1 Math prem

Trace is cyclic.
$trace(AB) = trace(BA)$
Dot product is related with trace in this fashion.
$< A, B >= trace(A^T B)$
$trace(ABC) = trace(BCA) = trace(CAB)$
$< AB, C >= trace((AB)^T C)$
$trace(s) = s$, if s is scaler.
Common notion of gradient form in multi-dimension input and output.

$$\Delta y =< \frac{dy}{dx}, \Delta x >= \frac{dy}{dx}^T \Delta x = trace(\frac{dy}{dx}^T \Delta x)$$

2 Neural Network

y_0 be input,$y*$ be label (desired output). $N_1, N_2, ..., N_m$ be m layers.

$$y_i = N_i(y_{i-1})$$

So, y_m is the ouput of NN (Neural Network). Let L be the loss function,

$$l = L(y_m, y*)$$

3 mse

Let $x \in \mathbf{R}^n$ be input, $y* \in \mathbf{R}^m$ be the label, $y \in \mathbf{R}^m$ be the output of mse layer.
Let $L \in \mathbf{R}$ be mean squared loss.

3.0.1 Forward pass

$$y = L(x, y*) = 1/2||x - y*||^2$$

3.1 Backward pass

$$dy := \frac{dL}{dy} = \frac{dL}{dL} = 1$$

$$y = 1/2(x - y*)^T(y_m - y*)$$

$$\begin{aligned}\Delta y &= 1/2(x + \Delta x - y*)^T(x + \Delta x - y*) - 1/2(x - y*)^T(x - y*)\\ &= 1/2((\Delta x)^T(x - y*) + (x - y*)^T\Delta x + (\Delta x)^T\Delta x)\\ &= \Delta x^T(x - y*)\\ &=< (x - y*), \Delta x >\end{aligned}$$

We got,

$$\Delta y =< (x - y*), \Delta x >$$

$$\Rightarrow \frac{dy}{dx} = (x - y*)$$

By chain rule,

$$\frac{dL}{dx} = \frac{dL}{dy}\frac{dy}{dx}$$

Putting the values,

$$\frac{dL}{dx} = (dy)(x - y*) = (x - y*)$$

$$dx := \frac{dL}{dx} = dy(x - y*)$$

3.2 Weight gradient

Since, there is no weights in mse layer, so nothing is required.

4 Linear

Let $x \in \mathbf{R}^n$ be input, $y \in \mathbf{R}^m$ be the output of linear layer. $w \in \mathbf{R}^{m\times n}$ be the weight matrix and $L \in \mathbf{R}$ be loss. $dy := \frac{dL}{dy} \in \mathbf{R}^m$ be output loss gradient.

4.1 forward pass

$$y = wx$$

4.2 backward pass

$$y = wx$$

$$\Delta y = w(x + \Delta x) - wx = w\Delta x$$

$$\Delta L =< \frac{dL}{dy}, \Delta y >$$

$$=< \frac{dL}{dy}, w\Delta x >$$

$$= \frac{dL}{dy}^T w\Delta x$$

$$= (\frac{dL}{dy}^T w)\Delta x$$

$$=< (\frac{dL}{dy}^T w)^T, \Delta x >$$

$$=< w^T \frac{dL}{dy}, \Delta x >$$

So we got the terms in required form,

$$\Delta L =< w^T \frac{dL}{dy}, \Delta x >$$

$$\frac{dL}{dx} = w^T \frac{dL}{dy}$$

$$dx := \frac{dL}{dx} = w^T \frac{dL}{dy}$$

4.3 Gradient pass

$$y = wx$$

$$\Delta y = (w + \Delta w)x - wx = \Delta wx$$

$$\Delta L =< \frac{dL}{dy}, \Delta y >$$

$$=< \frac{dL}{dy}, \Delta wx >$$

$$= \frac{dL}{dy}^T \Delta wx$$

$$= trace(\frac{dL}{dy}^T \Delta wx)$$

$$= trace(x \frac{dL}{dy}^T \Delta w)$$

$$=< (x \frac{dL}{dy}^T)^T, \Delta w >$$

$$=< \frac{dL}{dy} x^T, \Delta w >$$

we got,

$$\Delta L =< \frac{dL}{dy} x^T, \Delta w >$$

$$\Rightarrow \frac{dL}{dw} = \frac{dL}{dy} x^T$$

5 Add

Let $x \in \mathbf{R}^n$ be input, $y \in \mathbf{R}^n$ be the output of linear layer. $w \in \mathbf{R}^n$ be the weight matrix and $L \in \mathbf{R}$ be loss. $dy := \frac{dL}{dy} \in \mathbf{R}^n$ be output loss gradient.

