Q1.

a. The rate of change of *f* as a function of *x*

b. $f'(x)g(x)g'(x)$

c. $f'(x)g(x) + f(x)g'(x)$

d. $a^x \ln(a)$

e. $10x^9 - 16x^7 - \frac{8}{x^3}$

Q2.

a. $f'(x) = -(1 + e^{-x})^{-2}(-e^{-x})$

$= \frac{e^{-x}}{(1+e^{-x})^{-2}}$

$= \frac{(1+e^{-x})-1}{(1+e^{-x})^2}$

$= \frac{1}{(1+e^{-x})} - \frac{1}{(1+e^{-x})^2}$

$= f(x)\big(1 - f(x)\big)$

b. $g'(x) = \frac{(e^x+e^{-x})^2 - (e^x-e^{-x})^2}{(e^x+e^{-x})^2}$ (quotient rule)

$= 1 - \frac{(e^x-e^{-x})^2}{(e^x+e^{-x})^2}$

$= 1 - g^2(x)$

c. $g(x) = \frac{e^x-e^{-x}}{e^x+e^{-x}}$

$= \frac{e^x}{e^x+e^{-x}} - \frac{e^{-x}}{e^x+e^{-x}}$

$= \frac{1}{1+e^{-2x}} - \frac{e^{-2x}}{1+e^{-2x}}$

$= \frac{2}{1+e^{-2x}} - \frac{1+e^{-2x}}{1+e^{-2x}}$

$= 2f(2x) - 1$

Q3.

a.

b. The rate of change of *f* in the direction of *x*

c. $\nabla f(x, y, \ldots) = \langle \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \ldots \rangle$

That is, a vector containing each partial derivative of *f* at point *z*

d. $\frac{\partial z}{\partial t} = \frac{\dot{\partial} z}{\partial x} \frac{\partial x}{\partial t} + \frac{\partial z}{\partial y} . \frac{\partial y}{\partial t}$

e. $f'_x(x, y) = 3x^2 + 6xy + 2$

$f'_y(x, y) = 3x^2 + 3y^2$

f. w_i

g. $\frac{\partial f}{\partial z} = f(z)\big(1 - f(z)\big)$ (as in Q2(a))

$\frac{\partial f}{\partial w_i} = \frac{\dot{\partial} f}{\partial z} \frac{\partial z}{\partial w_i} = w_i f(z)\big(1 - f(z)\big)$

h. $\frac{\partial E}{\partial w_i} = \frac{\dot{\partial} E}{\partial z} \frac{\partial z}{\partial w_i}$

$= \big(t - f(z)\big)\big(-f'(z)\big)(w_i)$

$= -(t - f(z))(f(z))(1 - f(z))(w_i)$

Q4.

a. *x* is a vector; *y* is a vector of the same size as *x* with values that sum to 1

b. Softmax is used in the final layer, to select the predicted output.
c. The softmax function is a generalization of the sigmoid function. It takes a vector of any size as its input, whereas the sigmoid function takes scalar input.
d. Softmax is an alternative to the argmax function which is smooth (infinitely differentiable). Softmax needs to be used for training, because it is smooth and allows the probability for each output class to be determined. For decoding, either function can be used, but argmax is often used because we only need the most probable class.
e. softmax(x) = [.0861, .234, .636, .0116, .000580, .0317]
argmax(x) = [0, 0, 1, 0, 0, 0]