Chain Rule on Matrix Multiplication

Mostafa Samir

August 2021

1 Defining the Problem

Let $\mathbf{X} \in \mathbb{R}^{n \times m}$, $\mathbf{W} \in \mathbb{R}^{m \times d}$, and $\mathbf{Y} \in \mathbb{R}^{n \times d}$ such that:

$$\mathbf{Y} = \mathbf{W}\mathbf{X}$$

Moreover, let $f : \mathbb{R}^{n \times d} \to \mathbb{R}$ be a real-valued function of $\mathbf{Y}$, $f = f(\mathbf{Y})$. You can think of such a setting as a $\mathbf{Y}$ being the activations of a linear layer in a neural network with a $\mathbf{0}$ for the bias vector. In this setting, f serves as the loss function. The problem we'll try to solve here arises when we try to calculate $\frac{\partial f}{\partial \mathbf{X}}$ and $\frac{\partial f}{\partial \mathbf{W}}$ via the chain rule (which is how gradients are calculated when a neural network is being trained). By applying the chain rule we end up with:

$$\frac{\partial f}{\partial \mathbf{X}} = \frac{\partial f}{\partial \mathbf{Y}} \frac{\partial \mathbf{Y}}{\partial \mathbf{X}},$$
$$\frac{\partial f}{\partial \mathbf{W}} = \frac{\partial f}{\partial \mathbf{Y}} \frac{\partial \mathbf{Y}}{\partial \mathbf{W}}$$

The complications arises when we find out that $\frac{\partial \mathbf{Y}}{\partial \mathbf{X}}$ and $\frac{\partial \mathbf{Y}}{\partial \mathbf{W}}$ are 4th rank tensors (i.e. 4D arrays) of the sizes $n \times d \times n \times m$ and $n \times d \times m \times d$ respectively. We can see a small example of such tensor for $\mathbf{Y}, \mathbf{W} \in \mathbb{R}^{2 \times 2}$ such that:

$$\mathbf{Y} = \begin{pmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{pmatrix}, \mathbf{W} = \begin{pmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \end{pmatrix}$$

Taking the derivative of $\mathbf{Y}$ with respect to $\mathbf{W}$ means that we take the derivative of each element $\mathbf{Y}$ with respect to $\mathbf{W}$, that is:

$$\frac{\partial \mathbf{Y}}{\partial \mathbf{W}} = \begin{pmatrix} \frac{\partial Y_{11}}{\partial \mathbf{W}} & \frac{\partial Y_{12}}{\partial \mathbf{W}} \\ \frac{\partial Y_{21}}{\partial \mathbf{W}} & \frac{\partial Y_{22}}{\partial \mathbf{W}} \end{pmatrix}$$

Similarly, taking the derivative of a scalar like Y_{11} with respect to a matrix $\mathbf{W}$ means that we're taking the derivative of the scalar Y_{11} with respect to each element in the matrix $\mathbf{W}$

$$\frac{\partial Y_{11}}{\partial \mathbf{W}} = \begin{pmatrix} \frac{\partial Y_{11}}{\partial W_{11}} & \frac{\partial Y_{11}}{\partial W_{12}} \\ \frac{\partial Y_{11}}{\partial W_{21}} & \frac{\partial Y_{11}}{\partial W_{22}} \end{pmatrix}$$

Applying the same rule to all the elements of $\mathbf{Y}$, we end up with:

$$\frac{\partial \mathbf{Y}}{\partial \mathbf{W}} = \begin{pmatrix} \begin{pmatrix} \frac{\partial Y_{11}}{\partial W_{11}} & \frac{\partial Y_{11}}{\partial W_{12}} \\ \frac{\partial Y_{11}}{\partial W_{21}} & \frac{\partial Y_{11}}{\partial W_{22}} \end{pmatrix} & \begin{pmatrix} \frac{\partial Y_{12}}{\partial W_{11}} & \frac{\partial Y_{12}}{\partial W_{12}} \\ \frac{\partial Y_{12}}{\partial W_{21}} & \frac{\partial Y_{12}}{\partial W_{22}} \end{pmatrix} \\ \begin{pmatrix} \frac{\partial Y_{21}}{\partial W_{11}} & \frac{\partial Y_{21}}{\partial W_{12}} \\ \frac{\partial Y_{21}}{\partial W_{21}} & \frac{\partial Y_{21}}{\partial W_{22}} \end{pmatrix} & \begin{pmatrix} \frac{\partial Y_{22}}{\partial W_{11}} & \frac{\partial Y_{22}}{\partial W_{12}} \\ \frac{\partial Y_{22}}{\partial W_{21}} & \frac{\partial Y_{22}}{\partial W_{22}} \end{pmatrix} \end{pmatrix}$$

Which is a 4th-rank tensor of the size $2 \times 2 \times 2 \times 2$. Such structures become quickly inefficient to calculate once we start dealing with slightly larger matrices. This poses a practical problem when we're, for example, training a deep neural network with many large weight matrices. The goal of this document is to prove and show that there exists a cheaper way to compute the same two derivatives $\frac{\partial \mathbf{Y}}{\partial \mathbf{X}}$ and $\frac{\partial \mathbf{Y}}{\partial \mathbf{W}}$ without the need to handle such scary tensors.

