a) The cross-entropy loss is defined by $L_{CE}(y,\hat{y}) = -\sum_{w \in Vocab} y_w \log(\hat{y}_w)$.
Since y_w is a vector full of zeros except for the correct word where it is 1, we have $y_w = \begin{cases} 0 & \text{if } w \neq o \\ 1 & \text{if } w = o \end{cases}$, so all the terms in the sum are equal to 0, except the term for $y = o$. So we are left with: $-\sum_{y \in Vocab} y_w \log(\hat{y}_w) = -\log(\hat{y}_o)$

b. (i)

$$\frac{\partial J_{naive\text{-}softmax}}{\partial v_c}(v_c, o, U) = \frac{\partial}{\partial v_c}\left(-\log\left[\frac{\exp(u_o^T v_c)}{\sum_{w \in Vocab} \exp(u_w^T v_c)}\right]\right)$$

$$= \frac{\partial}{\partial v_c}\left(-u_o^T v_c + \log\left(\sum_{w \in Vocab} \exp(u_w^T v_c)\right)\right)$$

$$= -u_o + \frac{\sum_{w \in Vocab} u_w \exp(u_w^T v_c)}{\sum_{w \in Vocab} \exp(u_w^T v_c)}$$

$$= -u_o + \sum_{w \in Vocab} u_w \frac{\exp(u_w^T v_c)}{\sum_{z \in Vocab} \exp(u_z^T v_c)}$$

$$= -u_o + \sum_{w \in Vocab} u_w \hat{y}_w$$

$$= U^T(\hat{y} - y)$$

(ii) The gradient computed is equal to zero when $\hat{y} - y \in Ker(U)$

(iii) The gradient is the difference expected - observed. Therefore, by substracting this vector from the vector v_c, we perform a gradient descent which will bring v_c closer to it's observed value and not the current expected value

(iv) Let's say we have $x = \alpha y$ for $\alpha \in R^*$.
Then $\|x\|_2 = |\alpha| \|y\|_2$ and so $\frac{x}{\|x\|_2} = \alpha \frac{y}{\|x\|_2} = \frac{\alpha}{|\alpha|} \frac{y}{\|y\|_2} = \text{sign}(\alpha) \frac{y}{\|y\|_2}$

with $\text{sign}(x) = \begin{cases} +1 & \text{if } x > 0 \\ -1 & \text{if } x < 0 \end{cases}$.

This means that by normalizing the vectors we can now only have values of α such that $|\alpha| = 1$ (so +1 and -1). We lose the information regarding the magnitude of α.

c) $$\frac{\partial J_{\text{naive softmax}}}{\partial u_w}(v_c, o, U) = \frac{\partial}{\partial u_w}\left(-u_o v_c + \log \sum_{z \in \text{Vocab}} \exp(u_z^T v_c)\right)$$

$$= \frac{\partial}{\partial u_w}(-u_o v_c) + v_c \frac{\exp(u_w^T v_c)}{\sum_{y \in \text{Vocab}} \exp(u_y^T v_c)}$$

$$= \frac{\partial}{\partial u_w}(-u_o v_c) + v_c \hat{y}_w$$

So if $w \neq 0$: $\frac{\partial J_{\text{naive softmax}}}{\partial u_w}(v_c, o, U) = v_c \hat{y}_w$

And if $w = 0$: $\frac{\partial J_{\text{naive softmax}}}{\partial u_o}(v_c, o, U) = -v_c + v_c \hat{y}_o = v_c(\hat{y}_o - y_o)$

Since for $w \neq 0$, $y_w = 0$, we can conclude that in any cases

$$\boxed{\frac{\partial J_{\text{naive softmax}}}{\partial u_w}(v_c, o, U) = v_c(\hat{y}_w - y_w)}$$

d) The derivative of $J_{\text{naive-softmax}}$ with respect to the matrix U is simply the matrix which columns are the derivatives of J with respect to the column of U.

So, $$\frac{\partial J_{\text{naive-softmax}}}{\partial U}(v_c, o, U) = \left[\frac{\partial J_{\text{naive-softmax}}}{\partial u_1}(v_c, o, U), \dots, \frac{\partial J_{\text{naive-softmax}}}{\partial u_{|\text{vocab}|}}(v_c, o, U)\right]$$

e) We can rewrite f as $f(x) = \max(x, \alpha x) = \begin{cases} x & \text{if } x > 0 \\ \alpha x & \text{if } x < 0 \end{cases}$

f is differentiable on $\mathbb{R}^*$ and so:

$$\forall x \in \mathbb{R}^* \quad f'(x) = \begin{cases} 1 & \text{if } x > 0 \\ \alpha & \text{if } x < 0 \end{cases}$$

f) σ is differentiable on $\mathbb{R}$:

$$\forall x \in \mathbb{R} \quad \sigma'(x) = \frac{e^{-x}}{(1+e^{-x})^2} = \frac{e^{-x}}{1+e^{-x}} \cdot \frac{1}{1+e^{-x}} = \left(1 - \frac{1}{1+e^{-x}}\right)\left(\frac{1}{1+e^{-x}}\right) = \sigma(x)(1-\sigma(x))$$

