



Perceptron Simple y Multicapa

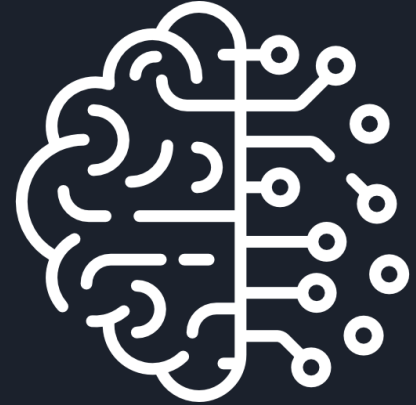
Sistemas de Inteligencia Artificial

Ariadna Fernandez Truglia
Faustino Maggioni
Florencia Chao



El perceptron simple

El perceptrón es una abstracción de la neuronas humanas, recibe entradas “estímulo” y produce una salida en bases a esos valores. Estos estímulos pueden tener un peso asociado, el cual le agrega o resta importancia a dicha entrada.



El perceptron multicapa

El perceptron multicapa es una red de perceptrones, organizada en N capas, con una cantidad de perceptrones cada una, donde las entradas que reciben los perceptrones de una capa son las salidas de las unidades de la capa anterior.





Perceptron Simple con función de activación escalón





① Función lógica “Y”

Entrada: $\{-1,1\},\{1,-1\},\{-1,-1\},\{1,1\}$

Salida esperada: $\{-1,-1,1,1\}$

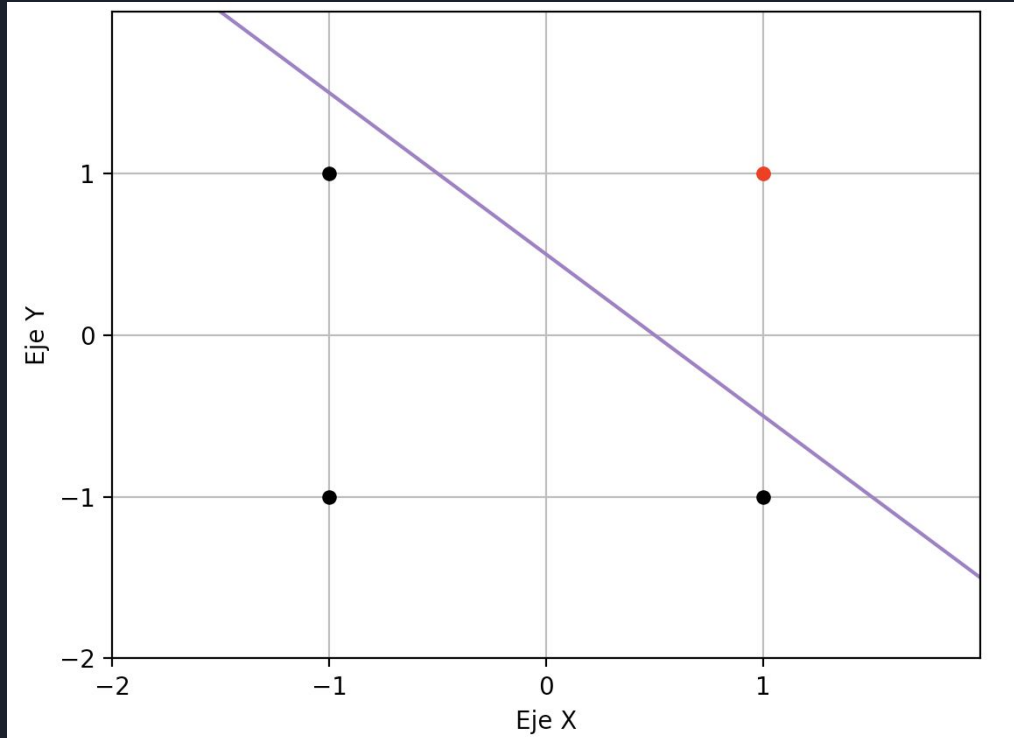
② Función lógica “O exclusivo”

Entrada: $\{-1,1\},\{1,-1\},\{-1,-1\},\{1,1\}$

Salida esperada: $\{1,1,-1,-1\}$

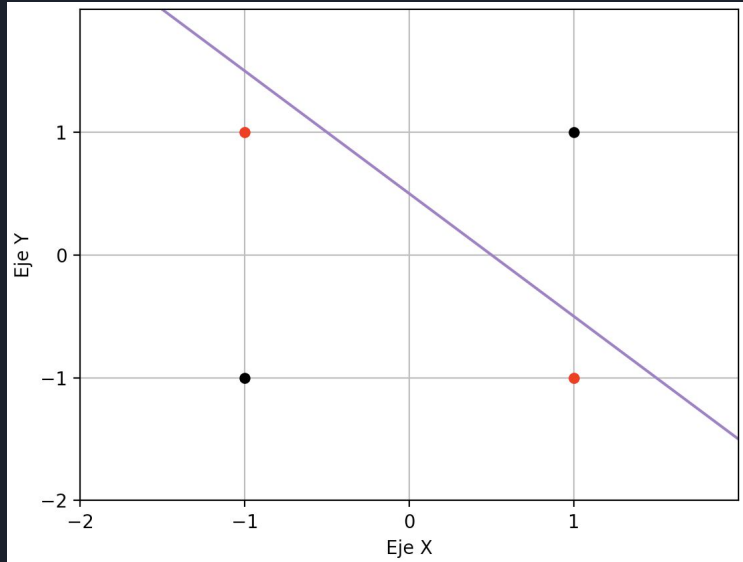
Función lógica “Y”

- Error mínimo: 0.0
- W mínimo: [0.4, 0.4, -0.2]
- Función: $w(x) = (w_1 * x + w_2) / -w_0$

