ECE C147 HW #3 Due: February 7, 2025

Caitlin Tang
505683631

1. Backpropagation for autoencoders

With an autoencoder, try to reconstruct the original data dimensions after some operation that reduces the data's dimensionality.
E.g. Consider $x \in \mathbb{R}^n$ and $W \in \mathbb{R}^{m \times n}$ where $m < n$. Then Wx is of lower dimensionality than x.

One way to design W s.t. Wx still contains key features of x is to minimize $\mathcal{L}$ w.r.t. W

$$\mathcal{L} = \frac{1}{2}\|W^T W x - x\|^2 \qquad \mathcal{L} = \frac{1}{2}\|f(W^T f(Wx)) - x\|^2$$

Linear Example — Nonlinear Example

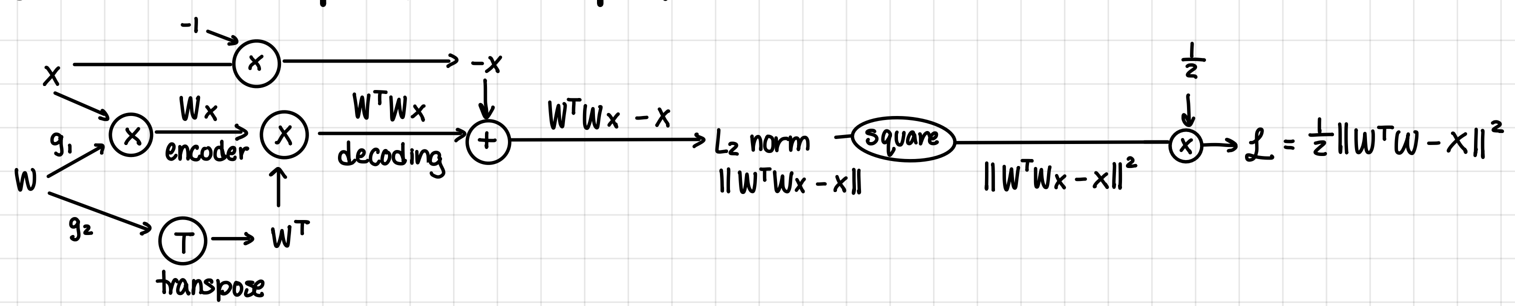
$\underbrace{(n\times m)(m\times n)}$ $\hookrightarrow W^T W x \in \mathbb{R}^n$ $(n \times 1)$

Use the linear example for the following:

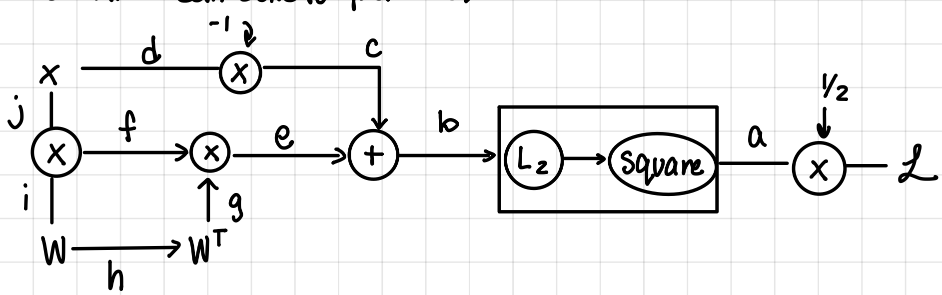
(a) Why does the minimization find a W that ought to preserve info about x

In this minimization of $\mathcal{L} = \frac{1}{2}\|W^T W x - x\|^2$, we ensure that we find a matrix W that will preserve the features of X because Wx will reduce the dimensions of x to m, but W^T will attempt to reconstruct x from the compressed representation. In other words Wx will result in an $n \times 1$ vector from a $(m \times n)(n \times 1)$ multiplication, whereas $W^T W x$ will result in an $n \times 1$ vector from a $(n \times m)(m \times n)(n \times 1)$ multiplication. If W were to be poorly chosen, important information would be lost and have a high reconstruction error. Minimizing $\mathcal{L}$ forces W to learn an optimal low-dimensional representation where it preserves key features (similar to PCA analysis)

(b) Draw the computational Graph for $\mathcal{L}$



Setup so that I can solve for part (d)



(c) In the computational graph, there should be 2 paths to W. How do we account for these two paths when calculating $\nabla_W \mathcal{L}$? Should include mathematical argument.