5.1 Forward pass

$$y = x + w$$

5.2 Backward pass

$$y = x + w$$
$$\Delta y = x + \Delta x + w - (x + w) = \Delta x$$
$$\Delta L =< \frac{dL}{dy}, \Delta y >$$
$$\Delta L =< \frac{dL}{dy}, \Delta x >$$
$$\Rightarrow \frac{dL}{dx} = \frac{dL}{dy}$$

5.3 Gradient pass

$$y = x + w$$
$$\Delta y = x + \Delta w + w - (x + w) = \Delta w$$
$$\Delta L =< \frac{dL}{dy}, \Delta y >$$
$$\Delta L =< \frac{dL}{dy}, \Delta w >$$
$$\Rightarrow \frac{dL}{dw} = \frac{dL}{dy}$$

6 Convolution with infinte dimension input

Let $x \in \mathbf{R}^\infty$ be input, $y \in \mathbf{R}^\infty$ be the output of convolution layer. $w \in \mathbf{R}^\infty$ be the weight matrix (filter) and $L \in \mathbf{R}$ be loss. $dy := \frac{dL}{dy} \in \mathbf{R}^\infty$ be output loss gradient.
We take $x \in \mathbf{R}^\infty$ to ignore the mode of padding (like 'full', 'valid','same', 'custom') in convolution , as in infinte dimensional input padding of input is meaningless. It makes proof little simple and yet conveys basic idea. Proof follows even if $w \in \mathbf{R}^k$ but breaks down with finite dimensional input. Reader is encouraged to find out in below proof where it would break down in case of finit-dimensional input.

Note - All summation index below runs from $-\infty$ to ∞

6.1 Definition

- Correlation
$y = x \odot w$
$y_i := \Sigma_j x_{i+j} w_j$

- Convolution
$y = x \otimes w$
$y_i := \Sigma_j x_{i-j} w_j$

6.2 Properties

Linearity of Convolution and Correlation

$(A + B) \odot C = A \odot C + B \odot C$
$A \odot (B + C) = A \odot B + A \odot C$
$(A + B) \otimes C = A \otimes C + B \otimes C$
$A \otimes (B + C) = A \otimes B + A \otimes C$
It is very easy to verify from the definition above.

Dot product with correlation and convolution

1) $< A, B \odot C > \quad = \quad < C, B \odot A >$
2) $< A, B \otimes C > \quad = \quad < A, B \otimes A >$
3) $< A, B \odot C > \quad = \quad < B, A \otimes C >$
4) $< A, B \otimes C > \quad = \quad < B, A \odot C >$

Proof,
1)
$< A, B \odot C >= \Sigma_i A_i (B \odot C)_i$ –def of dot product

$= \Sigma_i A_i (\Sigma_p B_{i+j} C_j)$ – def of correaltion

$= \Sigma_i \Sigma_j A_i B_{i+j} C_j$

$= \Sigma_j \Sigma_i C_j B_{i+j} A_i$ – switching summation

$= \Sigma_j C_j \Sigma_i B_{i+j} A_i$

$= \Sigma_j C_j (\Sigma_i B_{i+j} A_i)$

$= \Sigma_j C_j (B \odot A)_j$ – def of correlation

$< C, B \odot A >$ – def dot product

2). Similary,
$< A, B \otimes C > \quad = \quad < A, B \otimes A >$ just interchanging correlation and convolution.

3). $< A, B \odot C > \quad = \quad \Sigma_i A_i (B \odot C)_i$ –def of dot product

$= \Sigma_i A_i (\Sigma_p B_{i+j} C_j)$ –def of correlation.

$= \Sigma_i \Sigma_p A_i B_{i+j} C_j$ –
Change of variable, putting $p = i + j$ and $j = j$, $\Rightarrow i = k - j$

$$= \Sigma_p \Sigma_p A_{p-j} B_p C_j$$
$$= \Sigma_p B_p (\Sigma_p A_{p-j} C_j)$$

$$= \Sigma_p B_p (A \otimes C)_p$$

$$< B, A \otimes C >$$

4) Similary, $\quad < A, B \otimes C > \quad = \quad < B, A \odot C >$

7 Correlation1D

$x \in \mathbf{R}^n$ is input, $y \in \mathbf{R}^m$ is output, $w \in \mathbf{R}^{k \times l}$ is filter, $L \in \mathbf{R}$ is scaler valued Loss/cost.

7.1 Forward Pass

$$y = x \odot w$$

7.2 Backward Pass

$$y = x \odot w$$

$$\Delta y = (x + \Delta x) \odot w - x \odot w$$

$$= \Delta x \odot w$$

$$\Delta L =< \frac{dL}{dy}, \Delta y >$$

$$=< \frac{dL}{dy}, \Delta x \odot w >$$

$$=< \Delta x, \frac{dL}{dy} \otimes w >$$

$$=< \frac{dL}{dy} \odot w, \Delta x >$$

So, we got things in desired form,

$$\Delta L =< \frac{dL}{dy} \otimes w, \Delta x >$$

$$\Rightarrow \frac{dL}{dx} = \frac{dL}{dy} \otimes w$$

7.3 Gradient Pass

$y = x \odot w$

$\Delta y = x \odot (w + \Delta w) - x \odot w$

$= x \odot \Delta w$

$\Delta L =< \frac{dL}{dy}, \Delta y >$ – def of Δ

$=< \frac{dL}{dy}, x \odot \Delta w >$ – puting value of Δy

$=< \Delta w, x \odot \frac{dL}{dy} >$ – switching variable, see property 6.2.1

$=< x \odot \frac{dL}{dy}, \Delta w >$

So, we got things in desired form,

$$\Delta L =< x \odot \frac{dL}{dy}, \Delta w >$$

$$\Rightarrow \frac{dL}{dw} = x \odot \frac{dL}{dy}$$