Assignment - 3

Data mining and machine learning

(7) (ii) $\frac{\partial E_d}{\partial \omega_{11}^{(2)}}$

The error function E_d can be given as:

$$E_d = \frac{1}{2}(y - oy_1)^2$$

Using the chain rule

$$\frac{\partial E_d}{\partial \omega_{11}^{(2)}} = \frac{\partial E_d}{\partial oy_1} \cdot \frac{\partial oy_1}{\partial net y_1} \cdot \frac{\partial net y_1}{\partial \omega_{11}^{(2)}}$$

where:

$$\frac{\partial E_d}{\partial oy_1} = -(y - oy_1)$$

$$\frac{\partial oy_1}{\partial net y_1} = oy_1(1 - oy_1)$$ = derivative of the sigmoid

$$\frac{\partial net y_1}{\partial \omega_{11}^{(2)}} = oz_1$$

$\therefore$ The full chain rule formula is:

$$\frac{\partial E_d}{\partial \omega_{11}^{(2)}} = -(y - oy_1) \cdot oy_1(1 - oy_1) \cdot oz_1$$

7 (iii) Computing δ_{Y_1} as:

$$\delta_{Y_1} = (y - O_{Y_1}) \cdot O_{Y_1}(1 - O_{Y_1})$$

where:

y is the target output (0 in this case)

O_{Y_1} is the output (≈ 0.513122)

Substituting the values:

$$\delta_{Y_1} = (0 - 0.513122) \cdot 0.513122 \cdot (1 - 0.513122)$$

Calculating δ_{Y_1}

$$\delta_{Y_1} = -0.513122 \times 0.513122 \times 0.486878$$

$$\approx -0.1294$$

7 (iv) $\omega_{11}^{(2)}$

The weight update rule can be given as:

$$\omega_{ij}^{(new)} = \omega_{ij}^{(old)} + \eta \cdot \delta_j \cdot O_i$$

where:

η is the learning rate (0.2)

δ_j is the error term for the neuron (in this case δ_{Y_1})

O_i is the output from the previous layer (i.e O_{Z_1}

Using

$\omega_{11}^{(2)} = 0.1$

$\delta y_1 \approx -0.1294$

$O_{z1} \approx 0.524979$

Substituting into the update formula

$\omega_{11}^{(2,new)} = 0.1 + 0.2 \cdot (-0.1294) \cdot (0.524979)$

$\omega_{11}^{(2,new)} = 0.1 - 0.2 \cdot 0.0679 \approx 0.1 - 0.01358$

≈ 0.08642

7 (v) $\omega_{21}^{(1)}$

$\delta z1 = \delta y_1 \cdot \omega_{11}^{(2)} \cdot O_{z1}(1 - O_{z1})$

Using $\omega_{11}^{(2)} = 0.1$

$\delta y_1 \approx -0.1294$

$O_{z1} \approx 0.5249$

$\delta z1 = -0.1294 \cdot 0.1 \cdot 0.524979\,(1 - 0.524979)$

$\delta z1 = -0.1294 \cdot 0.1 \cdot 0.524979 \cdot 0.475021 \approx -0.0031$

Updating $\omega_{21}^{(1)}$

$\omega_{21}^{(1,new)} = \omega_{21}^{(1,old)} + \eta \cdot \delta z1 \cdot x_2$

Assuming $\omega_{21}^{(1)} = 0.1$ and $x_2 = 0$

$$\omega_{21}^{(1,new)} = 0.1 + 0.2 \cdot (-0.0031) \cdot 0 = 0.1$$

Therefore

$\delta y_1 \approx -0.1294$

Updated $\omega_{11}^{(2)} \approx 0.08642$

Updated $\omega_{21}^{(1)} = 0.1$

8 (i) Given image

2	4	5	9	7
6	3	1	4	6
4	2	7	7	2
6	5	9	9	8
8	4	3	6	5

Given filter

2	3	1
1	0	4
4	8	4

Taking the red region from top left corner

$$\begin{matrix} 2 & 4 & 5 \\ 6 & 3 & 1 \\ 4 & 2 & 7 \end{matrix}$$

So calculating each element of the filter with corresponding element of image region

$(2 \cdot 2) + (3 \cdot 4) + (1 \cdot 5) + (1 \cdot 6) + (0 \cdot 3) + (4 \cdot 1) + (4 \cdot 4)$

$+ (8 \cdot 2) + (4 \cdot 7)$

$= 4 + 12 + 5 + 6 + 0 + 4 + 16 + 16 + 28$

$= 91$

Applying the Relu activation function as

output $= \max(0, 91) = 91$

Since the output is positive Relu does not alter it.

(8) (ii) Given stride $= 2$, no padding

The size of the feature map can be calculated as output size $= \left\lfloor \frac{(w - F + 2P)}{S} \right\rfloor + 1$

where W = width of the input image
F = width of the filter
P = padding
S = stride

$W = 5$
$F = 3$
$P = 0$
$S = 2$

$$\text{Output width} = \left\lfloor \frac{(5-3+2\cdot 0)}{2} \right\rfloor + 1 = \left\lfloor \frac{2}{2} \right\rfloor + 1$$

$$= 1 + 1 = 2$$

Since the image is also 5x5 the height will be the same

Output height = 2

∴ The size of the feature map is 2X2

(8) (iii) Total number of parameters when there are 10 feature maps can be given as:

Total parameters = $(F \cdot F \cdot \text{input channels} + 1) \cdot$ no. of feature maps

Assuming the input image has 1 channel i.e grayscale :

filter size $F = 3$

no. of feature maps $= 10$

Parameters per feature map $= (3 \cdot 3 \cdot 1 + 1) = 9 + 1 = 10$

so for 10 feature maps the total parameters can be given as $10 \cdot 10 = 100$

$\therefore$ therefore output of feature map node $= 91$

Size of feature map $= 2 \times 2$

Total no. of parameters for 10 feature maps $= 100$

⑨ (i) Example of 1×1 convolution filter

$$\text{filter} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix}$$

where $\omega_1, \omega_2, \omega_3, \omega_4$ are the weights for each channel

So filter = $\begin{bmatrix} 0.5 \\ 1.0 \\ -0.5 \\ 2.0 \end{bmatrix}$

ⓐ (ii) Assuming the multi channel data at the upper left corner as:

channel 1	channel 2	channel 3	channel 4
1	2	3	4
5	6	7	8
9	10	11	12

Given values of the upper top left corner are

2 1 5 4

the values at the upper left corner are:

channel 1 : 1

channel 2 : 2

channel 3 : 3

channel 4 : 4

Using the 1x1 filter

$$\text{Output} = (w_1 \cdot 1) + (w_2 \cdot 2) + (w_3 \cdot 3) + (w_4 \cdot 4)$$

$$= (0.5 \cdot 2) + (1.0 \cdot 1) + (-1.5 \cdot 5) + (2.0 \cdot 4)$$

$$= (1) + (1) + (-7.5) + (8)$$

$$= 2 - 7.5 + 8$$

$$= 10 - 7.5$$

$$= 2.5$$