

Assegnamento 3 : Supervises Learning Neural Networks

January 24, 2022

Nicolò Toscani

1 Esercizio 1

Il training set xor-bin.arff contiene 4 possibili input per la funzione *XOR* per questo problema di classificazione.

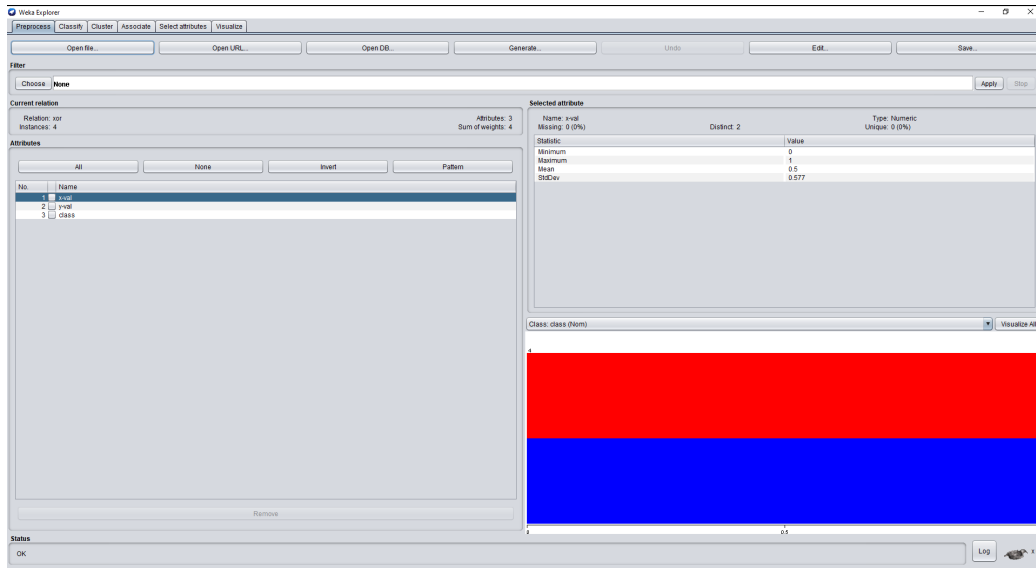


Figure 1: Training set xor-bin.arff

No.	1: x-val	2: y-val	3: class
	Numeric	Numeric	Nominal
1	0.0	0.0	0
2	1.0	0.0	1
3	0.0	1.0	1
4	1.0	1.0	0

Figure 2: Training set xor-bin.arff

Eseguendo su 3000 epoche senza layer nascosti con un valore *random* di seed, *momentum* di 0.2 (modifica dei pesi in base ai valori precedenti) e *learning rate* di 0.3 si ottiene la seguente matrice di confusione:

a	b	veri	falsi
0	2	0	2
0	2	2	0

Table 1: Matrice di confusione senza layer nascosti

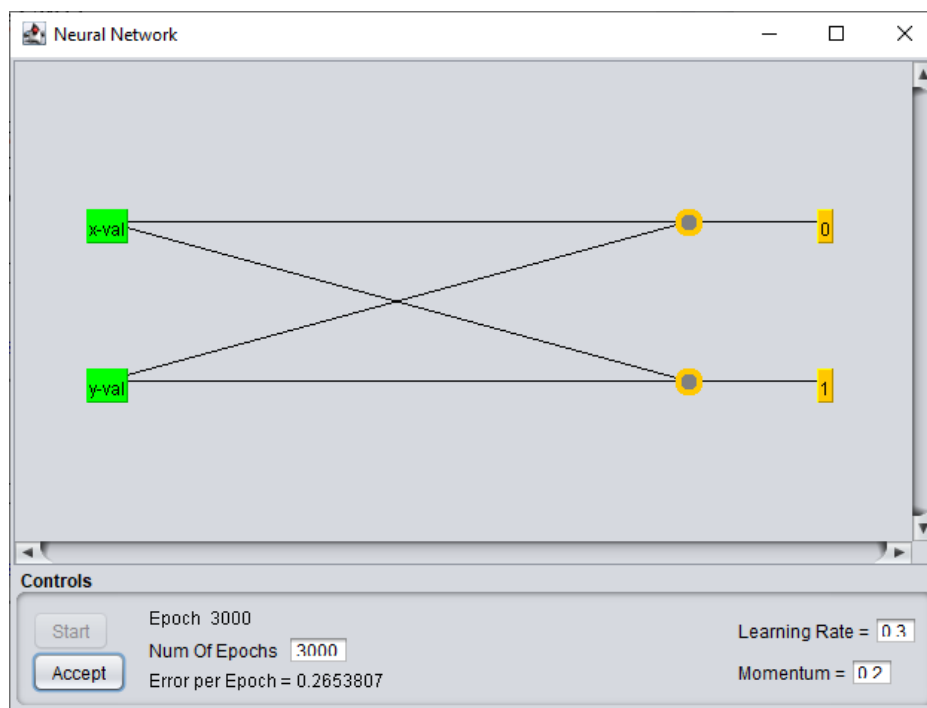


Figure 3: Rete senza layer nascosti

Con questa configurazione vengono classificati correttamente solo 2 patterns.