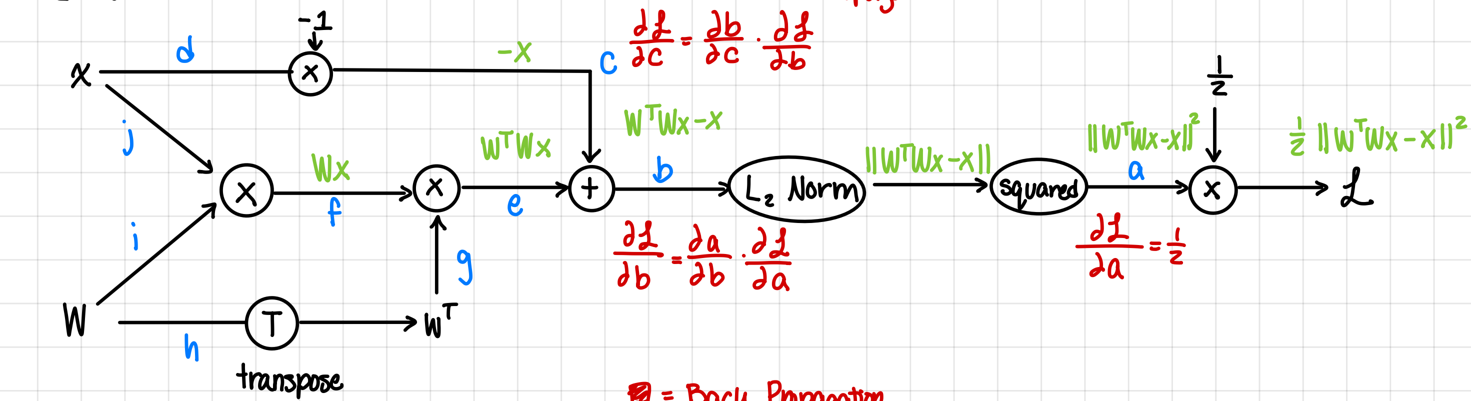
In the computational graph, the matrix W appears when W maps x to a lower dimension (Wx) and when we reconstruct ($W \xrightarrow{T} W^T \rightarrow W^T W x$). Both will ultimately converge at $W^T W x$.
Mathematically, we defined g_1 to be the path that W takes to become Wx and g_2 to be the path that W takes to get to $W \xrightarrow{T} W^T \rightarrow W^T W x$

$$\frac{\partial \mathcal{L}}{\partial W} = \frac{\partial g_1}{\partial W} \cdot \frac{\partial \mathcal{L}}{\partial g_1} + \frac{\partial g_2}{\partial W} \cdot \frac{\partial \mathcal{L}}{\partial g_2}$$

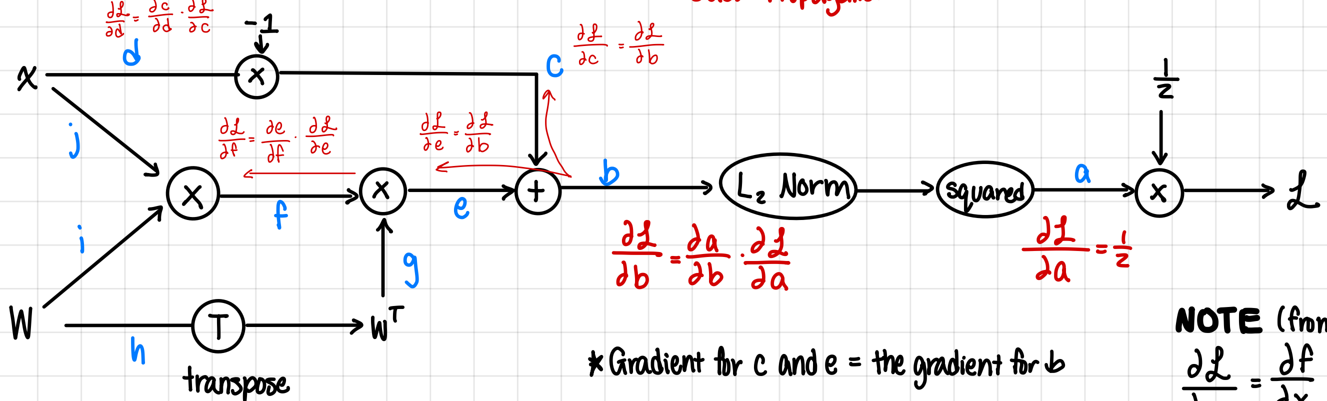
(d) Calculate the gradient $\nabla_W \mathcal{L}$

$\mathcal{L} = \frac{1}{2}\|W^T W x - x\|^2$

■ = Forward Propagation
■ = Back Propagation



■ = Back Propagation



NOTE (from Lecture 6)

$$\frac{\partial \mathcal{L}}{\partial x} = \underbrace{\frac{\partial f}{\partial x}}_{\text{local}} \underbrace{\frac{\partial \mathcal{L}}{\partial f}}_{\text{upstream}}$$