The work shown here is mainly derived from a handout by Justin Johnson from Stanford's CS231n course. The difference is that in the handout, a special case of the proof is worked for matrices with given sizes. Here we'll show that the same rules hold for any matrix with any arbitrary size.

2 Preliminary Knowledge

2.1 Matrix Multiplication

For any two matrices $\mathbf{X} \in \mathbb{R}^{n \times m}$, $\mathbf{W} \in \mathbb{R}^{m \times d}$, there is a matrix $\mathbf{Y} = \mathbf{XW}$ such that $\mathbf{Y} \in \mathbb{R}^{n \times d}$ and:

$$Y_{pq} = \sum_{r=1}^{m} X_{pr} W_{rq}$$

2.2 Multivariable Chain Rule

For a real-valued function $f = f(x_1(t), x_2(t), ..., x_n(t))$

$$\frac{df}{dt} = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} \frac{dx_i}{dt}$$

3 Calculating $\frac{\partial f}{\partial \mathbf{X}}$

Following the steps of Justin Johnson, we start simple by calculating $\frac{\partial f}{\partial X_{ij}}$ and then generalize from that. If we thought about f's dependency on X_{ij}, we'll

quickly recognize that it gets that dependency through $\mathbf{Y}$, i.e $f = f(\mathbf{Y}(X_{ij}))$. Moreover, we know that f depends on every element Y_{pq} of $\mathbf{Y}$, and Y_{pq} also has some sort of dependency on X_{ij}, which eventually yields the fact that:

$$f = f(Y_{11}(X_{ij}), Y_{12}(X_{ij}), ..., Y_{pq}(X_{ij}))$$

By applying the multivariable chain rule on f, we get:

$$\frac{\partial f}{\partial X_{ij}} = \sum_{p=1}^{n} \sum_{q=1}^{d} \frac{\partial f}{\partial Y_{pq}} \frac{\partial Y_{pq}}{\partial X_{ij}}$$

We can use the definition of matrix multiplication in order to calculate $\frac{\partial Y_{pq}}{\partial X_{ij}}$.

$$\frac{\partial Y_{pq}}{\partial X_{ij}} = \sum_{r=1}^{m} \frac{\partial}{\partial X_{ij}} (X_{pr} W_{rq}) = \begin{cases} W_{jq}, & p = i, r = j \\ 0 & \text{otherwise} \end{cases}$$

By substituting the derivative back into the chain rule expression we get:

$$\frac{\partial f}{\partial X_{ij}} = \sum_{p \neq i} \sum_{q=1}^{d} \frac{\partial f}{\partial Y_{pq}} \times 0 + \sum_{q=1}^{d} \frac{\partial f}{\partial Y_{iq}} W_{jq} = \sum_{q=1}^{d} \frac{\partial f}{\partial Y_{iq}} W_{jq}$$

Note that $W_{jq} = W_{qj}^{\top}$, hence:

$$\frac{\partial f}{\partial X_{ij}} = \sum_{q=1}^{d} \frac{\partial f}{\partial Y_{iq}} W_{qj}^{\top}$$

By comparing the equation above with the definition of matrix multiplication, we can easily conclude that:

$$\frac{\partial f}{\partial \mathbf{X}} = \frac{\partial f}{\partial \mathbf{Y}} \mathbf{W}^{\top}$$

4 Calculating $\frac{\partial f}{\partial \mathbf{W}}$

By an argument similar to what we did in the last section, we can write:

$$\frac{\partial f}{\partial W_{ij}} = \sum_{p=1}^{n} \sum_{q=1}^{d} \frac{\partial f}{\partial Y_{pq}} \frac{\partial Y_{pq}}{\partial W_{ij}}$$

From the definition of matrix multiplication, we get that:

$$\frac{\partial Y_{pq}}{\partial W_{ij}} = \sum_{r=1}^{m} \frac{\partial}{\partial W_{ij}} (X_{pr} W_{rq}) = \begin{cases} X_{pi}, & r = i, q = j \\ 0 & \text{otherwise} \end{cases}$$

Substituting that into the chain rule expression gives us:

$$\frac{\partial f}{\partial W_{ij}} = \sum_{p=i}^{n} \sum_{q \neq j} \frac{\partial f}{\partial Y_{pq}} \times 0 + \sum_{p=1}^{n} \frac{\partial f}{\partial Y_{pj}} X_{pi} = \sum_{p=1}^{n} \frac{\partial f}{\partial Y_{pj}} X_{pi}$$

Noticing that $X_{pi} = X_{ip}^{\top}$ yields:

$$\frac{\partial f}{\partial W_{ij}} = \sum_{p=1}^{n} X_{ip}^{\top} \frac{\partial f}{\partial Y_{pj}}$$

Which defines the individual elements of the matrix multiplication:

$$\frac{\partial f}{\partial \mathbf{W}} = \mathbf{X}^{\top} \frac{\partial f}{\partial \mathbf{Y}}$$