Aggiungendo alla rete un layer nascosto formato da 2 neuroni riusciamo a classificare correttamente 3 istanze (rispetto alle 2 della configurazione precedente.)

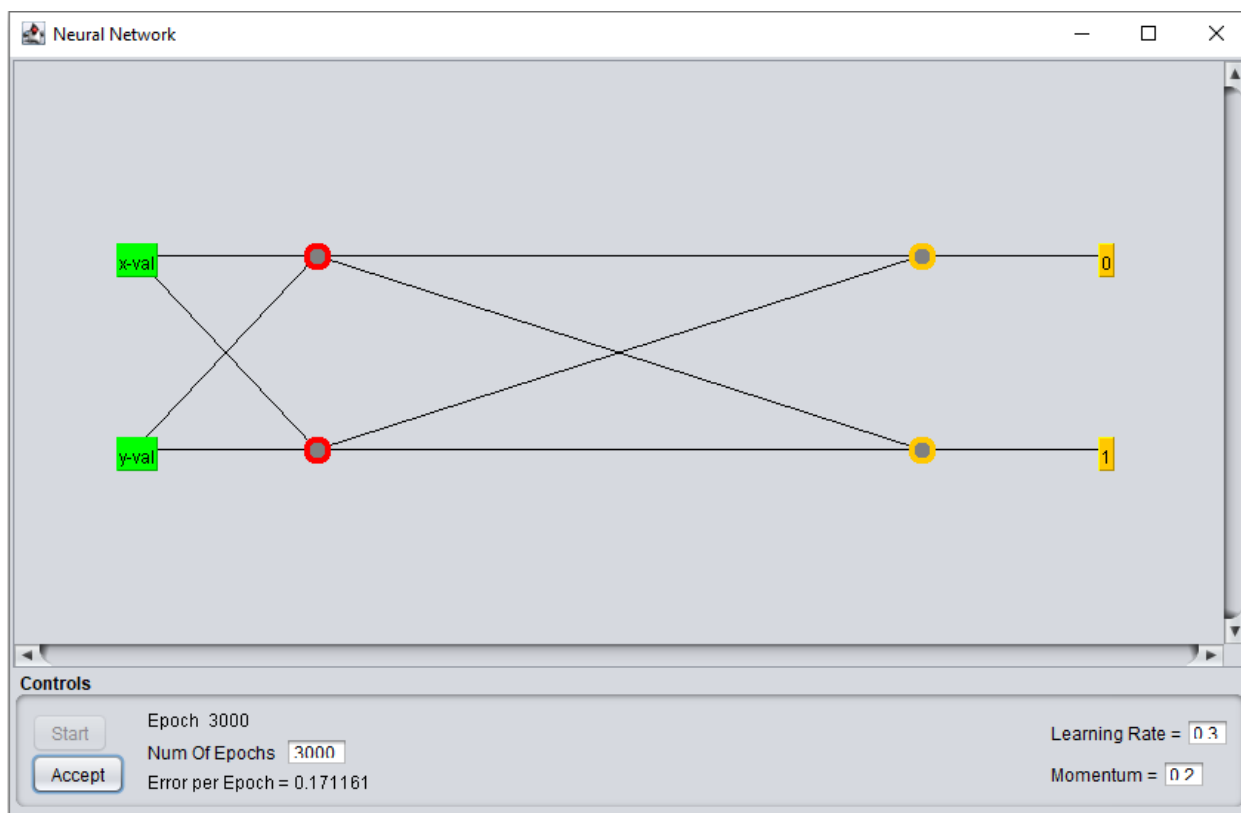


Figure 4: Rete con un layer nascosto

a	b	veri	falsi
1	1	1	1
0	2	2	0

Table 2: Matrice di confusione singolo layer nascosto

In questa configurazione otteniamo la seguente matrice di confusione.

Eseguendo la stessa configurazione con 50000 epoche arrivo a convergenza in quanto la rete riesce a riconoscere correttamente 4 patterns.

Riusciamo a dimostrare che una rete che non ha strati nascosti non riesce a valutare la funzione XOR ma risulta possibile solamente aggiungendo un layer con 2 neuroni.

