In order to compute the update rule for regression problems in neural networks, we have to use the mean squared error loss function as against log loss which is used for classification.

We have that;

$z_1 = w_1 x + b_1$

$a_1 = g(z_1)$

$z_2 = w_2 a_1 + b_2$

$a_2 = g(z_2)$

and $L(y, \hat{y}) = \frac{1}{m} \sum (y - \hat{y})^2$

In order to update the weights and biases at the output and hidden layer, we propagate the gradients backwards from the output to the input layer.

Therefore; we compute $\frac{\partial L}{\partial w_2}$, $\frac{\partial L}{\partial b_2}$, $\frac{\partial L}{\partial w_1}$ and $\frac{\partial L}{\partial b_1}$

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2}$$

We know that $\hat{y} = a_2 = g(z_2) = w_2 a_1 + b_2$

So, $\frac{\partial \hat{y}}{\partial w_2} = \frac{\partial}{\partial w_2}(w_2 a_1 + b_2) = a_1$

$$\frac{\partial L}{\partial \hat{y}} = \frac{\partial}{\partial \hat{y}}\left[\frac{1}{m}(y - \hat{y})^2\right] = \frac{1}{m}\frac{\partial}{\partial \hat{y}}\left[(y - \hat{y})^2\right]$$

Let $u = y - \hat{y}$ and $L = u^2$

$\frac{\partial u}{\partial \hat{y}} = -1$ $\qquad \frac{\partial L}{\partial u} = 2u = 2(y - \hat{y})$

Therefore;

$$\frac{\partial L}{\partial \hat{y}} = \frac{1}{m}\frac{\partial u}{\partial \hat{y}} \cdot \frac{\partial L}{\partial u} = \frac{-2}{m}(y - \hat{y})$$

Hence, the gradient is;

$$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w_2} = -\frac{2}{m}(y-\hat{y})\, a_1$$

$$\frac{\partial L}{\partial b_2} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial b_2}$$

$$\frac{\partial \hat{y}}{\partial b_2} = \frac{\partial}{\partial b_2}\left[w_2 a_1 + b_2\right] = 1$$

Therefore,

$$\frac{\partial L}{\partial b_2} = -\frac{2}{m}(y-\hat{y})$$

Also,

$$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial a_1} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z_1}{\partial w_1}$$

$$\frac{\partial L}{\partial \hat{y}} = -\frac{2}{m}(y-\hat{y})$$

$$\frac{\partial \hat{y}}{\partial a_1} = \frac{\partial}{\partial a_1}\left[w_2 a_1 + b_2\right] = w_2$$

$$\frac{\partial a_1}{\partial z_1} = \frac{\partial}{\partial z_1}(g(z_1)) = g'(z_1)$$

$$\frac{\partial z_1}{\partial w_1} = \frac{\partial}{\partial w_1}\left[w_1 x + b_1\right] = x$$

Hence,

$$\frac{\partial L}{\partial w_1} = -\frac{2}{m}(y-\hat{y})\, w_2\, g'(z_1)\, x$$

And,

$$\frac{\partial L}{\partial b_1} = \frac{\partial L}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial a_1} \cdot \frac{\partial a_1}{\partial z_1} \cdot \frac{\partial z_1}{\partial b_1}$$

$$= -\frac{2}{m}(y-\hat{y})\, w_2\, g'(z_1)$$

Using the gradients computed, we can update our weights and biases by applying the gradient descent algorithm.

Therefore,

$$w_2 := w_2 - \alpha \frac{\partial L}{\partial w_2}$$

$$b_2 := b_2 - \alpha \frac{\partial L}{\partial b_2}$$

$$w_1 := w_1 - \alpha \frac{\partial L}{\partial w_1}$$

$$b_1 := b_1 - \alpha \frac{\partial L}{\partial b_1}$$