*Gradient for c and e = the gradient for b

local gradient   upstream gradient

$$\frac{\partial \mathcal{L}}{\partial a} = \frac{1}{2} \qquad \frac{\partial \mathcal{L}}{\partial b} = \frac{\partial a}{\partial b} \cdot \frac{\partial \mathcal{L}}{\partial a} = 2b \cdot \frac{1}{2} = b = W^T W x - x$$

value comes from forward propagation

$\left(a = b^T b \quad \therefore \frac{\partial a}{\partial b} = 2b\right)$

$$\frac{\partial \mathcal{L}}{\partial c} = \frac{\partial \mathcal{L}}{\partial b} = b = W^T W x - x \quad \text{and} \quad \frac{\partial \mathcal{L}}{\partial e} = \frac{\partial \mathcal{L}}{\partial b} = b = W^T W x - x$$

NOTE: $\frac{\partial \mathcal{L}}{\partial c}$ and $\frac{\partial \mathcal{L}}{\partial e}$ come from ⊕ ∴ distributed gradient

$$\frac{\partial \mathcal{L}}{\partial d} = \frac{\partial c}{\partial d} \cdot \frac{\partial \mathcal{L}}{\partial c} = -1 \cdot b = -b = -(W^T W x - x)$$

$$\frac{\partial \mathcal{L}}{\partial f} = \frac{\partial e}{\partial f} \cdot \frac{\partial \mathcal{L}}{\partial e} = \frac{\partial e}{\partial f} \cdot b = \frac{\partial W^T W x}{\partial W x} \cdot b = (W^T)^T b = Wb = W(W^T W x - x)$$

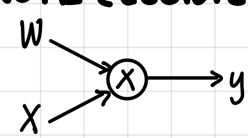
$$\frac{\partial \mathcal{L}}{\partial g} = \frac{\partial e}{\partial g} \times \frac{\partial \mathcal{L}}{\partial e} = b \cdot (Wx)^T = b x^T W^T = (W^T W x - x) x^T W^T$$

Uses the trick in class

$$\frac{\partial \mathcal{L}}{\partial h} = \left(\frac{\partial \mathcal{L}}{\partial g}\right)^T = (b x^T W^T)^T = W x b^T = W x (W^T W x - x)^T$$

$$\frac{\partial \mathcal{L}}{\partial i} = \frac{\partial f}{\partial i} \cdot \frac{\partial \mathcal{L}}{\partial f} = W b x^T$$

NOTE (Lecture 7)



$$\frac{\partial \mathcal{L}}{\partial x} = W^T \frac{\partial \mathcal{L}}{\partial y}$$

$$\frac{\partial \mathcal{L}}{\partial W} = \frac{\partial \mathcal{L}}{\partial y} x^T$$

$b = W^T W x - x$

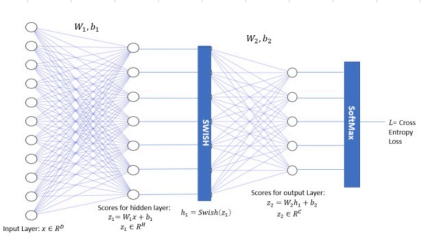
$$\frac{\partial \mathcal{L}}{\partial W} = \frac{\partial \mathcal{L}}{\partial h} + \frac{\partial \mathcal{L}}{\partial i} = W x b^T + W b x^T = W x (W^T W x - x) + W (W^T W x - x) x^T$$

Problem #2 : I am a C147 Student

Problem #3: NNDL

D = # of neurons in input layer, H = # of neurons in the hidden layer, C = # of neurons in the oupt ($C=7$)

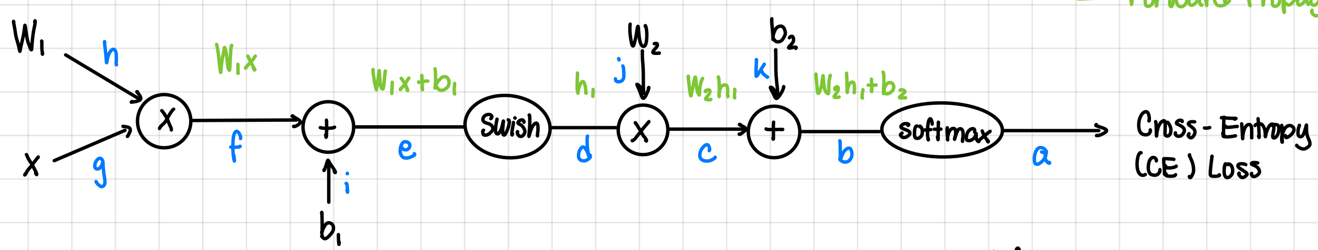
Swish activation function $\quad swish(k) = \dfrac{k}{1+e^{-k}} = k\sigma(k)$ where $\sigma(k)$ is sigmoid activation function