Utilizzando come test set il file *xorall.arff* possiamo osservare come lo spazio viene partizionato:

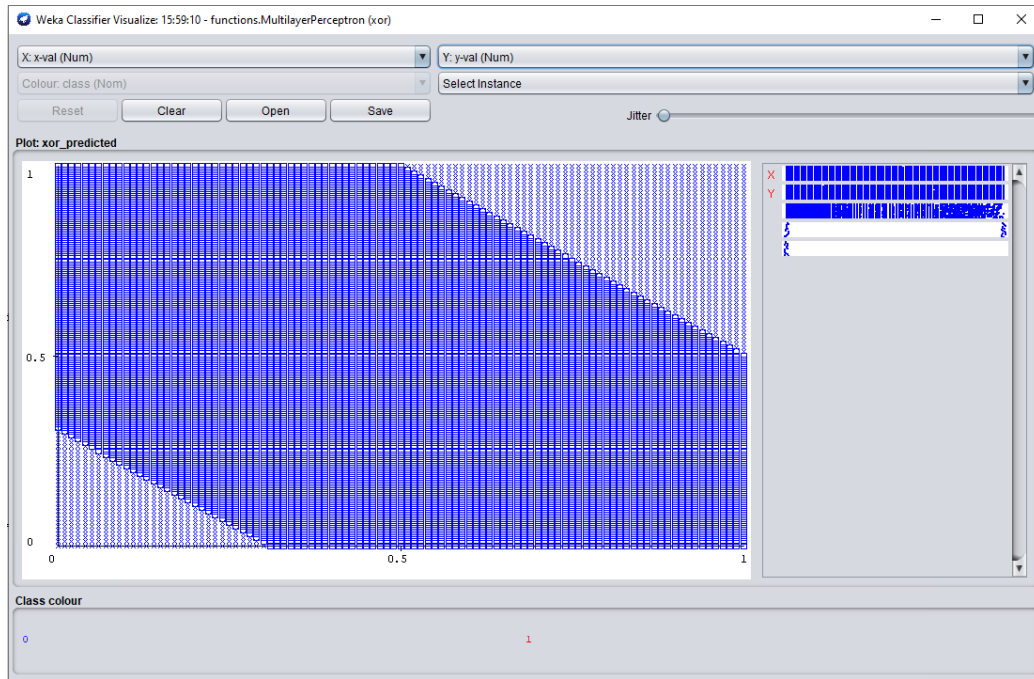


Figure 5: Classifier errors

Utilizzando ora come training set il file *xor-real.arff* trasformiamo il prob-

lema in un compito di regressione lineare.

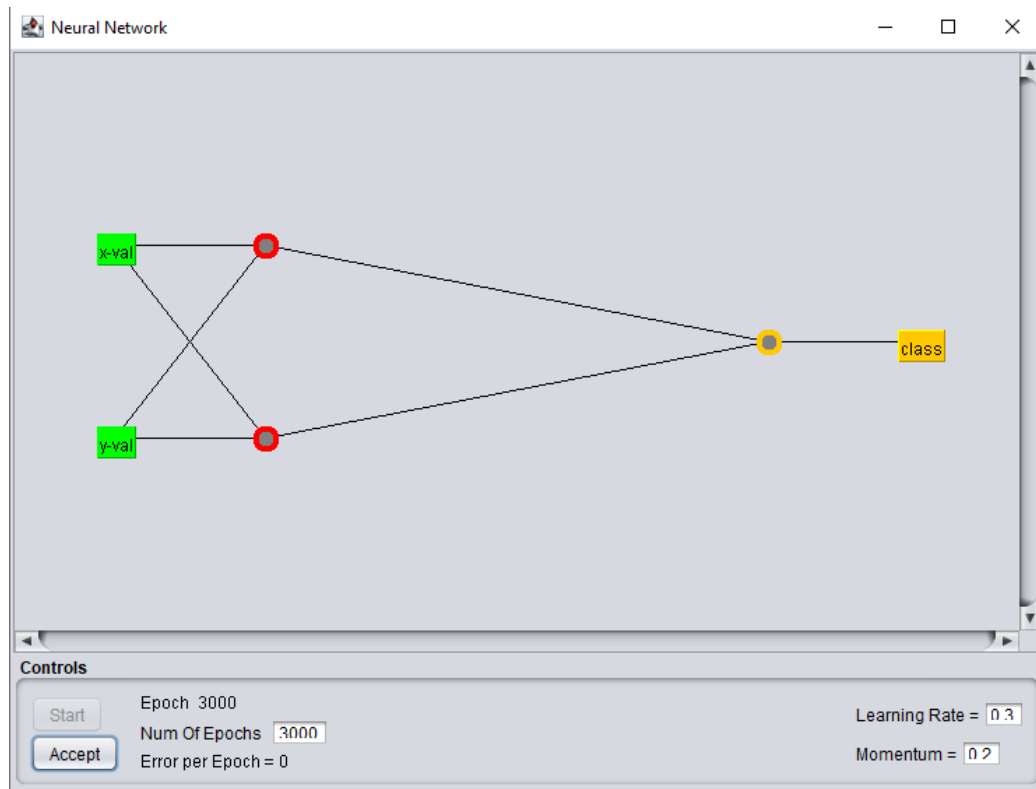


Figure 6: Layer nascosto

=== Summary ===

Correlation coefficient	1	
Mean absolute error	0	
Root mean squared error	0	
Relative absolute error	0	%
Root relative squared error	0	%
Total Number of Instances	4	