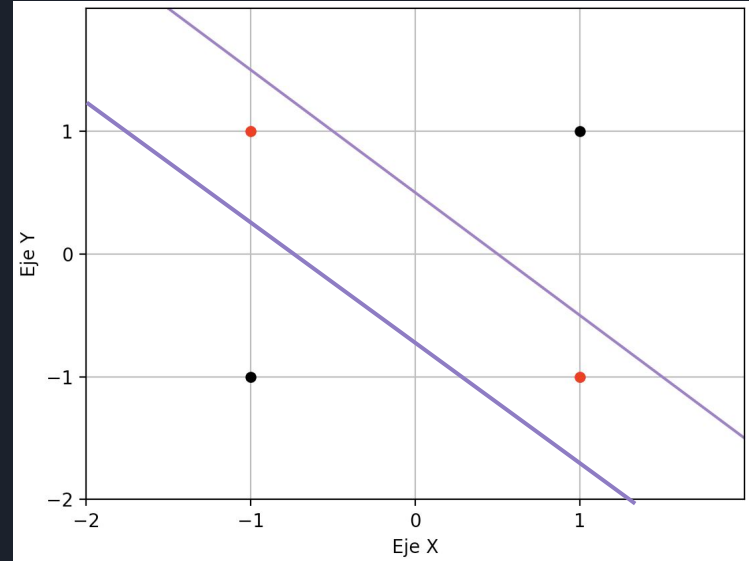


Función lógica "O exclusivo"

Resultado obtenido:



Ejemplo de resultado esperado:




→ Error mínimo:

2.0

→ W mínimo: [0.4 ,
0.4, -0.2]

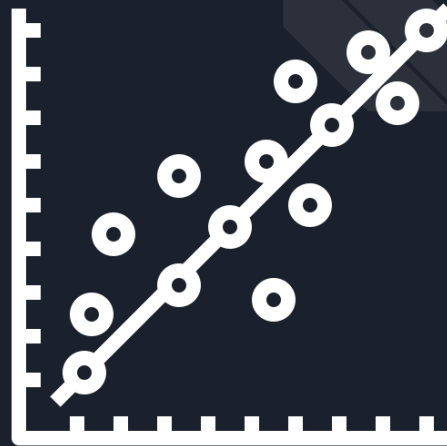
→ Función: $w(x) = (w_1 * x + w_2) / -w_0$



¿Qué puede decir acerca de los problemas que puede resolver el perceptron simple escalón en relación a la resolución de los problemas que se le pidió que haga que el perceptron aprenda?

