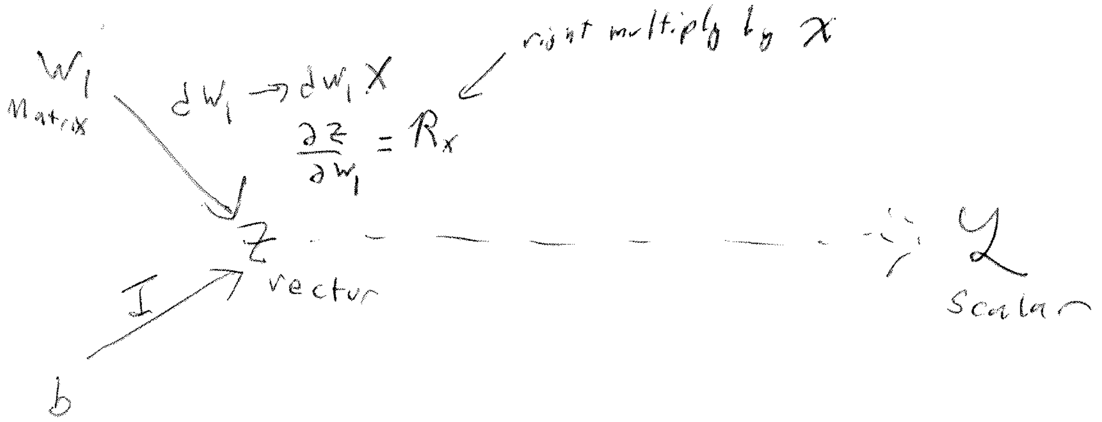
(1) March 13

CR lecture 10
Neural Network

You've seen this but it's worth going through again + in depth

x, t inputs (no variation) black in CR's diagram

Forward Pass
$$\begin{cases} z = W_1 x + b_1 \\ h = \sigma_0(z) \\ y = W_2 h + b_2 \\ \mathcal{L} = \frac{1}{2}\|y - t\|^2 \end{cases}$$



$$\frac{d\mathcal{L}}{dW_1} = \frac{\partial \mathcal{L}}{dz}\frac{\partial z}{\partial W_1}$$ ← This gets confusing

but remember $\nabla_{W_1}\mathcal{L}$ is a matrix (same shape as W_1)

$$d\mathcal{L} = \overbrace{\nabla_{W_1}\mathcal{L}}^{\text{matrix}} \cdot \overbrace{dW_1}^{\text{matrix}}$$ (directional derivative)

$$= tr\left((\nabla_{W_1}\mathcal{L})^T dW_1\right)$$ $\left(\underbrace{tr\, A^T B = A \cdot B}_{\text{math}} = \underbrace{sum(A .* B)}_{\text{Julia}}\right)$

$$d\mathcal{L} = \frac{d\mathcal{L}}{dW_1}[dW_1]$$

(2) March 13

$$\underset{\text{Matrix}}{W_1} \longrightarrow \underset{\text{vector}}{z} \longrightarrow \underset{\text{Scalar}}{\mathcal{L}}$$

Combined

$$W_1 \xrightarrow[\nabla_{W_1}\mathcal{L} \cdot dW_1]{dW_1 \to \frac{d\mathcal{L}}{dW_1}[dW_1]} \mathcal{L} \text{ scalar}$$

$$tr\left((\nabla_{W_1}\mathcal{L})^T dW_1\right)$$ ← Any nice scalar function of a matrix

$$tr \circ \left[I \otimes (\nabla_{W_1}\mathcal{L})^T\right]$$

The pieces

$$W_1 \xrightarrow[dW_1 \to dW_1 x]{} z$$

R_x ← right mult operator

$x^T \otimes I$

Why no gradient?