So: $\boxed{\forall x \in \mathbb{R} \quad \sigma'(x) = \sigma(x)(1-\sigma(x))}$

g) (i) $$\frac{\partial J_{neg\text{-}sample}}{\partial v_c}(v_c, o, U) = \frac{\partial}{\partial v_c}\left(-\sum_s \log\left[\sigma(-u_{w_s}^T v_c)\right]\right) - \frac{\partial}{\partial v_c}\left(\log\left[\sigma(u_o^T v_c)\right]\right)$$

$$= -\sum_s \frac{-u_{w_s}\sigma'(-u_{w_s}^T v_c)}{\sigma(-u_{w_s}^T v_c)} - u_o \frac{\sigma'(u_o^T v_c)}{\sigma(u_o^T v_c)}$$

$$= \boxed{\sum_s u_{w_s}\left(1-\sigma(-u_{w_s}^T v_c)\right) - u_o\left(1-\sigma(u_o^T v_c)\right)}$$

$$\frac{\partial J_{neg\ sample}}{\partial u_o}(v_c, o, U) = 0 - \frac{\partial}{\partial u_o}\left(\log\left[\sigma(u_o^T v_c)\right]\right) = \boxed{-v_c\left(1-\sigma(u_o^T v_c)\right)}$$

$$\frac{\partial J_{neg\ sample}}{\partial u_{w_s}}(v_c, o, U) = \frac{\partial}{\partial u_{w_s}}\left(-\log\left[\sigma(-u_{w_s}^T v_c)\right]\right) + 0 = \boxed{+v_c\left(1-\sigma(-u_{w_s}^T v_c)\right)}$$

(ii) We need to store $1-\sigma(u_o^T v_c)$ as it is used in $\frac{\partial J}{\partial v_c}$ and $\frac{\partial J}{\partial u_o}$, as well $1-\sigma(-u_{w_s}^T v_c)$ for all $s \in [[1, K]]$ as they are used in $\frac{\partial J}{\partial v_c}$ and $\frac{\partial J}{\partial u_{w_s}}$.

Therefore, we should compute and store: $1-\sigma(U_{o,\{w_1,\dots,w_K\}} v_c)$

(iii) • This loss function consists of $K+1$ vector multiplications and evaluations of σ and log. Let's note d the dimension of u_o. Then, the negative sample loss has a computational complexity of $O(Kd)$.

• For the naive softmax loss, there are $|vocab|$ vector multiplications, so the complexity is in $O(|vocab|\, d)$ and $|vocab| \gg K$.

h) We will simply reuse the previous gradient computation and separate the sum for $\underline{w_t = w_s}$ and $\underline{w_t \neq w_s}$.

$$\frac{\partial J_{neg\text{-}sample}}{\partial U_{w_s}}(v_c, o, U) = \frac{\partial}{\partial U_{w_s}}\left(-\log\left[\sigma(u_o^T v_c)\right]\right) - \frac{\partial}{\partial U_{w_s}}\left(\sum_{\substack{t \\ w_t = w_s}} \log\left[\sigma(-u_{w_t}^T v_c)\right]\right) - \frac{\partial}{\partial U_{w_s}}\left(\sum_{\substack{t \\ w_t \neq w_s}} \log\left[\sigma(-u_{w_t}^T v_c)\right]\right)$$

$$= 0 + \sum_{\substack{t \\ w_t = w_s}} v_c\left(1-\sigma(-U_{w_s}^T v_c)\right) + 0$$

So $$\boxed{\frac{\partial J_{neg\text{-}sample}}{\partial U_{w_s}}(v_c, o, U) = \sum_{\substack{t \\ w_t = w_s}} v_c\left(1-\sigma(-U_{w_s}^T v_c)\right)}$$

I wrote $\sigma(-U_{w_s}^T v_c)$ and not $\sigma(-U_{w_t}^T v_c)$ as $w_t = w_s$

i) (i) $\dfrac{\partial J_{skip\text{-}gram}(v_c, w_{t-m}, \ldots, w_{t+m}, U)}{\partial U} = \sum_{\substack{-m \leq j \leq m \\ j \neq 0}} \dfrac{\partial J}{\partial U}(v_c, w_{t+j}, U)$

(ii) $\dfrac{\partial J_{skip\text{-}gram}(v_c, w_{t-m}, \ldots, w_{t+m}, U)}{\partial v_c} = \sum_{\substack{-m \leq j \leq m \\ j \neq 0}} \dfrac{\partial J}{\partial v_c}(v_c, w_{t+j}, U)$

(iii) $\dfrac{\partial J_{skip\text{-}gram}(v_c, w_{t-m}, \ldots, w_{t+m}, U)}{\partial v_w} = 0$ for $\underline{w \neq c}$

Coding c) We can observe some relevant clusters such as "woman", "female" and "man". However we could have expected "male" to be part of that cluster which is not the case. Another relevant cluster is "amazing", "wonderful", "boring" and "great".
There is one outstanding bias: "queen" and "dumb" are clustered together but "king" is not part of that cluster. This illustrates some bias in the training data.