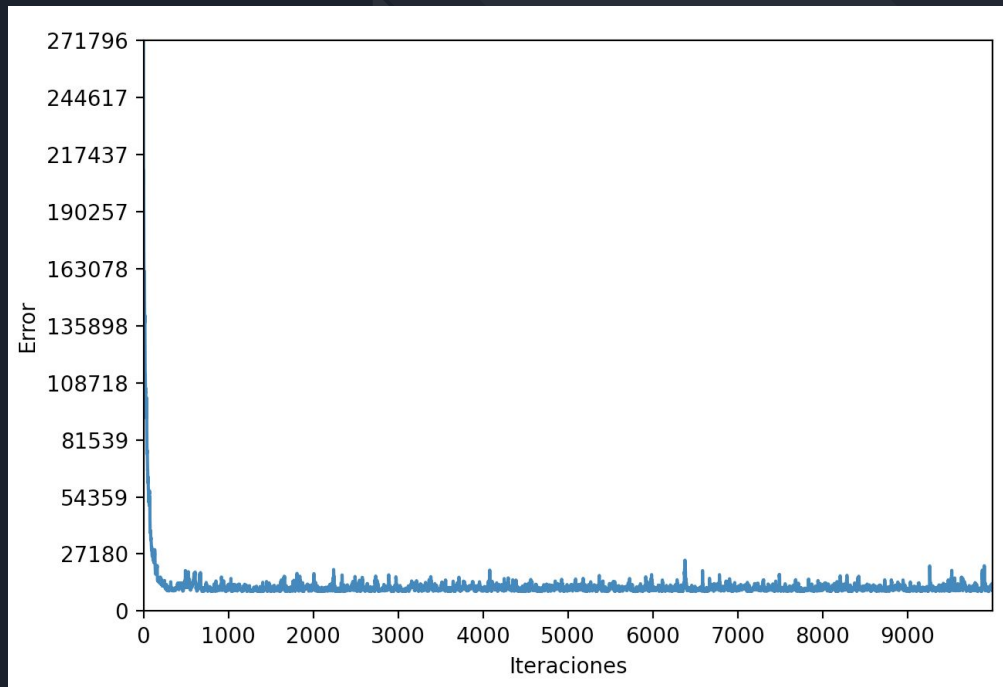


Perceptron Simple Lineal y No Lineal



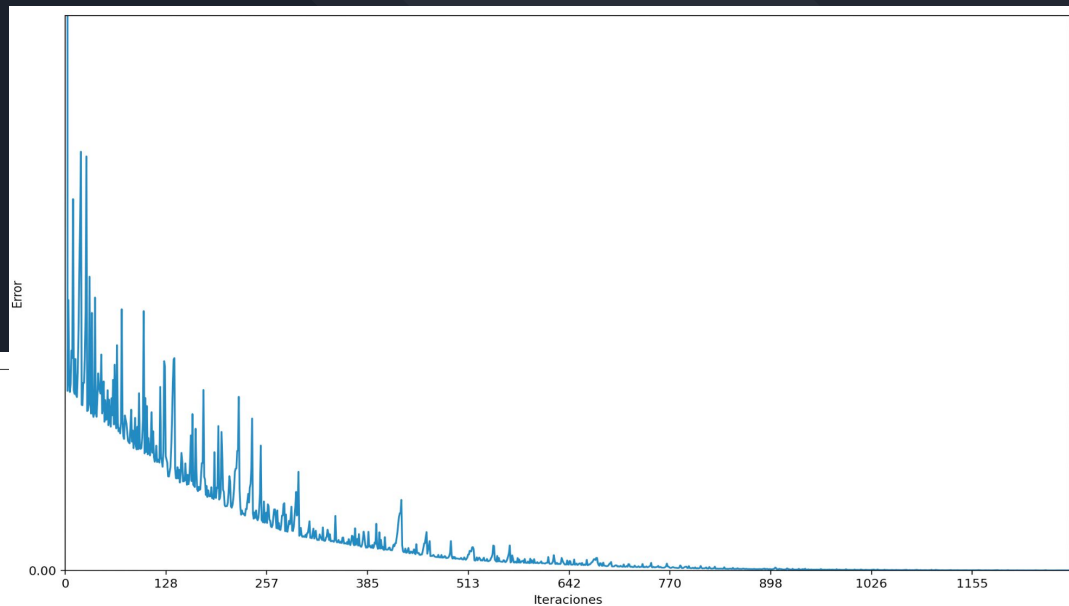
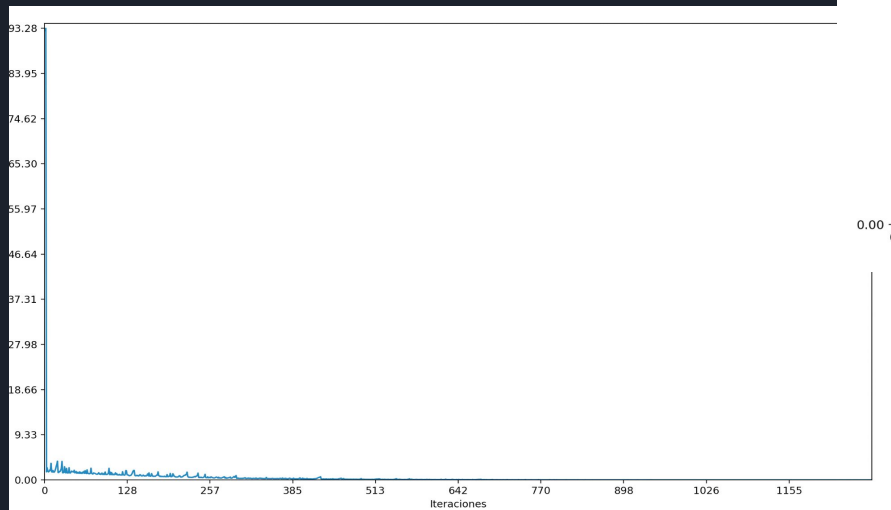
Perceptron Simple Lineal

- Error mínimo: 9263.239734144
- W mínimo: [6.83699289, 7.01597567, 5.9629689, 42.58681381]



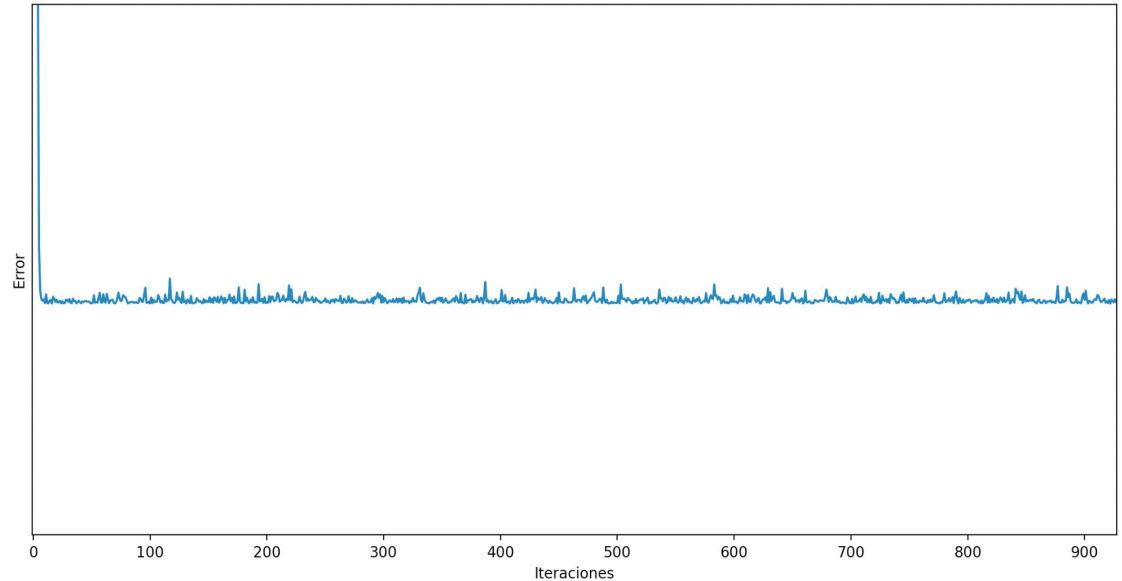
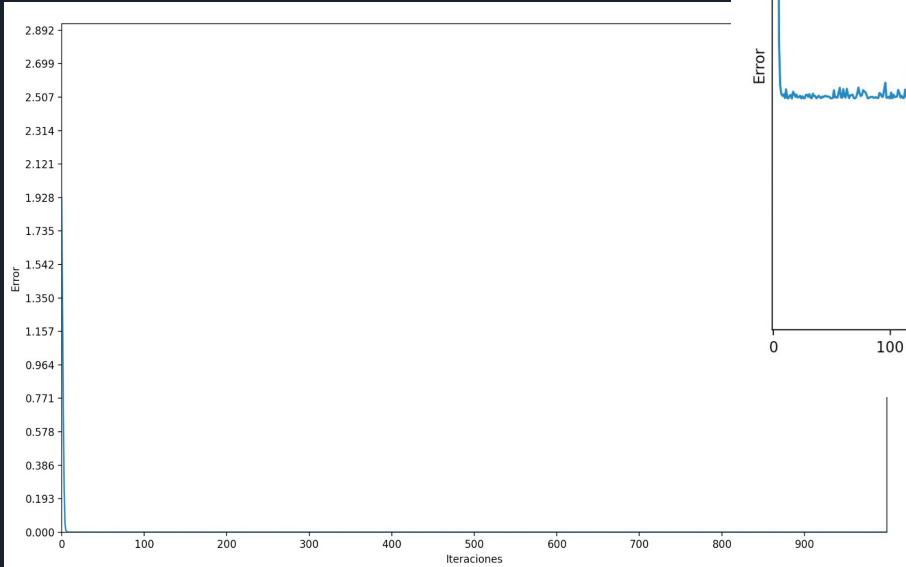
Perceptron Simple Lineal

