Matrix Calculus

Jack David Carson - January 31, 2025

In machine learning, the fundmental operation of backpropogation is to take the derivative of a matrix with respect to a vector or another matrix. We may seek to calculate the gradient of a matrix $A_{m \times n}$ with respect to $B_{p \times q}$ and accumulate the partial derivatives in a Jacobian **tensor** J whose enteries are given as $J_{ijkl} = \partial A_{ij} / \partial B_{kl}$.

0.1 Gradient of a Matrix with Respect to a Matrix

For the matrix expression $A_{m \times p} = B_{m \times n} C_{n \times p}$, we can coordinate expand into $A_{ij} = \sum_{r=1}^{n} B_{ir} C_{rj}$ by definition of matrix multiplication such that the derivative

$$\begin{aligned} \frac{\partial A_{ij}}{\partial B_{kl}} &= \frac{\partial}{\partial B_{kl}} \left(\sum_{r=1}^{n} B_{ir} C_{rj} \right) \\ &= \sum_{r=1}^{n} \frac{\partial B_{ir}}{\partial B_{kl}} C_{rj} \end{aligned}$$

and we can define the **Kronecker delta** δ_{ij} as

$$\delta_{i,j} := \begin{cases} 1 & \text{if} \quad i = j \\ 0 & \text{if} \quad i \neq j \end{cases}$$

which effectively "selects" the term of $i = j$ from the sum. Since for an isolated matrix, a value B_{ij} does not depend on any other term but itself, its gradient with respect to other terms will always be 0 or 1. Hence

$$\begin{aligned} \frac{\partial A_{ij}}{\partial B_{kl}} &= \sum_{r=1}^{n} \frac{\partial B_{ir}}{\partial B_{kl}} C_{rj} \\ &= \sum_{r=1}^{n} \delta_{ik} \delta_{rl} C_{rj} \\ &= \delta_{ik} C_{lj}. \end{aligned}$$

Here, the first Kronecker delta δ_{ik} is not affected by the summation and is pulled out, and the second δ_{rl} sets every term except for C_{lj} to 0 where $r = l$.

0.2 Gradient of a Vector with Respect to a Matrix

In the context of an expression $\boldsymbol{y} = A\boldsymbol{x}$, we can take the derivative starting again in coordinate form $y_i = \sum_{j=1}^{n} A_{ij} x_j$ to find

$$\frac{\partial y_i}{\partial A_{kl}} = \frac{\partial A_{ij}}{\partial A_{kl}} x_j = \delta_{ik} x_l$$

0.2.1 Simple Loss Function

If we have simple loss $\mathcal{L}$ for a one-layer neural network then we can say

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial A_{kl}} &= \sum_{i=1}^{m} \frac{\partial \mathcal{L}}{\partial y_i} \frac{\partial y_i}{\partial A_{kl}} \\
&= \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{\partial \mathcal{L}}{\partial y_i} \frac{\partial A_{ij}}{\partial A_{kl}} x_j \\
&= \sum_{i} \frac{\partial \mathcal{L}}{\partial y_i} \delta_{ik} x_l \\
&= \frac{\partial \mathcal{L}}{\partial y_k} x_l
\end{aligned}$$

which is a *very* useful result in machine learning.

Single Layer Network — Example 0.2.1

For a simple network with weights W, target vector $\boldsymbol{t}$, output vector $\boldsymbol{y} = W\boldsymbol{x}$ and loss function $\mathcal{L}(W) = \frac{1}{2}\|W\boldsymbol{x} - \boldsymbol{t}\|^2$, we seek to find the gradient $\boldsymbol{\nabla}_W \mathcal{L}$.

In coordinate form

$$(W\boldsymbol{x})_i - t_i = \sum_{j=1}^{n} W_{ij} x_j - t_i$$

and using the chain rule to compute

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial W_{ij}} &= \frac{\partial}{\partial W_{ij}} \left[\frac{1}{2} \left(\sum_{k=1}^{n} W_{ik} x_k - t_i \right)^2 \right] \\
&= \underbrace{\left(\sum_{k=1}^{n} W_{ik} x_k - t_i \right)}_{\text{define as } r_i} \underbrace{x_j}_{\frac{\partial}{\partial W_{ij}(\cdot)}} \\
&= ((W\boldsymbol{x})_i - t_i) x_j
\end{aligned}$$

Now we connect this the vector form to say $\boldsymbol{r} = W\boldsymbol{x} - \boldsymbol{t}$ such that

$$\frac{\partial \mathcal{L}}{\partial W_{ij}} = r_i x_j$$

$$(\boldsymbol{\nabla}_W \mathcal{L})_{ij} = (W\boldsymbol{x} - \boldsymbol{t})_i x_j$$

Then by the **outer product**

$$\boldsymbol{r}\boldsymbol{x}^T = (W\boldsymbol{x} - \boldsymbol{t})\boldsymbol{x}^\top$$
$$\boldsymbol{\nabla}_W \mathcal{L} = (W\boldsymbol{x} - \boldsymbol{t})\boldsymbol{x}^\top$$

□

Identities for Computing Gradients
$\frac{\partial}{\partial \boldsymbol{X}} \boldsymbol{f}(\boldsymbol{X})^\top = \left(\frac{\partial \boldsymbol{f}(\boldsymbol{X})}{\partial \boldsymbol{X}}^\top\right)$
$\frac{\partial}{\partial \boldsymbol{X}} \operatorname{tr}(\boldsymbol{f}(\boldsymbol{X})) = \operatorname{tr}\left(\frac{\partial \boldsymbol{f}(\boldsymbol{X})}{\partial \boldsymbol{X}}\right)$
$\frac{\partial}{\partial \boldsymbol{X}} \det(\boldsymbol{f}(\boldsymbol{X})) = \det(\boldsymbol{f}(\boldsymbol{X})) \operatorname{tr}\left(\boldsymbol{f}(\boldsymbol{X})^{-1} \frac{\partial \boldsymbol{f}(\boldsymbol{X})}{\partial \boldsymbol{X}}\right)$
$\frac{\partial}{\partial \boldsymbol{X}} \boldsymbol{f}(\boldsymbol{X})^{-1} = -\boldsymbol{f}(\boldsymbol{X})^{-1} \frac{\partial \boldsymbol{f}(\boldsymbol{X})}{\partial \boldsymbol{X}} \boldsymbol{f}(\boldsymbol{X})^{-1}$
$\frac{\partial}{\partial \boldsymbol{X}} \boldsymbol{x}^\top \boldsymbol{a} = \boldsymbol{a}^\top$
$\frac{\partial}{\partial \boldsymbol{X}} \boldsymbol{a}^\top \boldsymbol{x} = \boldsymbol{a}^\top$
$\frac{\partial}{\partial \boldsymbol{X}} \boldsymbol{a}^\top \boldsymbol{X} \boldsymbol{b} = \boldsymbol{a}\boldsymbol{b}^\top$
$\frac{\partial}{\partial \boldsymbol{X}} \boldsymbol{x}^\top \boldsymbol{A} \boldsymbol{x} = \boldsymbol{x}^\top(\boldsymbol{A} + \boldsymbol{A}^\top)$