We'll call:

- $a^0_{j,k}, 0 \leq j,k \leq 27$, the input activations;
- $w^1_{l,m}, 0 \leq l,m \leq 4$, the shared weights for the convolutional layer;
- b^1, the shared bias for the convolutional layer;
- $z^1_{j,k} 0 \leq j,k \leq 23$, the weighted input to neuron (j,k) (line j, column k) in the conv layer:

$$z^1_{j,k} = b^1 + \sum_{l=0}^{4} \sum_{m=0}^{4} w^1_{l,m} a^0_{j+l,k+m}$$

$a^1_{j,k}, 0 \leq j,k \leq 23$, the activation of neuron (j,k) in the convolutional layer:

$$a^1_{j,k} = \sigma\left(z^1_{j,k}\right)$$

$a^2_{j,k}, 0 \leq j,k \leq 11$, the activation of neuron (j,k) in the max-pooling layer:

$$a^2_{j,k} = max\left(a^1_{2j,2k}, a^1_{2j,2k+1}, a^1_{2j+1,2k}, a^1_{2j+1,2k+1}\right)$$

So, neuron $(\boldsymbol{j},\boldsymbol{k})$ in the convolutional layer will contribute to the computation of the max for neuron $\left(\left\lfloor \frac{j}{2} \right\rfloor, \left\lfloor \frac{k}{2} \right\rfloor\right)$.

Note: the symbol $\left\lfloor \frac{j}{2} \right\rfloor$, indicate the floor function of $\frac{j}{2}$.

- Note that the max-pooling layer doesn't have any weights, biases, or weighted inputs!
- $\boldsymbol{w^3_{l;j,k} 0 \leq j,k \leq 11, 0 \leq l \leq}$, the weight of the connection between neuron $(\boldsymbol{j},\boldsymbol{k})$ in the max-pooling layer and neuron $\boldsymbol{l}$ in the output layer;
- $\boldsymbol{b^3_l 0 \leq l \leq}$, the bias of neuron $\boldsymbol{l}$ in the output layer;
- $\boldsymbol{z^3_l 0 \leq l \leq}$, the weighted input of neuron $\boldsymbol{l}$ in the output layer:

$$z^3_l = b^3_l + \sum_{0 \leq j,k \leq 11} w^3_{l;j,k} a^2_{j,k}$$

  - $a^3_l 0 \leq l \leq$, the output activation of neuron l in the output layer:

$$a^3_l = \sigma(z^3_l)$$

Now for comparison, here are equations BP1 - BP4 for regular fully connected networks:

  - **BP1**: $\delta^L_j = \frac{\partial C}{\partial a^L_j} \sigma'\left(z^L_j\right)$

- **BP2**: $\delta_j^l = \sum_k w_{kj}^{l+1}\,\delta_k^{l+1}\sigma'\left(z_j^l\right)$
- **BP3**: $\frac{\partial C}{\partial b_j^l} = \delta_j^l$
- **BP4**: $\frac{\partial C}{\partial w_{jk}^l} = a_k^{l-1}\delta_j^l$

And their shortened derivations (only writing $\frac{\partial x}{\partial y}$ when y has an influence on x):

BP1:
$$\delta_j^L = \frac{\partial C}{\partial z_j^L} = \frac{\partial C}{\partial a_j^L}\frac{\partial a_j^L}{\partial z_j^L} = \frac{\partial C}{\partial a_j^L}\sigma'\left(z_j^L\right) \tag{1}$$

BP2:
$$\delta_j^l = \frac{\partial C}{\partial z_j^l} = \sum_k \frac{\partial C}{\partial z_k^{l+1}}\frac{\partial z_k^{l+1}}{\partial a_j^l}\frac{\partial a_j^l}{\partial z_j^l} = \sum_k \delta_k^{l+1} w_{kj}^{l+1}\sigma'\left(z_j^l\right) \tag{2}$$

BP3:
$$\frac{\partial C}{\partial b_j^l} = \frac{\partial C}{\partial z_j^l}\frac{\partial z_j^l}{\partial b_j^l} = \delta_j^l \times 1 \tag{3}$$

BP4:
$$\frac{\partial C}{\partial w_{jk}^l} = \frac{\partial C}{\partial z_j^l}\frac{\partial z_j^l}{\partial w_{jk}^l} = \delta_j^l a_k^{l-1} \tag{4}$$

Let's look at each equation in turn, with our new network architecture.

