \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} = a_1b_1 + a_2b_2 + \cdots + a_nb_n = \sum_{i=1}^{n} a_ib_i = a_ib_i \tag{13}$$

This is a slightly informal but usually accepted use of the notation. Properly, because we are summing over two sets of variables, we would write this using the *Kronecker delta function*, defined as follows:

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{14}$$

With the Kronecker delta, we compute the dot product by individually summing over all the components of the vector, and selecting only the desired components:

$$\vec{a} \cdot \vec{b} = \sum_{i=1}^{n} \sum_{j=1}^{n} a_i b_j \delta_{ij} = a_i b_j \delta_{ij} \tag{15}$$

Note that as per convention, we see two of each index in this expression.

Similarly to the dot product, we can write the cross product of two vectors in Einstein notation. This requires a slightly more involved starting coefficient. Explicitly, the cross product is written in terms of a determinant, but a determinant is just a specific type of summation rule, which we will develop from here.

$$\vec{a} \times \vec{b} = \begin{vmatrix} \hat{e_1} & \hat{e_2} & \hat{e_3} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix} \tag{16}$$

where we use numbered coordinates rather than the usual $\hat{x}, \hat{y}, \hat{z}$ so that numbered coordinates are more intuitive. We evaluate this determinant and find

$$\vec{a} \times \vec{b} = \hat{e_1}\,(a_2 b_3 - a_3 b_2) - \hat{e_2}\,(a_1 b_3 - a_3 b_1) + \hat{e_3}\,(a_1 b_2 - a_2 b_1) \tag{17}$$

Note that none of the numerical coefficients in any individual expression are the same, but we iterated through all $3! = 6$ possibilities for permutations of 1, 2, 3 without repetition. The most general statement we can make about the positioning of the plus and minus signs is based on the number of inversions (number of swaps you would have to do to restore the order) in the order 123: if it is even (including zero), the coefficient ends up being +1, and if it is odd, the coefficient is -1.

Order	**Number of inversions**	**Coefficient**
123	0	+1
132	1	-1
213	1	-1
231	2	+1
312	2	+1
321	3	-1

Instead of looking at inversions, it is also possible to look at rotations: if the order is 123 or a rotation of 123, the coefficient is +1, and if it is 321 or a rotation thereof, the coefficient is -1.

This allows us to define the *Levi-Civita tensor*, a 27-element rank-three tensor which, thanks to Einstein notation, we only have to interact with as if it were a function of indices rather than through tensor properties or rules. 21 of its elements are zeroes, and 6 of its elements follow the above rotation/inversion rules:

$$\epsilon_{ijk} = \begin{cases} 0 & i = j \text{ or } j = k \text{ or } k = i \\ 1 & \text{even orientation: } 123, 231, 312 \\ -1 & \text{odd orientation: } 132, 213, 321 \end{cases} \tag{18}$$

Based on this, we can write a general expression for a determinant,

$$\begin{vmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{vmatrix} = \sum_i \sum_j \sum_k \epsilon_{ijk} a_i b_j c_k \tag{19}$$

and therefore the cross product is given by

$$\vec{a} \times \vec{b} = \epsilon_{ijk} a_i b_j \hat{e_k} \tag{20}$$

4 Divergence and Curl

The divergence and curl of a vector can easily be represented in Einstein notation, as they can be represented easily as dot or cross products:

$$\vec{\nabla} \cdot \vec{a} = \frac{\partial}{\partial e_i} a_j \delta ij \tag{21}$$

$$\vec{\nabla} \times \vec{a} = \epsilon_{ijk} \frac{\partial}{\partial e_i} b_j \hat{e_k} \tag{22}$$

(Note that this section is very short: see how easy everything is with Einstein notation?)

5 Levi-Civita Tensor Combinations

The Levi-Civita tensor can be combined with itself with different indices, with one index matching up:

$$\epsilon_{ijk} \epsilon_{ilm}$$

The result can be written in terms of a combination of Kronecker deltas. First consider the case where $j = l$; when it is also the case that $k = m$, we get complete symmetry and therefore an output of 1. (Complete symmetry means ijk and ilm must have the same sequence of corresponding numbers, i.e. if the first Levi-Civita tensor term yields 1, then the other must as well, and if the first term yields -1, then the same goes for the second, so either way their product is 1.) This corresponds to the delta term $\delta_{jl}\delta_{km}$. When it is not the case that $k = m$, the output must be zero, therefore no delta terms need to be added to account for this. Then, consider the case $j = m$; when it is also the case that $k = l$, we get complete antisymmetry and therefore an output of -1. This can be shown through a case-by-case analysis; there is a summation over five terms, but most of them are zero. The method here is much easier, though.

$$\epsilon_{ijk} \epsilon_{ilm} = \delta_{jl}\delta_{km} - \delta_{jm}\delta_{kl} \tag{23}$$

6 The Advantage

Einstein notation makes it easy to manipulate vectors and prove identities that would otherwise be essentially impossible due to how impractical it is to manipulate an arbitrary component-specific vector (e.g. take an arbitrary $\vec{r} = a\hat{i} + b\hat{j} + c\hat{z}$, take its curl, ...) For example, consider the well-known fact in vector calculus that *the divergence of the curl is zero.* We will prove it here with Einstein notation.