- Error mínimo: 0.0009543617054370596
- W mínimo: [0.49817003, 0.49817003, 0.02543424]
- Entrada: { {0,0}, {1,1}, {2,2}, {3,3}, {4,4}, {5,5}, {6,6}, {7,7}, {8,8}, {9,9} }
- Salida esperada: {0,1,2,3,4,5,6,7,8,9}



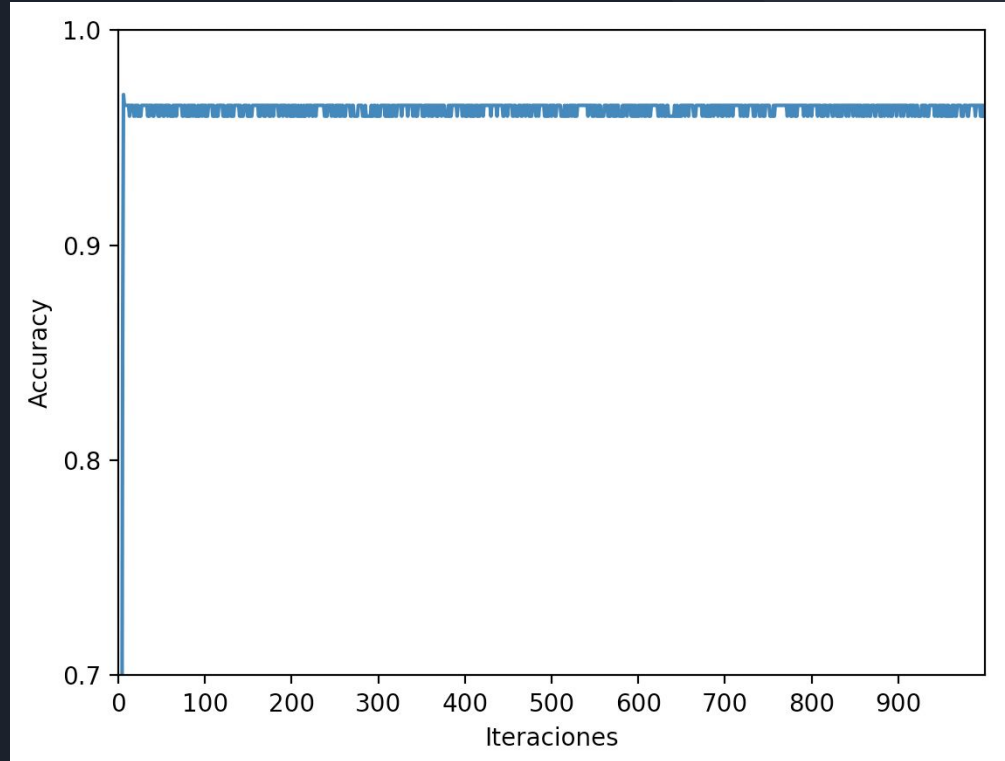
Perceptron Simple No Lineal

- Error mínimo: 0.0002951548291614954
- Con momentum
- Con función logística



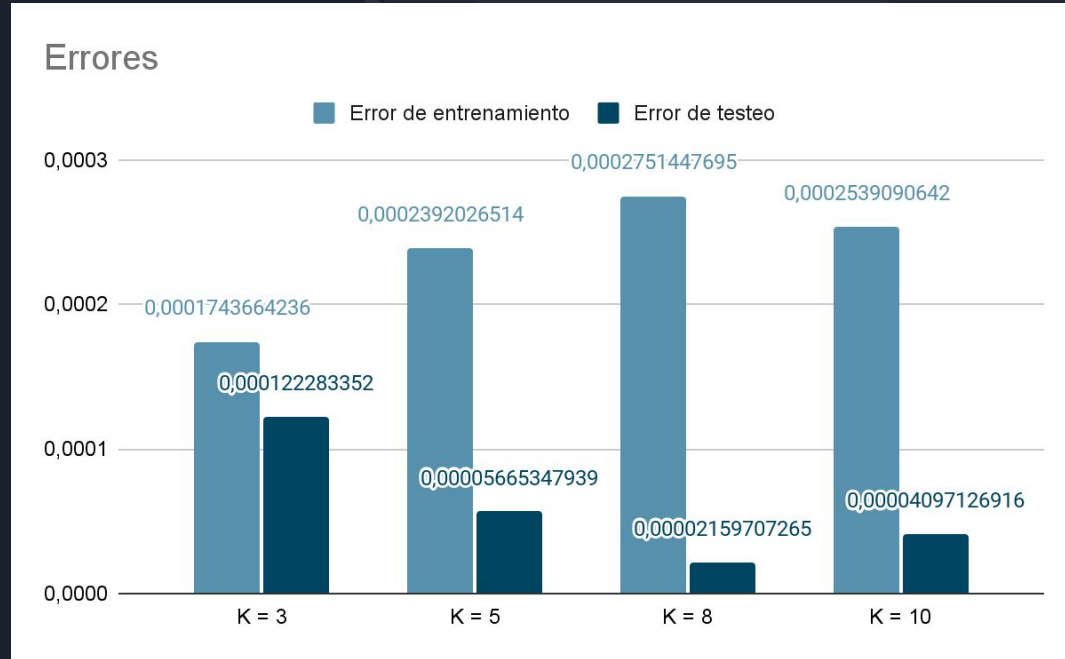
Perceptron Simple No Lineal - Accuracy

→ Accuracy: 0.965



Perceptron Simple No Lineal con Testeo

- $k = 3$
 - ◆ Porcentaje del conjunto entrenado: 66.5
- $k = 5$
 - ◆ Porcentaje del conjunto entrenado: 80.0
- $k = 8$
 - ◆ Porcentaje del conjunto entrenado: 87.5
- $k = 10$
 - ◆ Porcentaje del conjunto entrenado: 90.0





¿Cómo podría escoger el mejor conjunto de
entrenamiento?