Figure 7: Layer nascosto - risultati

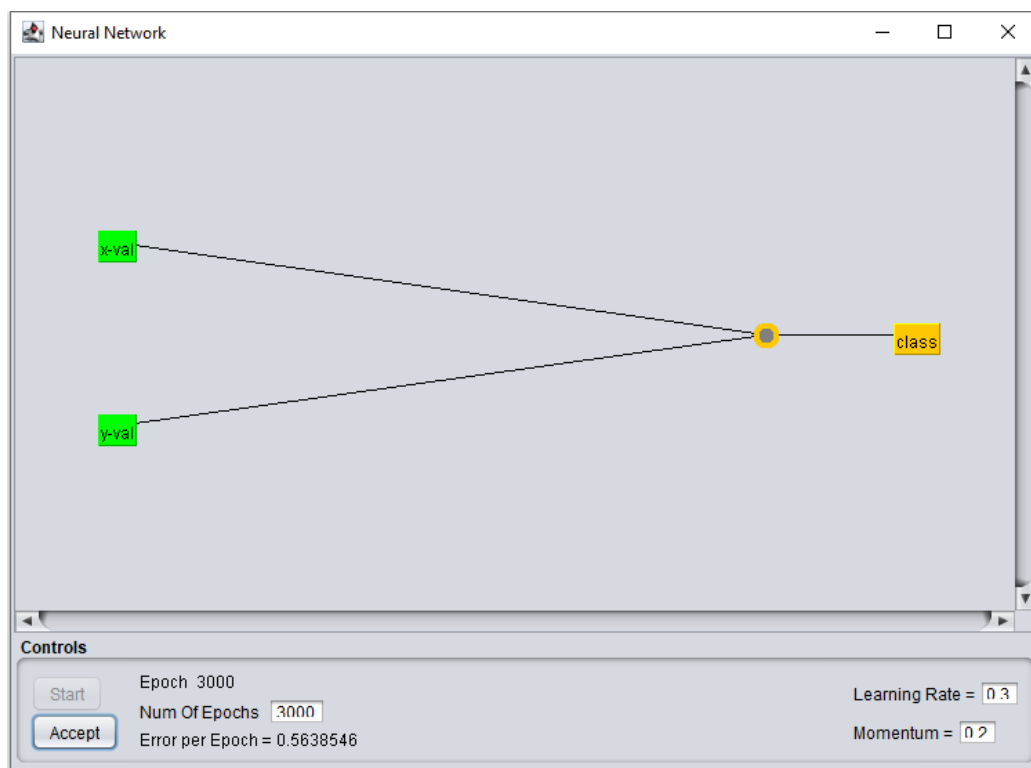


Figure 8: Nessun layer nascosto

```

=== Summary ===

Correlation coefficient          0
Mean absolute error             0.5
Root mean squared error        0.6109
Relative absolute error        100      %
Root relative squared error    122.1754 %
Total Number of Instances      4

```

Figure 9: Nessun layer nascosto - risultati

1.1 Esercizio 1b

Generiamo tre differenti file per l'addestramento della rete basati sulla funzione:

$$f(x, y) = \sin(\pi(x^2 + y^2))$$

dove

$$(x, y) \in [-1, 1]$$

Il primo test viene effettuato sul set di campionamento a 0.2.

Test 1

- Network size: n.1 hidden layer
- Neurons: n.2 neuroni per layer.
- Epochs: 3000
- Momentum: 0.2
- Learning rate: 0.2
- Seed: 0