Proof.

$$\vec{\nabla} \cdot \left(\vec{\nabla} \times \vec{a}\right) = \vec{\nabla} \cdot \left(\epsilon_{ijk} \frac{\partial}{\partial e_i} a_j \hat{e_k}\right) = \frac{\partial}{\partial e_k} \epsilon_{ijk} \frac{\partial}{\partial e_i} a_j \tag{24}$$

Often, in this notation, partial derivatives are also written more compactly. Here, since the Levi-Civita tensor is invariant under the derivative, we get

$$\vec{\nabla} \cdot \left(\vec{\nabla} \times \vec{a}\right) = \epsilon_{ijk}\partial_k\partial_i a_j \tag{25}$$

We can rewrite this to exploit the antisymmetry of the Levi-Civita tensor,

$$\vec{\nabla} \cdot \left(\vec{\nabla} \times \vec{a}\right) = \frac{1}{2}\left(\epsilon_{ijk}\partial_k\partial_i a_j + \epsilon_{ijk}\partial_k\partial_i a_j\right) \tag{26}$$

We create one inversion in the second tensor expression, in the two variables denoting the derivatives,

$$\vec{\nabla} \cdot \left(\vec{\nabla} \times \vec{a}\right) = \frac{1}{2}\left(\epsilon_{ijk}\partial_k\partial_i a_j - \epsilon_{kji}\partial_k\partial_i a_j\right) \tag{27}$$

Switching i and k in the second part of the expression,

$$\vec{\nabla} \cdot \left(\vec{\nabla} \times \vec{a}\right) = \frac{1}{2}\left(\epsilon_{ijk}\partial_k\partial_i a_j - \epsilon_{ijk}\partial_i\partial_k a_j\right) \tag{28}$$

Because of Clairaut's theorem, we can switch the order of the derivatives, and we get

$$\vec{\nabla} \cdot \left(\vec{\nabla} \times \vec{a}\right) = \frac{1}{2}\left(\epsilon_{ijk}\partial_k\partial_i a_j - \epsilon_{ijk}\partial_k\partial_i a_j\right) = 0 \tag{29}$$

$\square$

Similarly, we can analyze the *vector triple product*, in which the Levi-Civita interchange rule will become useful due to two cross products.

$$\vec{a} \times (\vec{b} \times \vec{c}) = \vec{b}(\vec{a} \cdot \vec{c}) - \vec{c}(\vec{a} \cdot \vec{b}) \tag{30}$$

Proof.

$$\vec{b} \times \vec{c} = \epsilon_{ijk} b_i c_j \hat{e_k} \implies \vec{a} \times (\vec{b} \times \vec{c}) = \epsilon_{lkn} a_l (\epsilon_{ijk} b_i c_j) \hat{e_n} \tag{31}$$

Note here that k appears twice; we first drop it from the $\hat{e_k}$ as it becomes implicit that in a sum in which i and j appear twice but k appears once that the output is in k, which is then matched by the k in ϵ_{lkn}. We reorder the terms and permute the Levi-Civita tensors,

$$\vec{a} \times (\vec{b} \times \vec{c}) = \epsilon_{knl}\epsilon_{kij} a_l b_i c_j \hat{e_n} \tag{32}$$

Applying the Levi-Civita product identity,

$$\vec{a} \times (\vec{b} \times \vec{c}) = (\delta_{ni}\delta_{lj} - \delta_{nj}\delta li) a_l b_i c_j \hat{e_n} \tag{33}$$

We can split this,

$$\vec{a} \times (\vec{b} \times \vec{c}) = \delta_{ni}\delta_{lj} a_l b_i c_j \hat{e_n} - \delta_{nj}\delta_{li} a_l b_i c_j \hat{e_n} \tag{34}$$

Then, we group the deltas with the components of the corresponding indices,

$$\vec{a} \times (\vec{b} \times \vec{c}) = (b_i \hat{e_n} \delta_{in}) \, (a_l c_j \delta_{lj}) - (c_j \hat{e_n} \delta_{jn}) \, (a_l b_i \delta_{li}) \tag{35}$$

We can group together indices that are guaranteed to be the same by the Kronecker deltas attached to them,

$$\vec{a} \times (\vec{b} \times \vec{c}) = (b_i \hat{e_i}) \, (a_j c_j) - (c_j \hat{e_j}) \, (a_i b_i) \tag{36}$$

By definition, this becomes

$$\vec{a} \times (\vec{b} \times \vec{c}) = \vec{b}(\vec{a} \cdot \vec{c}) - \vec{c}(\vec{a} \cdot \vec{b}) \tag{37}$$

□