CS 224N Assignment #1.

1. Softmax

(a) $$\text{softmax}(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

$$\text{softmax}(x_i+c) = \frac{e^{x_i+c}}{\sum_j e^{x_j+c}} = \frac{e^c \cdot e^{x_i}}{e^c \cdot \sum_j e^{x_j}} = \frac{e^{x_i}}{\sum_j e^{x_j}} = \text{softmax}(x_i).$$

(b). CODE.

2. Neural Network Basics.

(a) $$\sigma(x) = \frac{1}{1+e^{-x}} \Rightarrow \frac{d}{dx}\sigma(x) = \frac{-(-e^{-x})}{(1+e^{-x})^2} = (1-\sigma x)\,\sigma x.$$

(b)

$$\hat{y} = \text{softmax}(\theta)$$

$$\frac{\partial CE(y,\hat{y})}{\partial \theta} = \frac{\partial CE(y,\hat{y})}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial \theta}.$$

The first step is to calculate the derivative of softmax() function.

$$\hat{y_i} = \frac{e^{\theta_i}}{\sum_k e^{\theta_k}} \qquad \frac{\partial \hat{y_i}}{\partial \theta_i} = \frac{\partial}{\partial \theta_i}\frac{1}{1+Ae^{-\theta_i}} \quad (\text{where } A = \sum_k e^{\theta_k} - e^{\theta_i}).$$

$$= \frac{1}{(1+Ae^{-\theta_i})}\left(1 - \frac{1}{1+Ae^{-\theta_i}}\right) = \hat{y_i}(1-\hat{y_i}).$$

$$\frac{\partial \hat{y_i}}{\partial \theta_j} = \frac{\partial}{\partial \theta_j}\frac{e^{\theta_i}}{A+e^{\theta_j}} \quad (\text{where } A = \sum_k e^{\theta_k} - e^{\theta_j}).$$

$$= \frac{e^{\theta_i}}{A+e^{\theta_j}} \cdot \frac{-e^{\theta_j}}{A+e^{\theta_j}} = -\hat{y_i}\cdot\hat{y_j}$$

$$\frac{\partial CE(y,\hat{y})}{\partial \theta_i} = -\sum_k y_k \frac{\partial \log \hat{y_k}}{\partial \theta_i} = -\sum_k y_k \cdot \frac{1}{\hat{y_k}}\frac{\partial \hat{y_k}}{\partial \theta_i}$$

$$= -(y_i \cdot \hat{y_i} \cdot \frac{1}{\hat{y_i}}(1-\hat{y_i}) + \sum_{k\neq i} y_k \frac{1}{\hat{y_k}}(-\hat{y_i}\hat{y_k})) \qquad \Rightarrow \frac{\partial CE(y,\hat{y})}{\partial \theta} = \hat{y} - y$$

$$= -y_i + \sum_k \hat{y_i} y_k = \hat{y_i} - y_i$$

(c).

$$\frac{\partial J}{\partial x} = \frac{\partial CE(y,\hat{y})}{\partial (hW_2+b_2)} \cdot \frac{\partial (hW_2+b_2)}{\partial h} \cdot \frac{\partial h}{\partial (xW_1+b_1)} \cdot \frac{\partial (xW_1+b_1)}{\partial x}$$

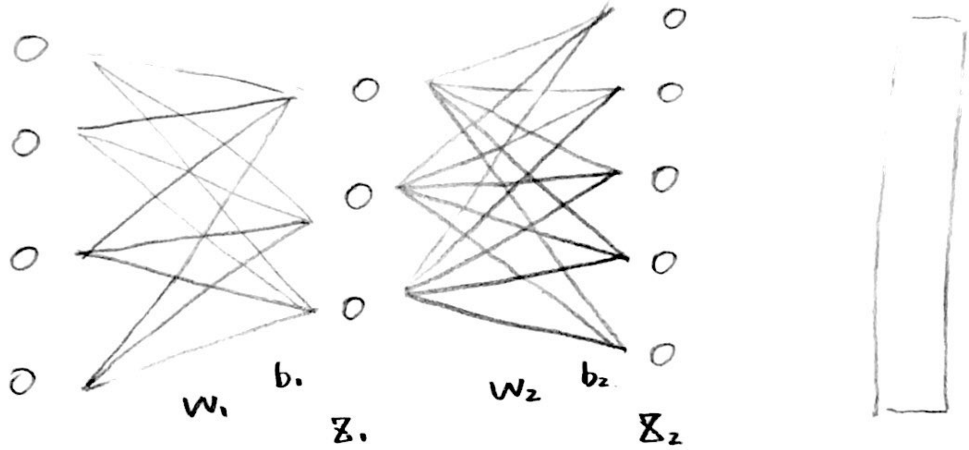
$$= (\hat{y}-y)\, W_2^T \circ \sigma'(z_1)\, W_1^T$$

(d) $W_1 \sim H \times D_x$ $\quad b_1 \sim H$ $\quad W_2 \sim D_y \times H$. $\quad b_2 \sim D_y$

total : $H \times D_x + H + D_y \times H + D_y$

(e) ~ (g) Coding Part.

a few more details on the derivatives of Neural Networks



Y ~ actual result

Input Layer

X

A_1 (activation function), A_2

$Z_1 = W_1 X + b_1 \qquad A_1 = A_1(Z_1)$

$Z_2 = W_2 A_1 + b_2 \qquad A_2 = A_2(Z_2) = \hat{Y}$ (predicted result).