Hidden Layer: $z_1 = W_1 x + b_1 \quad z_1 \in \mathbb{R}^H \qquad h_1 = Swish(z_1) \qquad z_2 = W_2 h_1 + b_2 \quad z_2 \in \mathbb{R}^C$

(a) Draw the computational graph for the 2-layer FC Net

= Forward Propagation



(b) Compute $\nabla_W \mathcal{L}$ $\nabla_{b_2}\mathcal{L}$ (For the gradient computations you can keep it as $\frac{\partial \mathcal{L}}{\partial z_2}$)

Cross-entropy Loss $= \mathcal{L} = -\sum y_i \log \hat{y}_i$
where $\hat{y}_i$ is the predicted value

$$\hat{y}_i = \frac{e^{z_{2,i}}}{\sum_j e^{z_{2,j}}} \quad \therefore \mathcal{L} = -\sum y_i \log\left(\frac{e^{z_{2,i}}}{\sum_j e^{z_{2,j}}}\right) = -\sum y_i (z_{2,i} - z_{2,j})$$

$$\frac{\partial \mathcal{L}}{\partial z_{2,i}} = \frac{\partial}{\partial z_{2,i}}\left(-\sum y_i (z_{2,i} - z_{2,j})\right) = \hat{y}_i - y_i$$

$$\boxed{\therefore \frac{\partial \mathcal{L}}{\partial z_2} = \hat{y} - y}$$

$$\frac{\partial \mathcal{L}}{\partial W_2} = \frac{\partial \mathcal{L}}{\partial z_2} \cdot \frac{\partial z_2}{\partial W_2}$$ we know that $\frac{\partial \mathcal{L}}{\partial z_2} = \hat{y} - y$ and $\frac{\partial z_2}{\partial W_2} = \frac{\partial W_2 h_1 + b_2}{\partial W_2} = h_1^T$

$\hookrightarrow \nabla_{W_2}\mathcal{L} = (\hat{y} - y)h_1^T$

$$\nabla_{b_2}\mathcal{L} = \frac{\partial \mathcal{L}}{\partial b_2} = \frac{\partial \mathcal{L}}{\partial z_2} = (\hat{y} - y)$$

$$\boxed{\begin{array}{l} \text{Results} \\ \nabla_{W_2}\mathcal{L} = (\hat{y}-y)h_1^T \\ \nabla_{b_2}\mathcal{L} = (\hat{y}-y) \end{array}}$$

y comes from the Jacobian

$$J = \frac{\partial y}{\partial x} = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_m} \\ \frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & \cdots & \frac{\partial y_2}{\partial x_m} \\ \vdots & & & \\ \frac{\partial y_n}{\partial x_1} & \frac{\partial y_n}{\partial x_2} & \cdots & \frac{\partial y_n}{\partial x_m} \end{bmatrix}$$

(c) Compute $\nabla_{W_1}\mathcal{L}$, $\nabla_{b_1}\mathcal{L}$

Calculating $\nabla W_1 \mathcal{L}$ requires $\frac{\partial}{\partial z_1}$

$h_1 = \text{Swish}(z_1) \rightarrow \text{Swish}(x) = x\sigma(x)$ and $\sigma(x) = \frac{1}{1+e^{-z}}$

$$\frac{\partial\, \text{Swish}(x)}{\partial x} = \sigma(x) + x\sigma(x)(1-\sigma(x))$$

Replacing x with z_1 we then have $\sigma(z_1) + z_1\sigma(z_1)(1-\sigma(z_1))$

$$\frac{\partial \mathcal{L}}{\partial z_1} = \frac{\partial \mathcal{L}}{\partial h_1} \cdot \frac{\partial h_1}{\partial z_1} = (W_2^T(\hat{y}-y)) \odot \frac{\partial h_1}{\partial z_1} = ((W_2^T(\hat{y}-y)) \odot [\sigma(z_1) + z_1\sigma(z_1)(1-\sigma(z_1))])$$

we use $\frac{\partial \mathcal{L}}{\partial z_1}$ to find $\frac{\partial \mathcal{L}}{\partial W_1}$ and $\frac{\partial \mathcal{L}}{\partial b_1}$

$$\nabla_{W_1}\mathcal{L} = \frac{\partial \mathcal{L}}{\partial z_1} \cdot \frac{\partial z_1}{\partial W_1} = (\nabla_{z_1}\mathcal{L})(x^T) = [(W_2^T(\hat{y}-y)) \odot (\sigma(z_1) + z_1\sigma(z_1)(1-\sigma(z_1)))]$$

Comes from the trick in class $\frac{\partial z_1}{\partial W_1} = \frac{\partial W_1 x + b_1}{\partial W_1} = x^T$

$$\nabla_{b_1}\mathcal{L} = \nabla_{z_1}\mathcal{L} = (W_2^T(\hat{y}-y)) \odot ($$