- **BP1:** The last layer following the previous network architecture, we see that the derivation of BP1 remains correct. Therefore, BP1 doesn't change.
- **BP2:** since the max-pooling layer doesn't have any weighted inputs, we'll just have to compute $\delta_{j,k}^1$.

$$\delta_{j,k}^1 = \frac{\partial C}{\partial z_{j,k}^1}$$

$$= \sum_{l=0}^{9} \frac{\partial C}{\partial z_l^3}\frac{\partial z_l^3}{\partial z_{j,k}^1}$$

$$= \sum_{l=0}^{9} \delta_l^3 \frac{\partial z_l^3}{\partial a_{j',k'}^2}\frac{\partial a_{j',k'}^2}{\partial z_{j,k}^1}$$

with $j' = \left\lfloor \frac{j}{2} \right\rfloor, k' = \left\lfloor \frac{k}{2} \right\rfloor$

$a_{j',k'}^2$ being the only activation in the max-pooling layer affected by $z_{j,k}^1$.

$$= \sum_{l=0}^{9} \delta_l^3 w_{l;j',k'}^3 \frac{\partial a_{j',k'}^2}{\partial z_{j,k}^1}$$

$$= \sum_{l=0}^{9} \delta_l^3 w_{l;j',k'}^3 \frac{\partial a_{j',k'}^2}{\partial a_{j,k}^1} \frac{\partial a_{j,k}^1}{\partial z_{j,k}^1}$$

$$= \sum_{l=0}^{9} \delta_l^3 w_{l;j',k'}^3 \frac{\partial a_{j',k'}^2}{\partial a_{j,k}^1} \sigma'\left(z_{j,k}^1\right)$$

Now since $a_{j',k'}^2 = \max\left(a_{2j',2k'}^1, a_{2j',2k'+1}^1, a_{2j'+1,2k'}^1, a_{2j'+1,2k'+1}^1\right)$ and we're talking about infinitesimal changes, we have:

$$\frac{\partial a_{j',k'}^2}{\partial a_{j,k}^1} = \begin{cases} 0, & if \;\; a_{\{j,k\}}^1 \neq \; max \; \left(a_{\{2j',2k'\}}^1, a_{\{2j',2k'+1\}}^1, a_{\{2j'+1,2k'\}}^1, a_{\{2j'+1,2k'+1\}}^1 \right) \\ 1, & if \;\; a_{\{j,k\}}^1 = \; max \; \left(a_{\{2j',2k'\}}^1, a_{\{2j',2k'+1\}}^1, a_{\{2j'+1,2k'\}}^1, a_{\{2j'+1,2k'+1\}}^1 \right) \end{cases}$$

This is because $a_{j,k}^1$ only affects $a_{j',k'}^2$ if $a_{j,k}^1$ is the maximum activation in its local pooling field. In this case, we have $a_{j',k'}^2 = a_{j,k}^1$, so $\frac{\partial a_{j',k'}^2}{\partial a_{j,k}^1}$=1.

And so to conclude the derivation of our new BP2:

$$\delta_{j,k}^1 = \begin{cases} 0, & if \;\; a_{\{j,k\}}^1 \neq \; max \; \left(a_{\{2j',2k'\}}^1, a_{\{2j',2k'+1\}}^1, a_{\{2j'+1,2k'\}}^1, a_{\{2j'+1,2k'+1\}}^1 \right) \\ \sum_{l=0}^{9} \delta_l^3 w_{l;j',k'}^3 \sigma'\left(z_{j,k}^1\right), & if \;\; a_{\{j,k\}}^1 = \; max \; \left(a_{\{2j',2k'\}}^1, a_{\{2j',2k'+1\}}^1, a_{\{2j'+1,2k'\}}^1, a_{\{2j'+1,2k'+1\}}^1 \right) \end{cases}$$

- **BP3**: we consider two cases:
  - $\frac{\partial C}{\partial b_l^3} = \delta_l^3$ as the third layer respects the previous architecture (the derivation still works);
  - $\frac{\partial C}{\partial b^1}$. This one is different, since the bias b^1 is shared for all neurons in the convolutional layer.

We have:

$$\frac{\partial C}{\partial b^1} = \sum_{0 \leq j,k \leq 23} \frac{\partial C}{\partial z_{j,k}^1} \frac{\partial z_{j,k}^1}{\partial b^1}$$

$$= \sum_{0 \le j,k \le 23} \delta^1_{j,k} \frac{\partial z^1_{j,k}}{\partial b^1}$$

$$= \sum_{0 \le j,k \le 23} \delta^1_{j,k} \qquad \text{as } \; z^1_{j,k} = b^1 + \sum_{l=0}^{4} \sum_{m=0}^{4} w^1_{l,m} a^0_{j+l,k+m}$$

- **BP4**:
  - $\frac{\partial C}{\partial w^3_{l;j,k}} = a^2_{j,k} \delta^3_l$ since, again, the derivation still works for the third layer;
  - $\frac{\partial C}{\partial w^1_{l,m}}, 0 \le l, m \le 4$. These 25 weights are shared, and each of them is used in the computation of the weighted input of each neuron in the convolutional layer:

$$\frac{\partial C}{\partial w^1_{l,m}} = \sum_{0 \le j,k \le 23} \frac{\partial C}{\partial z^1_{j,k}} \frac{\partial z^1_{j,k}}{\partial w^1_{l,m}}$$

$$= \sum_{0 \le j,k \le 23} \delta^1_{j,k} \frac{\partial z^1_{j,k}}{\partial w^1_{l,m}}$$

$$= \sum_{0 \le j,k \le 23} \delta^1_{j,k} a^0_{j+l,k+m} \qquad \text{as } z^1_{j,k} = b^1 + \sum_{l=0}^{4} \sum_{m=0}^{4} w^1_{l,m} a^0_{j+l,k+m}$$