Artificial Neural Networks
Final Exam Review
Summer 2022/2023

Q1 T/F

Q2 Choose the best answer

Q3: Game:
Let us say that every team has 12 players. You are given the following input data to the network:

1) The skill level of every player on each team. 0: ("is bad player") to 5: ("Good player"). How can you represent the skill of each player and how many neurons is needed?

Represent each player's skill level by a single input (neuron).
Divide the skill level by 5 so that the input values range from 0 to 1.
Total input neurons required: 24

2) The # of matches that every team has played during the last three weeks.
There are never more than 8 matches in that period of time.
Represent each team's value by 1 neuron.
Divide the value by 8 so that the input is scaled from 0 to 1.
Total input neurons required: 2

3) The statistics of former matches between the same two teams within the past 5 years (e.g., Team A won 20% of the matches, Team B 55%, and 25% of the matches were tied).

Represent the proportion of wins by Team A with one neuron and the one for Team B by another one. You could also use a third neuron for the proportion of ties, but this redundant information may not help with the learning.
Total input neurons required: 2

4) The continent that each team comes from (North America, South America, Europe, Africa, Asia, or Australia).

These values represent categories.
So use 2 neuron to representation the categories 1-6 for each team.

5) Where the match takes place (Team A's stadium, Team B's stadium, or neutral place).
You could use just one neuron, using a scale like 0 = Team A's stadium, 0.5 = neutral, 1 = Team B' stadium.
Let's say you are using 1 neuron

6) The phase of the football season (early season vs. late season).
Since this is only a binary variable, one neuron (value 0 or 1) is sufficient.

7) How many hidden-layer neurons is sufficient for this network?
Total number of input neurons is 32.
The best number of hidden-layer neurons is difficult to estimate; let's say 10.
You could use 2 output neurons, one for each team's number of goals scored. You would have to estimate the maximum number of goals ever to be scored, and then scale the desired output values so that output ε represents 0 goals, and output (1-ε) represents that maximum number of goals.

8) How many weights is sufficient for this network?
As usual, you should use a completely connected network, which means that it has a total of 32 x 10 + 10 x 2 = 340 weights.

9) How many training samples is sufficient for this network?
Ideally you should collect at least 340 *10 = 3400 training samples. You could maybe find them in online databases of soccer matches during the past years.

10) Which NN Algorithm is sufficient for this game problem?

The backpropagation learning is good. After the network error is below a certain threshold, test the network's performance using another, possibly smaller set of exemplars, called the test set.

Q4: Application Design: create dataset

rule 1: if hair is blond and eyes is blue and height 5.8 then Class A
rule 2: if hair is blond and nose is short and height is 5.8 then Class A
rule 3: if hair is red and eyes is blue and weight 90 then Class B
rule 4: if hair is red and eyes is green and weight 80 then Class B
rule 5: if hair is Black and eyes is blue and height 5.5 then Class C
rule 6: if hair is Brown and eyes is green and height 5.4 then Class C

Input attributes						Output		
hair	eyes	height	nose	weight		Class A	Class B	Class C
blond	Blue	5.8	?	?		1	0	0
blond	?	5.8	short	?		1	0	0
red	blue	?	?	90		0	1	0
Red	green	?	?	80		0	1	0
black	blue	5.5	?	?		0	0	1
brown	green	5.4	?	?		0	0	1

Q5: Suppose you have the following dataset

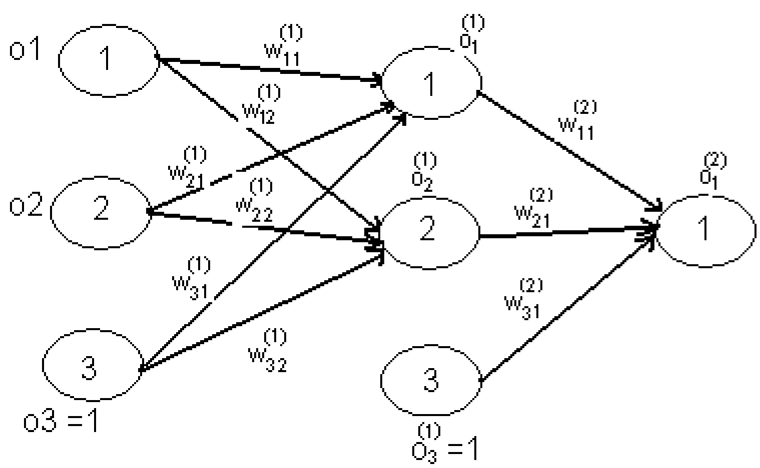
x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	y
1	1	1	1	1	1	1	1	1	0	0	1
1	1	1	0	0	1	1	1	1	0	0	1
0	0	1	1	1	1	1	1	1	1	1	0
0	0	1	1	0	1	1	1	1	1	1	0