¿Como podría evaluar la máxima capacidad de
generalización del perceptron para este conjunto
de datos?



Perceptron Multicapa





1

Función lógica “O exclusivo”

Entrada: $\{\{-1,1\},\{1,-1\},\{-1,-1\},\{1,1\}\}$ - Salida esperada: $\{1,1,-1,-1\}$

2

Conjunto de números del 0 al 9

- Capas ocultas: 1
- Nodos de las capas ocultas: 10

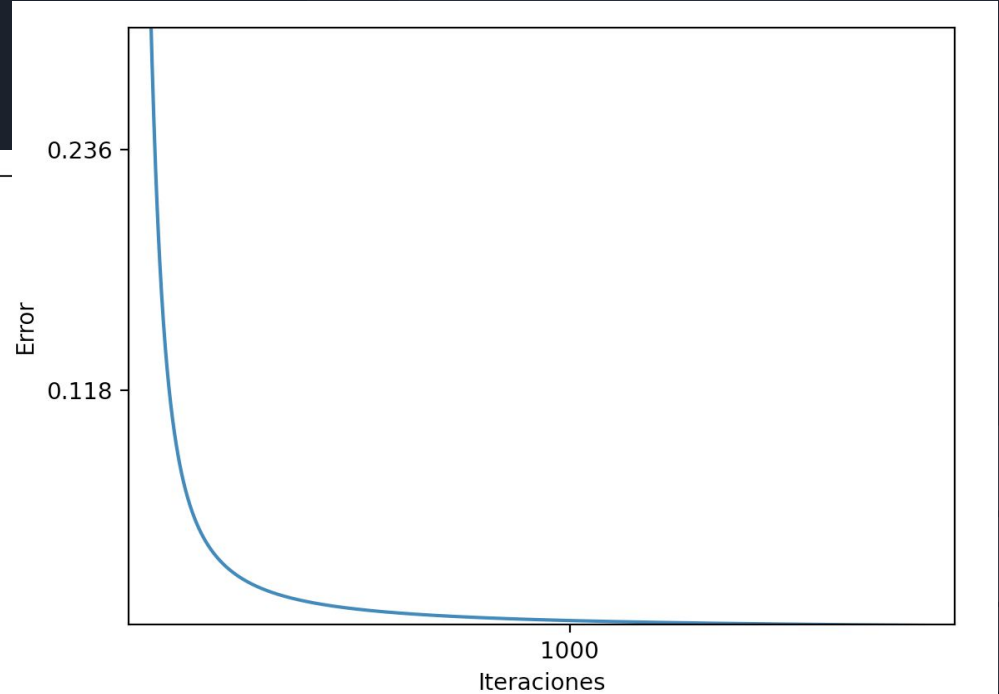
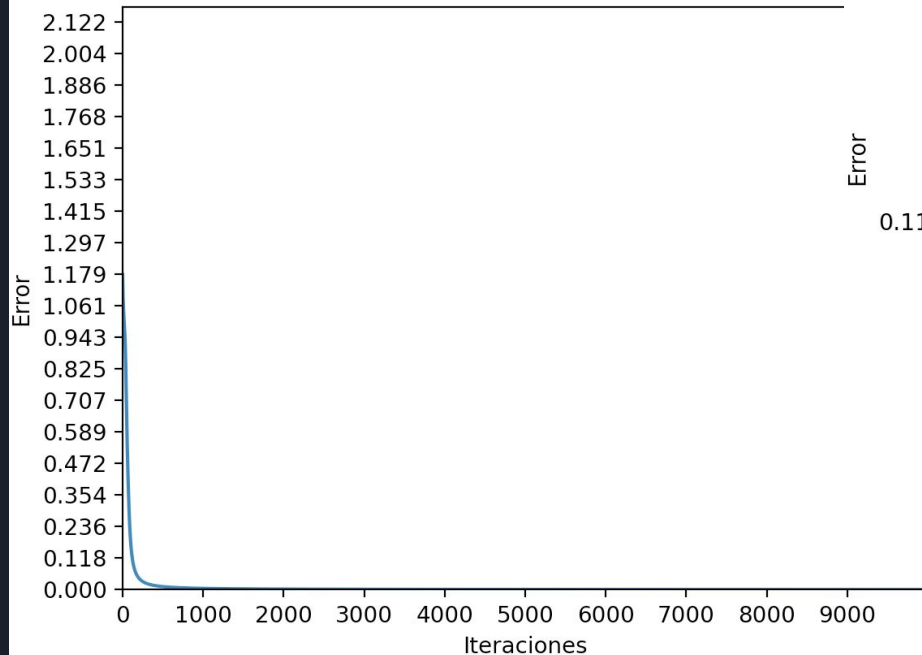
3