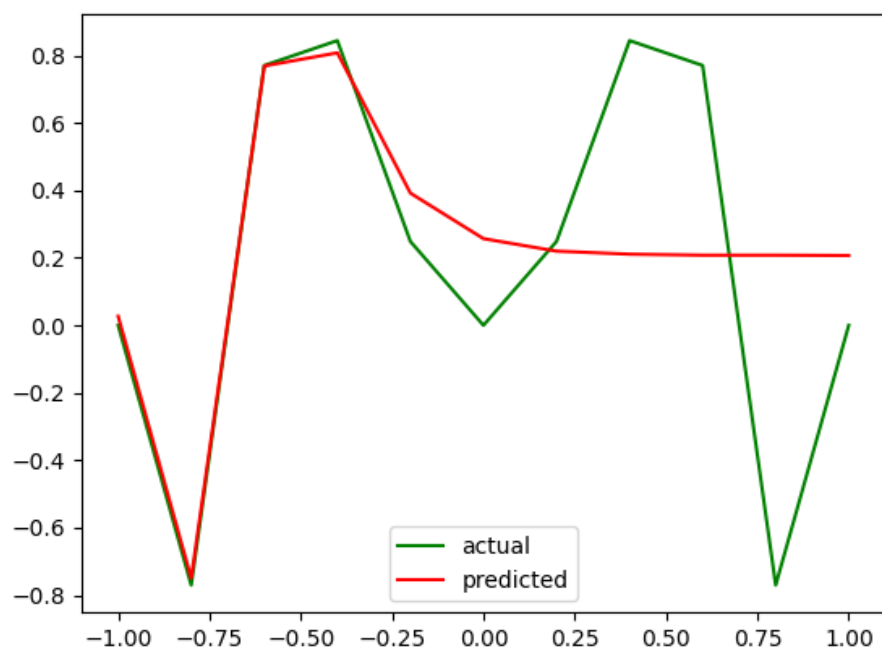


Figure 10: Test 1

Test 2

- Network size: n.1 hidden layer
- Neurons: n.30 neuroni per layer.
- Epochs: 3000
- Momentum: 0.2
- Learning rate: 0.2
- Seed: 5

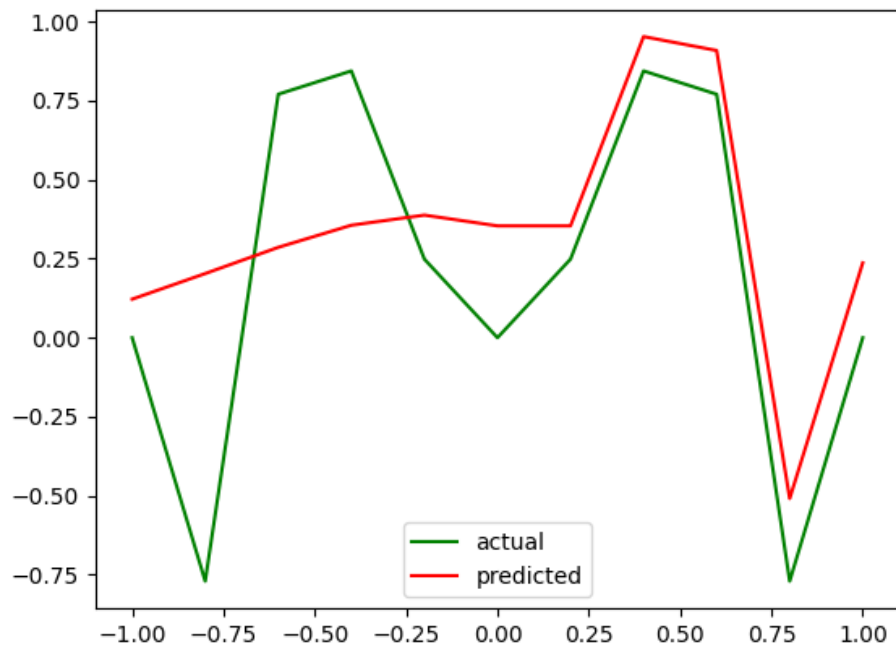


Figure 11: Test 2

Test 3

- Network size: n.1 hidden layer
- Neurons: n.12 neuroni per layer.
- Epochs: 5000
- Momentum: 0.2
- Learning rate: 0.3
- Seed: 0

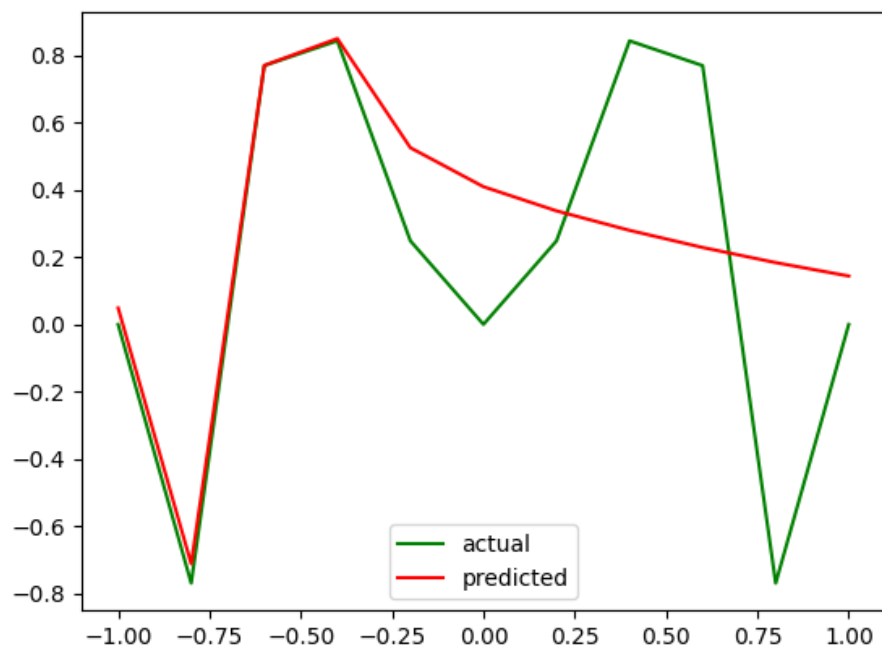


Figure 12: Test 3

Test 4

- Network size: n.1 hidden layer
- Neurons: n.11 neuroni per layer.
- Epochs: 3000
- Momentum: 0.15
- Learning rate: 0.7
- Seed: 2

