Matrix Multiplication

Our goal here is to understand multiplication of matrices using sum notation. Starting from the definition of a matrix product, $\boldsymbol{C} = \boldsymbol{AB}$, where $\boldsymbol{A}$ is an $m \times n$ matrix, and $\boldsymbol{B}$ is a $n \times p$ matrix, such that $\boldsymbol{C}$ is a $m \times p$ matrix. We could write this out with arbitrary 2×2 matrices to illustrate:

$$\begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} \tag{1}$$

So, the obvious question is: how are the elements c_{ij} are related to the elements a_{ij} and b_{ij}? It's defined in the following way (this is really the *definition* of the matrix product):

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} \tag{2}$$

We've introduced this extra variable k to run from 1 to n, where n is the number of columns in $\boldsymbol{A}$ or rows in $\boldsymbol{B}$. To understand the point of doing this, it's worth just writing this out in full as before:

$$\begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} \tag{3}$$

Work through it and convince yourself the formula works! That extra variable k is what ensures that we sum over all the relevant elements. Extending this to higher dimensions is easy, but obviously as the dimension increases, so does n, and therefore the sum over k contains more terms!

In the problem sheet this week, we are asked to show that matrix multiplication is associative, which means that $\boldsymbol{X}(\boldsymbol{YZ}) = (\boldsymbol{XY})\boldsymbol{Z}$ - i.e. if we have a product of three matrices, it doesn't matter which pair we do first. Let's manipulate this sum notation to let us do this. Let us use the notation that the element (i, j) of a matrix $\boldsymbol{X}(\boldsymbol{YZ})$ is denoted by $[\boldsymbol{X}(\boldsymbol{YZ})]_{ij}$. Now we will express this element using sum notation. Note first that:

$$[\boldsymbol{YZ}]_{kj} = \sum_l y_{kl} z_{lj} \tag{4}$$

Using equation (2). We've renamed the "summing variable" to l, because we're about to bang this into another sum, and it will be easier (trust me!)[1]. Hopefully you can see this is exactly the same as equation (2) - it's just element (k, j) of the matrix $\boldsymbol{YZ}$. Now to find $[\boldsymbol{X}(\boldsymbol{YZ})]_{ij}$, we write another sum:

$$[\boldsymbol{X}(\boldsymbol{YZ})]_{ij} = \sum_k x_{ik} [\boldsymbol{YZ}]_{kj} = \sum_k x_{ik} \left(\sum_l y_{kl} z_{lj} \right) \tag{5}$$

Which follows just from the definition of the matrix product (first part), and then plugging in the result from equation (4) (second part). We want to prove that this is the same thing as $[(\boldsymbol{XY})\boldsymbol{Z})]_{ij}$. Like all good mathematical operatives, we will make this work and then discuss the legality of the steps afterwards:

$$\begin{aligned} [\boldsymbol{X}(\boldsymbol{YZ})]_{ij} &= \sum_k x_{ik} \left(\sum_l y_{kl} z_{lj} \right) = \sum_k \left(\sum_l x_{ik} y_{kl} z_{lj} \right) = \sum_l \left(\sum_k x_{ik} y_{kl} z_{lj} \right) \\ &= \sum_l \left(\sum_k x_{ik} y_{kl} \right) z_{lj} = \sum_l [\boldsymbol{XY}]_{il} z_{lj} = [(\boldsymbol{XY})\boldsymbol{Z})]_{ij} \end{aligned}$$

Now let's discuss the legality. Firstly, we just use the definition from equation (5). Secondly, we pull the sum over l so that x_{ik} is *inside* the sum - this is perfectly legal, as the sum is over l, so x_{ik} is basically a constant (as far as the sum is concerned). Thirdly, we swap the order of the two sums - this is also perfectly legal, because we haven't moved anything *outside* a sum which depends on the variable being summed over. This would be definitely **illegal** - I can't factorise out x_{ik} such that it is *outside* the sum over k, for example. In the final step, we just do the opposite of the second step, and move z_{lj} outside of the sum over k - again, perfectly legal, as z_{lj} doesn't depend on k in any way. Then we just pack it all up in a nice matrix shaped box and call it a job well done. Nice!

[1] I've also made the sum just "over l" rather than explicitly from 1, it means the same thing.