Derivatives !!

$dZ_2 = \hat{Y} - Y$ ☆ (as proved in (b))

$dW_2 = dZ_2 \cdot A_1^T$ or $A_1^T \cdot dZ_2$ (depends on how you write it).

$db_2 = dZ_2.\text{sum}()$ ←

$dZ_1 = W_2^T \cdot dZ_2 \,/\, dZ_2 W_2^T \; * \; A'(Z_1)$

$dW_1 = dZ_1 \cdot X^T \,/\, X^T dZ_1$

$db_1 = dZ_1.\text{sum}()$ ← careful for the axis (= 0/1)

3. word2vec

(a) skip gram - obtain center word from surroundings

~~first, derivative of the softmax form is:~~

$$\hat{y_o} = \frac{e^{u_o^T v_c}}{\sum_w e^{u_w^T v_c}} \qquad \frac{\partial \hat{y_o}}{\partial v_c} = \frac{u_o e^{u_o^T v_c} (\sum_w e^{u_w^T v_c}) - e^{u_o^T v_c} (\sum_w u_w e^{u_w^T v_c})}{(\sum_w e^{u_w^T v_c})^2}$$

$$= \hat{y_o} \frac{\sum_w (u_o - u_w) \cdot e^{u_w^T v_c}}{\sum_w e^{u_w^T v_c}}$$

$$\frac{\partial CE(y, \hat{y})}{\partial v_c} = - \sum_k y_k \frac{\partial \log \hat{y_k}}{\partial v_c}$$

$$= - \sum_k y_k \cdot \frac{1}{\hat{y_k}} \cdot \frac{\partial \hat{y_k}}{\partial v_c}$$

$$= - \sum_k y_k \cdot \frac{1}{\hat{y_k}} \cdot \hat{y_k} \cdot \sum_w (u_o^T - u_w^T)$$

from previous section, we know that: (for classical softmax() functions)

$$\frac{\partial CE(y, \hat{y})}{\partial \theta} = \hat{y} - y.$$

set $\theta' = U^T \theta$, so for word2vec loss functions, it will be:

$$\frac{\partial CE(y, \hat{y})}{\partial \theta'} = \hat{y} - y$$

Thus.

$$\frac{\partial CE(y, \hat{y})}{\partial \theta} = \sum_i (\hat{y_i} - y_i) u_i^T = U(\hat{y} - y)$$

(b) first to calculate its derivative of softmax form function

$$\frac{\partial \hat{y_o}}{\partial u_o} = \frac{\partial}{\partial u_o} \frac{1}{1 + constant \cdot e^{-u_o^T v_c}} \quad (\text{where } constant = \sum_{w \neq o} e^{u_w^T \cdot v_c}) = \hat{y_o}(1 - \hat{y_o}) \cdot v_c$$

$$\frac{\partial \hat{y_o}}{\partial u_{w (w \neq o)}} = \cdots = - \hat{y_o} \hat{y_w} \cdot v_c$$

$$\frac{\partial CE(y, \hat{y})}{\partial u_w} = - \sum_k y_k \frac{1}{\hat{y_k}} \frac{\partial \hat{y_k}}{\partial u_w} = \begin{cases} (\hat{y_w} - 1) \cdot v_c & w = o \\ \hat{y_w} \cdot v_c & w \neq o \end{cases}$$

(c) $$\frac{\partial J}{\partial v_c} = -\frac{1}{\sigma(u_o^T v_c)}\sigma'(u_o^T v_c)\, u_o - \sum_{k=1}^{K}\frac{1}{\sigma(-u_k^T v_c)}\sigma'(-u_k^T v_c)\cdot(-u_k)$$

$$= (\sigma(u_o^T v_c) - 1)\, u_o - \sum_{k=1}^{K}(\sigma(-u_k^T v_c) - 1)\, u_k$$

$$\frac{\partial J}{\partial u_o} = -\frac{1}{\sigma(u_o^T v_c)}\sigma'(u_o^T v_c)\, v_c = [\sigma(u_o^T v_c) - 1]\, v_c$$

$$\frac{\partial J}{\partial u_k} = -(\sigma(-u_k^T v_c) - 1)\, v_c$$

don't know the reason why it execute more quickly...

(d)

From part (b) & (c), we already know how to derive the derivation of single cost functi

(i) for skip-gram

$$\frac{\partial J_{skip\text{-}gram}(w_{t-m \dots t+m})}{\partial U} = \sum_{-m \le j \le m,\, j \ne 0}\frac{\partial F(w_{t+j}, v_c)}{\partial U}$$

$$\frac{\partial J_{skip\text{-}gram}(w_{t-m \dots t+m})}{\partial v_c} = \sum_{-m \le j \le m,\, j \ne 0}\frac{\partial F(w_{t+j}, v_c)}{\partial v_c}$$

$$\frac{\partial J_{skip\text{-}gram}(w_{t-m \dots t+m})}{\partial v_{w_{t+j}}} = 0 \quad (\text{for all } j \ne c).$$

for CBOW,

just switch the "$\sum_j \partial F(w_{t+j}, v_c)$" into "$\partial F(w_t, \hat{v})$". and we've done