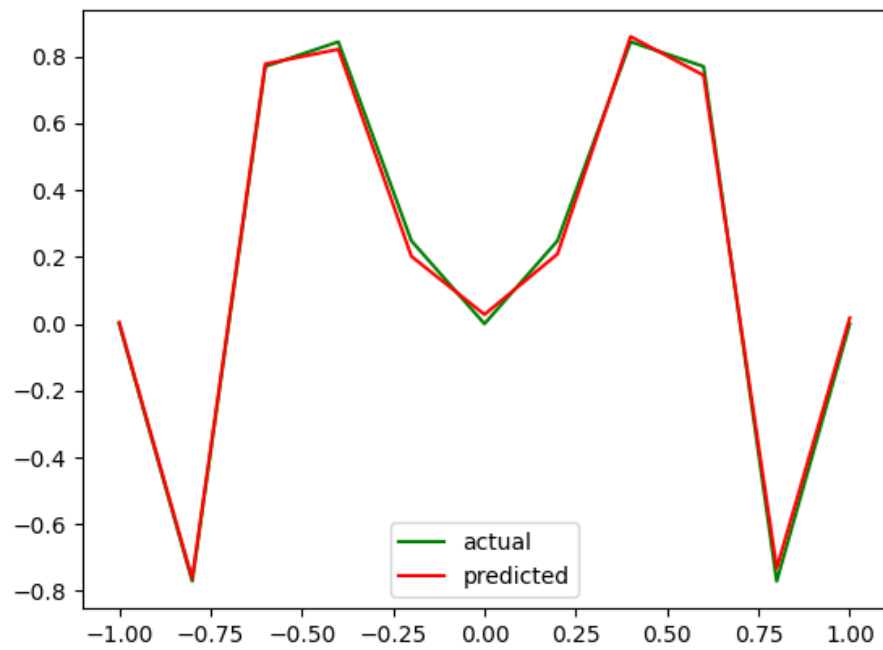


Figure 13: Test 4

Il secondo test viene effettuato sul set di campionamento a 0.1.

Test 1

- Network size: n.1 hidden layer
- Neurons: n.15 neuroni per layer.
- Epochs: 3000
- Momentum: 0.2
- Learning rate: 0.2
- Seed: 0

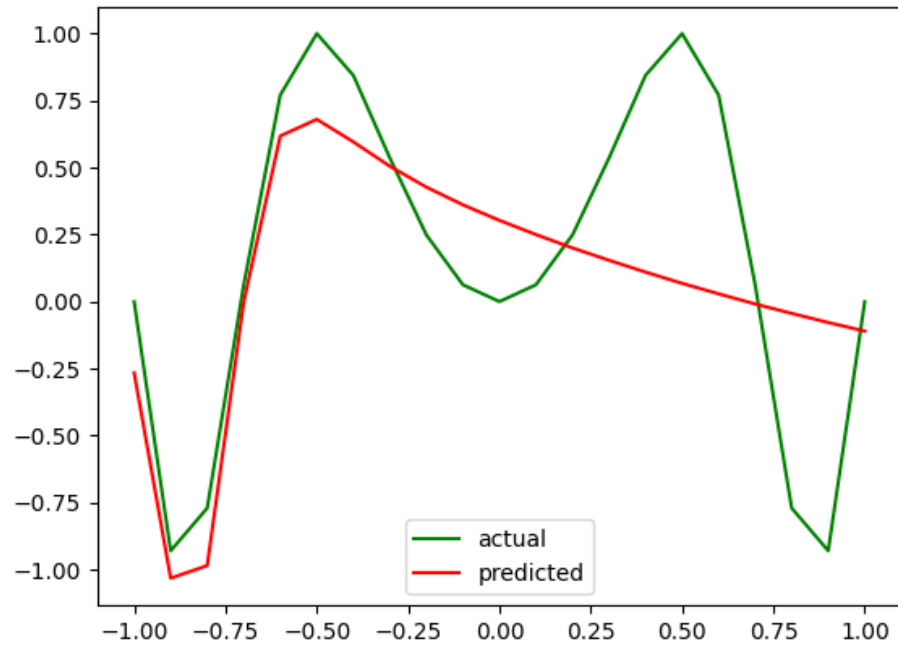


Figure 14: Test 1

Test 2

- Network size: n.2 hidden layer
- Neurons: n.15 neuroni layer 1, 7 layer 2
- Epochs: 3000
- Momentum: 0.2
- Learning rate: 0.2
- Seed: 0