$$z \xrightarrow[g^T dz]{\nabla_z \mathcal{L} \cdot dz} \mathcal{L}$$

g = gradient

$g \cdot dz$

$$\underbrace{\frac{d\mathcal{L}}{dz}}_{\text{row vector}} dz$$

$$tr\left((\nabla_{W_1}\mathcal{L})^T dW_1\right) = (\nabla_z \mathcal{L})^T dW_1 x$$ cyclic prop

$$= tr\left[x(\nabla_z \mathcal{L})^T dW_1\right]$$

$$\Rightarrow (\nabla_{W_1}\mathcal{L}) = (\nabla_z \mathcal{L}) x^T$$

The reverse mode people are good at back propagating gradients

First they make the notation less daunting

$$\nabla_{any} \mathcal{L} = \overline{any}$$

pushback
Note that we go from $\bar{z} \to \bar{W}_1$

so $$\boxed{\bar{W}_1 = \bar{z} x^T}$$

Let's try another example one step away

$$\underset{\text{vector}}{z} \xrightarrow{D = \text{diag}(\sigma_o'(z))} \underset{\text{vector}}{h} \xrightarrow{\bar{h}^T} \underset{\text{scalr}}{\mathcal{L}}$$

$$d\mathcal{L} = \bar{h}^T (D\, dz) = (D \bar{h})^T dz$$

$$\boxed{\bar{z}} = D\bar{h} = \boxed{\sigma_o'(z) .* \bar{h}}$$

$$y = W_2 h + b_2$$

$$\underset{\text{vector}}{h} \xrightarrow{dy = W_2\, dh} \underset{\text{vector}}{y} \xrightarrow{\bar{y}^T} \underset{\text{scalar}}{\mathcal{L}}$$

$$\begin{aligned} \partial \mathcal{L} &= \bar{y}^T W_2\, dh \\ &= (W_2^T \bar{y})^T dh \\ &= \bar{h}^T dh \end{aligned}$$

$$\bar{h} = W_2^T \bar{y}$$

By now perhaps you can see the pattern and not follow the slow way but do it the reverse way

$$
\begin{tikzcd}
W_1 \arrow[dr, "R_x"] & & & W_2 \arrow[d, "R_h"] & \\
& z \arrow[r, "\mathrm{diag}(\sigma'(z))"] & h \arrow[r, "W_2"] & y \arrow[r, "(y-t)^T"'] & \mathcal{L} \\
b \arrow[ur, "I"'] & & & b_2 \arrow[u, "I"] &
\end{tikzcd}
$$

$$\bar{\mathcal{L}} = 1$$

$$\bar{y} = \bar{\mathcal{L}}\,(y-t) \qquad \text{(which is } y-t\text{)}$$

$$\bar{W_2} = \bar{y}\, h^T$$

$$\bar{b_2} = \bar{y}$$

$$\bar{h} = W_2^T\, \bar{y}$$

$$\bar{z} = \bar{h} \,.\!*\, \sigma'(z) \qquad \text{(same as } \mathrm{diag}^T\text{)}$$

$$\bar{W_1} = \bar{z}\, x^T \qquad (\bar{z}\, R_{x^T})$$

$$\bar{b_1} = \bar{z}$$

Why is this called an adjoint?

In general if L is a linear operator from a vector space X to Y

we look for an adjoint L^* (I may use L^T)

s.t. $\langle y, Lx \rangle = \langle L^T y, x \rangle$ for all $x \in X$, $y \in Y$.

For us $\langle -, - \rangle$ = vector or matrix dot products

e.g. $(R_A)^T = R_{A^T}$ proof

$$\langle y, Ax \rangle = y^T A x = \langle A^T y, x \rangle$$

eg $x \to v .* x$ is self-adjoint

$$\langle y, v.*x \rangle = \sum_i y_i (v_i x_i) = \langle v.*y, x \rangle$$

So you can take adjoints ~~of~~ from end to start to compute gradients

7

really it's just

$$x_1 \xrightarrow{J_1} x_2 \xrightarrow{J_2} x_3 \xrightarrow{J_3 = g^T} \text{scalar}$$

$$\Big(\underbrace{J_1^T \; J_2^T g}_{\text{reverse}}\Big)^T = \Big(g^T \underbrace{(J_2 \, J_1)}_{\text{forward}}\Big)$$

$$= (g^T J_2) J_1 \quad \text{reverse}$$

$g^T J$

vjp - vector transpose Jacobian

What is $\left(\frac{d}{dx}\right)^T$?