Calculate the correlation confident of each attribute with the output y

x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	y
1	1	1	1	1	1	1	1	1	-1	-1	1
1	1	1	-1	-1	1	1	1	1	-1	-1	1
-1	-1	1	1	1	1	1	1	1	1	1	-1
-1	-1	1	1	-1	1	1	1	1	1	1	-1
+1	**1**	**0**	**-1/2**	**0**	**0**	**0**	**0**	**0**	**-1**	**-1**	

$X1 = ¼(1*1+1*1+(-1)*(-1)+(-1)*(-1))= +1$

Q6: Backpropagation Network



The initial weights and threshold levels are set randomly as follows:

0.5	0.9	0.4	1.0	–1.2	1.1	0.8	–0.1	0.3.
$W_{11}^{(1)}$	$W_{12}^{(1)}$	$W_{21}^{(1)}$	$W_{22}^{(1)}$	$W_{11}^{(2)}$	$W_{21}^{(2)}$	$W_{31}^{(1)}$	$W_{32}^{(1)}$	$W_{31}^{(2)}$

You can consider a training set where inputs $o_1 = 1$ and $o_2 = 1$ and desired target output is 0.
Learning Rate = 0.1

a) Calculate the output at the hidden layer $o_1^{(1)}$ and $o_2^{(1)}$
b) Calculate the output of the network $o_1^{(2)}$
c) Calculate the error E
d) Calculate $\delta_1^{(2)}$
e) Calculate $\delta_1^{(1)}$ and $\delta_2^{(1)}$
f) Determine the weight corrections

$o_1^{(1)} = \text{sigmoid}\,(O_1W_{11}^{(1)} + O_2W_{21}^{(1)} - W_{31}^{(1)})$								0.3100
$o_2^{(1)} = \text{sigmoid}\,(O_1W_{12}^{(1)} + O_2W_{22}^{(1)} - W_{32}^{(1)})$								0.5250
$o_1^{(2)} = \text{sigmoid}\,(O_1^{(1)} W_{11}^{(2)} + O_2^{(1)} W_{21}^{(2)} - W_{31}^{(2)})$								0.4764
				$e = o_1^{(2)} - t1 =$				0.4764
			$\delta_1^{(2)} = o_1^{(2)}(1 - o_1^{(2)})\,\varepsilon$					0.1188
$\Delta wij^{(2)} = y o i^{(1)}\,\delta^{(2)}$, for i = 1, ... , 3 and j = 1 // only one output								
						$\Delta w_{11}^{(2)} =$		-0.0037
						$\Delta w_{21}^{(2)} =$		-0.0062
						$\Delta w_{31}^{(2)} =$		0.0119
Next, we calculate the backpropagated error at the hidden layer								
$\delta\varphi^{(1)} = o\varphi^{(1)}(1 - o\varphi^{(1)}) \sum^{\mu}_{\theta=1} w\varphi\theta^{(2)}\ \delta\theta^{(2)}$								
			$\delta_1^{(1)} = o_1^{(1)}(1 - o_1^{(1)})\, w_{11}^{(2)}$					-0.0305
			$\delta_2^{(1)} = o_2^{(1)}(1 - o_2^{(1)})\, w_{21}^{(2)}$					0.0326
We then determine the weight corrections:								
					$\Delta w_{11}^{(1)} = -y\, o_1\, \delta_1^{(1)} =$			0.0000

$\Delta w_{12}^{(1)} = -y\ o_1\ \delta_2^{(1)}$ =	0.0000
$\Delta w_{21}^{(1)} = -y\ o_2\ \delta_1^{(1)}$ =	0.0000
$\Delta w_{22}^{(1)} = -\ y\ o_2\ \delta_2^{(1)}$ =	0.0000
(θ3) $\Delta w_{31} = -y\ (-1)\ \delta_1^{(1)}$ =	-0.0031
(θ4) $\Delta w_{32}^{(1)} = -\ y\ (-1)\ \delta_2^{(1)}$ =	0.0033
At last, we update all weights and threshold:	
$w_{11}^{(1)} = w_{11}^{(1)} + \Delta w_{11}^{(1)}$ =	0.5000
$w_{12}^{(1)} = w_{12}^{(1)} + \Delta w_{12}^{(1)}$ =	0.9000
$w_{21}^{(1)} = w_{21}^{(1)} + \Delta w_{21}^{(1)}$ =	0.4000
$w_{22}^{(1)} = w_{22}^{(1)} + \Delta w_{22}^{(1)}$ =	1.0000
$W_{11}^{(2)} = W_{11}^{(2)} + \Delta\ W_{11}^{(2)}$ =	-1.2037
$W_{21}^{(2)} = W_{21}^{(2)} + \Delta\ W_{21}^{(2)}$ =	1.0938
(θ3) $w_{31}^{(1)} = W_{31}^{(1)} + \Delta\ W_{31}^{(1)}$ =	0.8119
(θ4) $w_{32}^{(1)} = W_{32}^{(1)} + \Delta\ W_{32}^{(1)}$ =	-0.0967
(θ5) $W_{31}^{(2)} = W_{31}^{(2)} + \Delta\ W_{31}^{(2)}$ =	0.3119

End of Exam