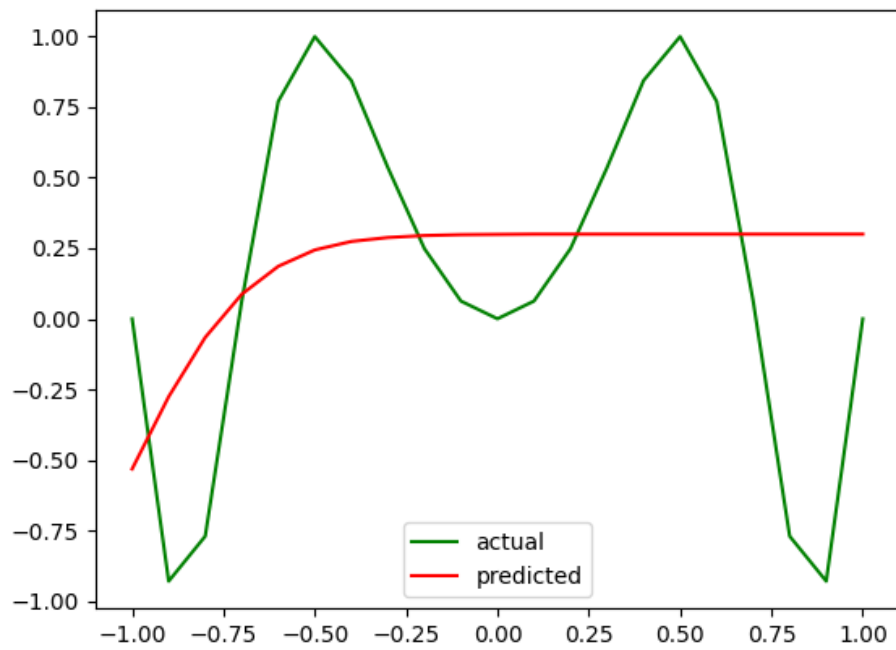


Figure 15: Test 2

Test 3

- Network size: n.1 hidden layer
- Neurons: n.100 neuroni
- Epochs: 5000
- Momentum: 0.1
- Learning rate: 0.2
- Seed: 2

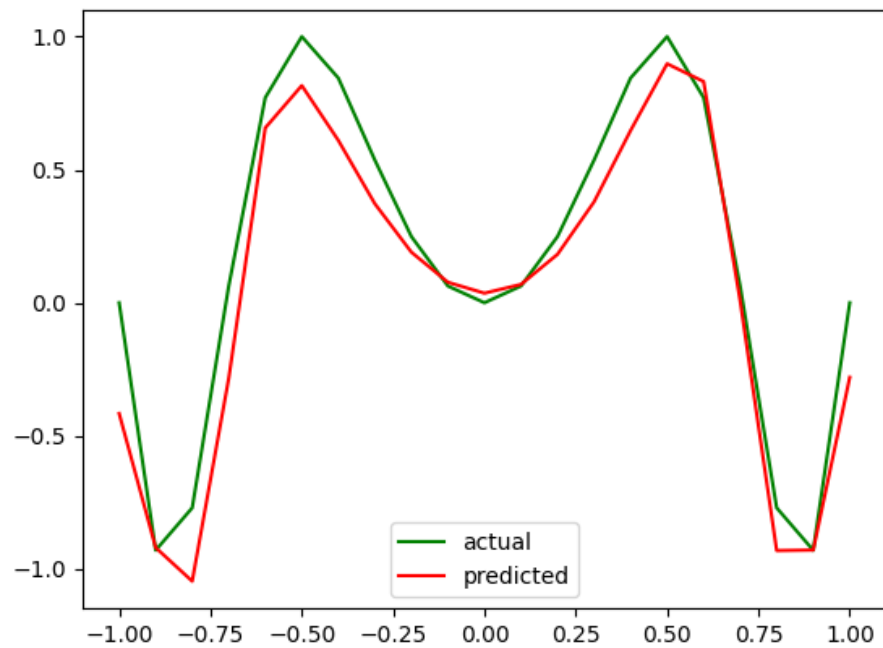


Figure 16: Test 3

Il terzo test viene effettuato sul set di campionamento a 0.05.

Test 1

- Network size: n.1 hidden layer
- Neurons: n.100 neuroni per layer.
- Epochs: 3000
- Momentum: 0.2
- Learning rate: 0.1
- Seed: 0

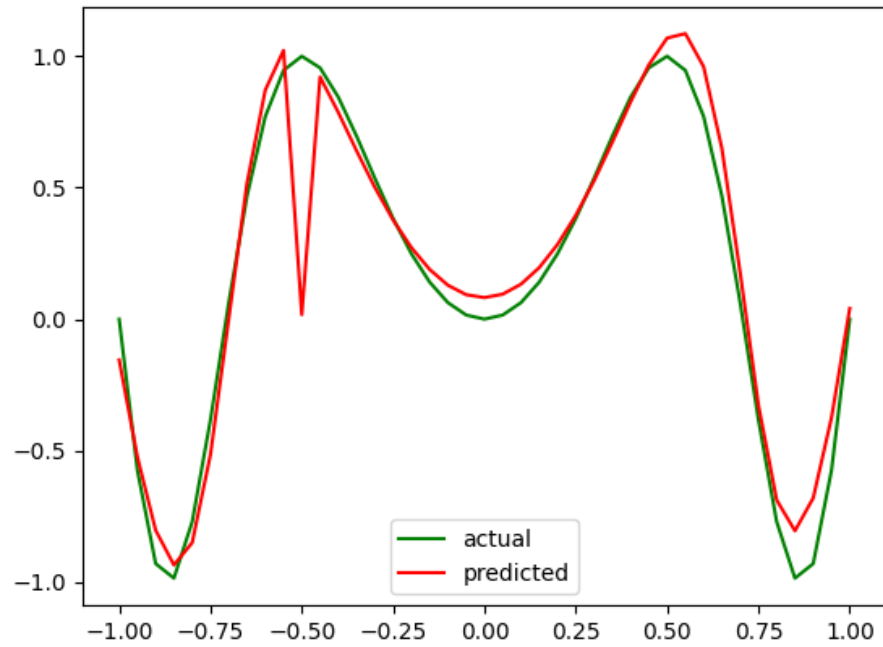


Figure 17: Test 1

Test 2

- Network size: n.1 hidden layer
- Neurons: n.41 neuroni per layer.
- Epochs: 5000
- Momentum: 0.2
- Learning rate: 0.2
- Seed: 0

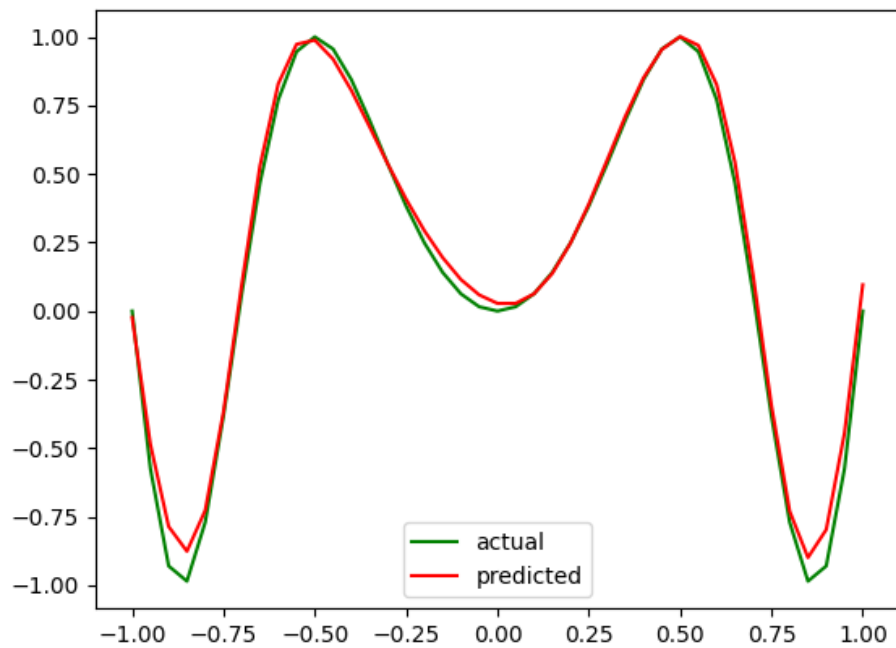


Figure 18: Test 2

1.2 Esercizio 2

Utilizzando il file *Bigtest1_104.arff* come training set e *Bigtest2_104.arff* come test set, si cerca di addestrare la rete fino a che l'errore non diventa inferiore a 0.0005.

Test 1

- momentum 0.2
- learning rate 0.2
- hidden layer: n.1 (a)

Test 2

- momentum 0.2
- learning rate 0.2
- hidden layer: n.2 (a,a)

Test 3

- momentum 0.2
- learning rate 0.2
- hidden layer: n.3 (a,a,a)

Test 4

- momentum 0.2
- learning rate 0.35
- hidden layer: n.2 (a,a)

Test 5

- momentum 0.7
- learning rate 0.2
- hidden layer: n.2 (a,a)

Test 6

- momentum 0.1
- learning rate 0.1
- hidden layer: n.1 (a)

Test 6

- momentum 0.1
- learning rate 0.1
- hidden layer: n.2 (a,a)

accuracy	correct	incorrect	test time
97.3453 %	4877	133	3.16 s
97.3253 %	4876	134	2.61 s
96.7066 %	4845	165	2.77 s
97.2056 %	4870	140	2.62 s
96.9062 %	4855	155	2.88 s
97.6447 %	4892	118	1.91 s

Table 3: Esecuzioni