Conjunto de números del 0 al 9 con 10 unidades de salida

- Capas ocultas: 1
- Nodos de las capas ocultas: 10

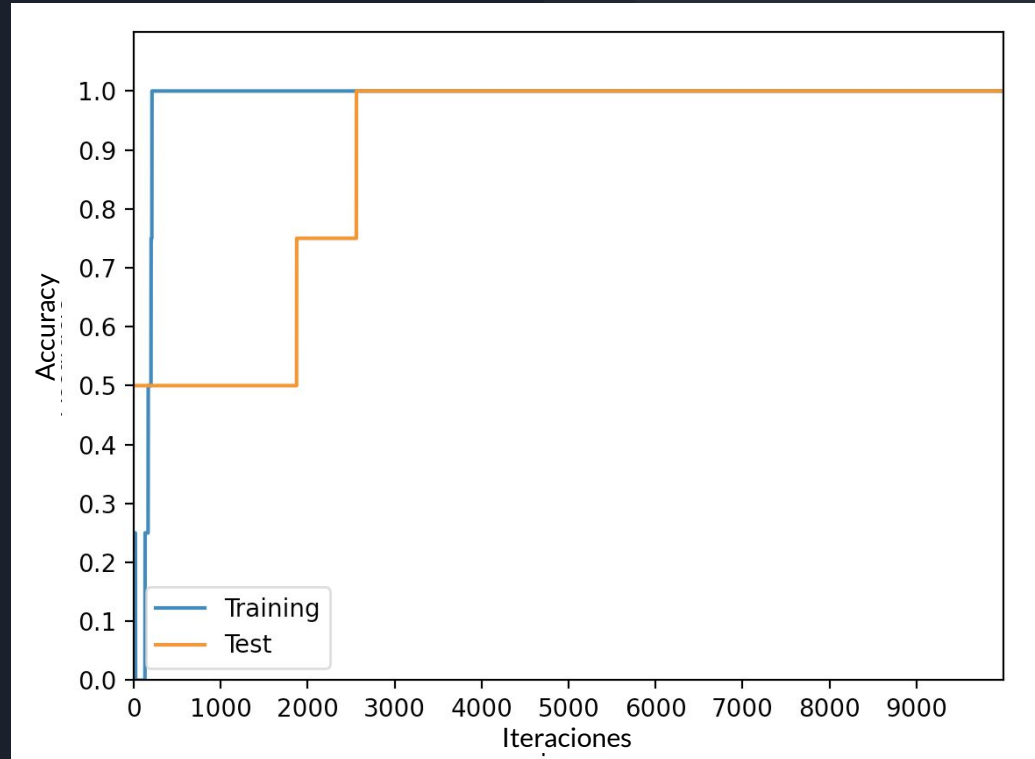
Función lógica “O exclusivo”

- Error mínimo: 0.0004116362794672741
- Capas ocultas: 1
- Nodos de las capas ocultas: 5



Función lógica “O exclusivo”

- Capas ocultas: 1
- Nodos de las capas ocultas: 5



Conjunto de números del 0 al 9 con $k = 3$

→ Entrenamiento:

3 ~ impar

4 ~ par

5 ~ impar

6 ~ par

7 ~ impar

8 ~ par

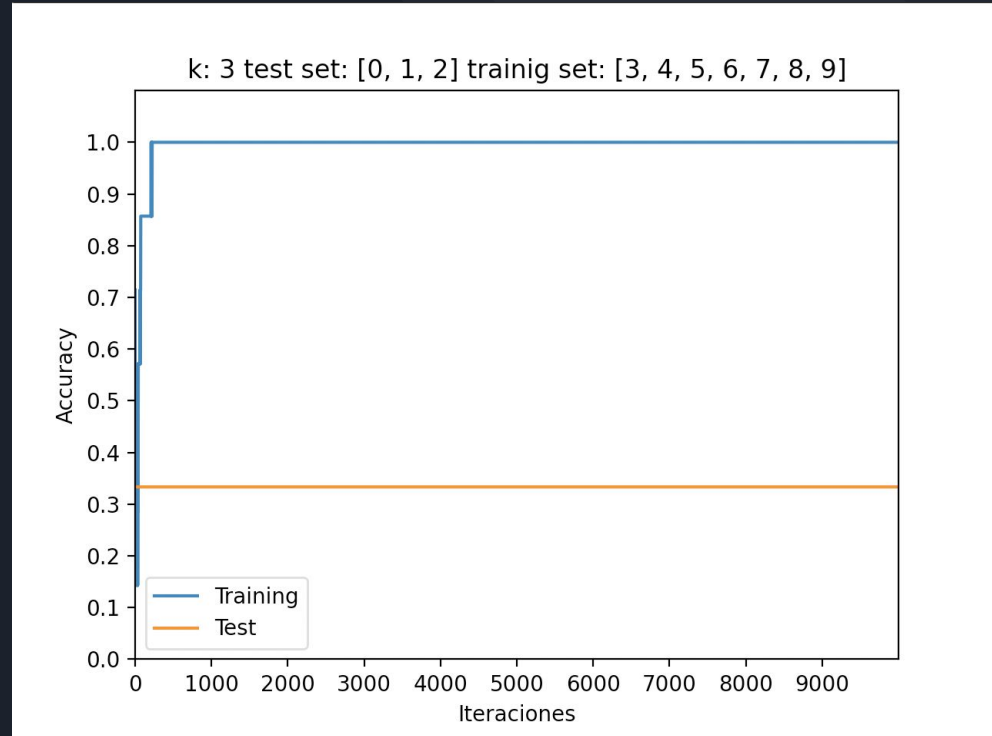
9 ~ impar

→ Testeo:

0 ~ impar

1 ~ par

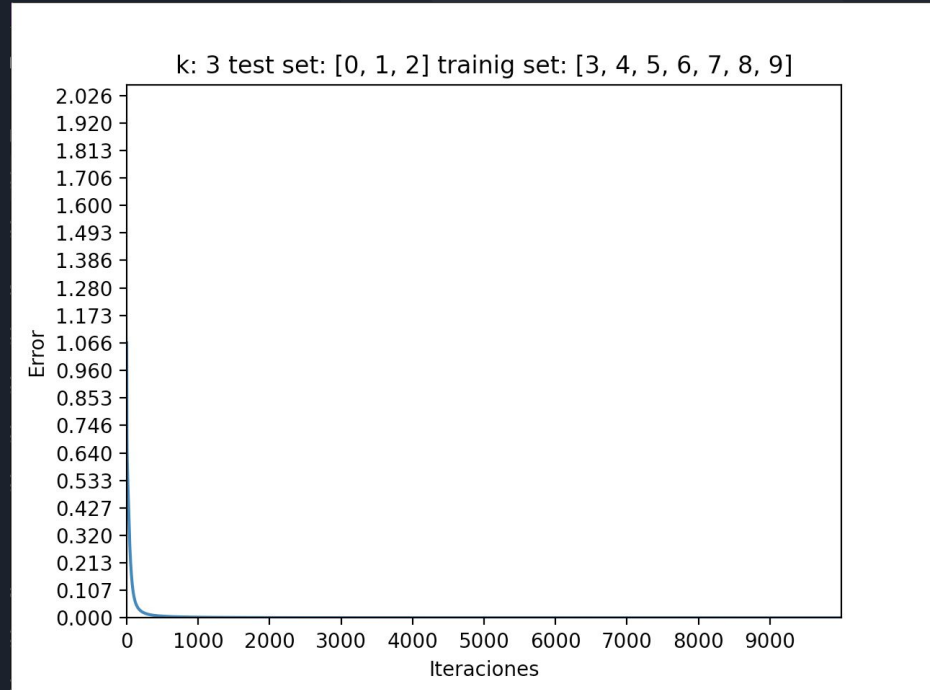
2 ~ impar



Conjunto de números del 0 al 9 con $k = 3$

→ Error mínimo de entrenamiento:
0.00018178654096406206

→ Error mínimo de testeo:
2.0312153254825724



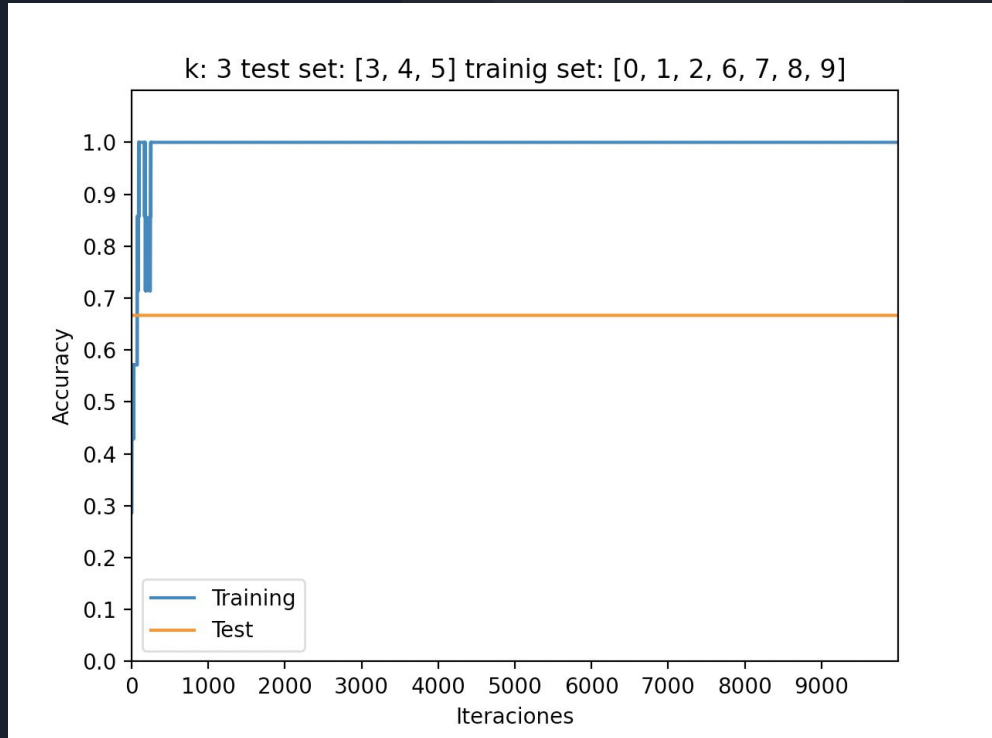
Conjunto de números del 0 al 9 con $k = 3$

→ Entrenamiento:

0 ~ par
1 ~ impar
2 ~ par
6 ~ par
7 ~ impar
8 ~ par
9 ~ impar

→ Testeo:

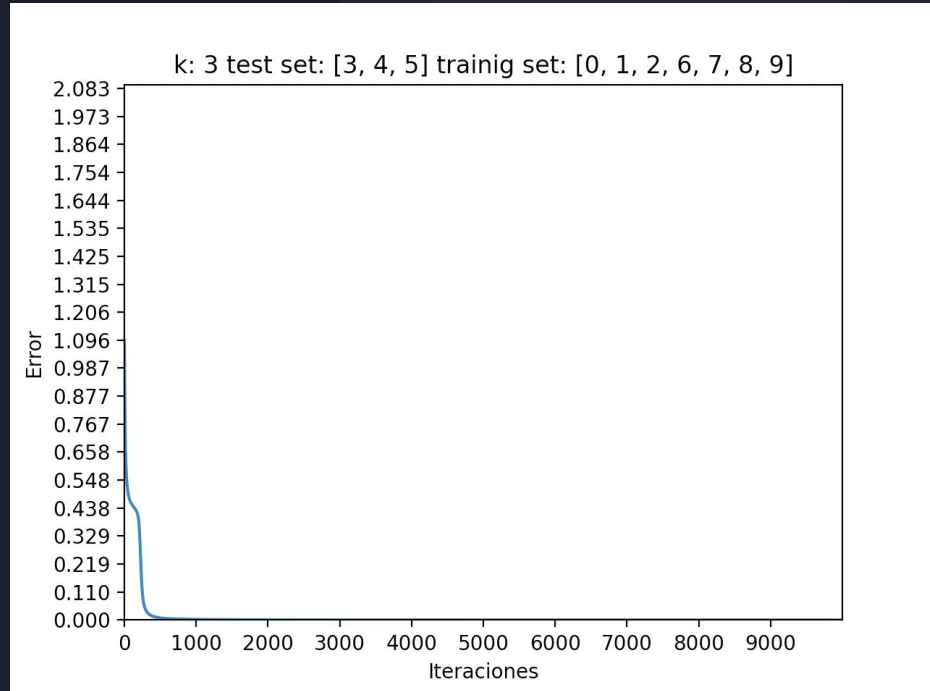
3 ~ par
4 ~ impar
5 ~ par



Conjunto de números del 0 al 9 con $k = 3$

→ Error mínimo de entrenamiento:
0.00013497408681127112

→ Error mínimo de testeo:
1.2409225852010746



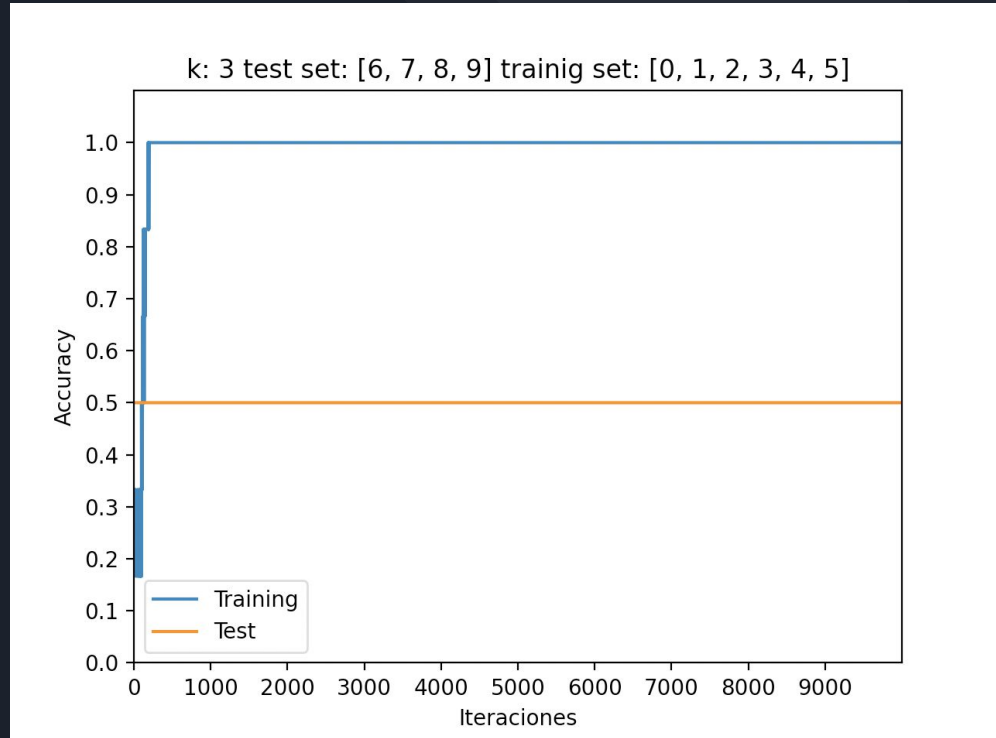
Conjunto de números del 0 al 9 con $k = 3$

→ Entrenamiento:

0 ~ par
1 ~ impar
2 ~ par
3 ~ impar
4 ~ par
5 ~ impar

→ Testeo:

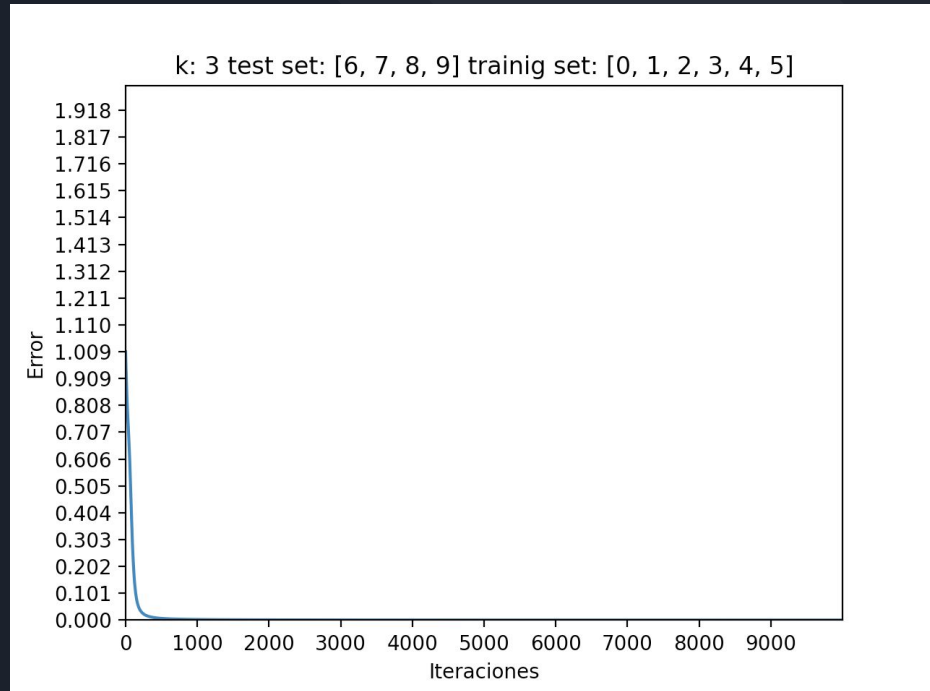
6 ~ impar
7 ~ par
8 ~ impar
9 ~ impar



Conjunto de números del 0 al 9 con $k = 3$

→ Error mínimo de entrenamiento:
0.00015531623333281822

→ Error mínimo de testeo:
1.025426776402205



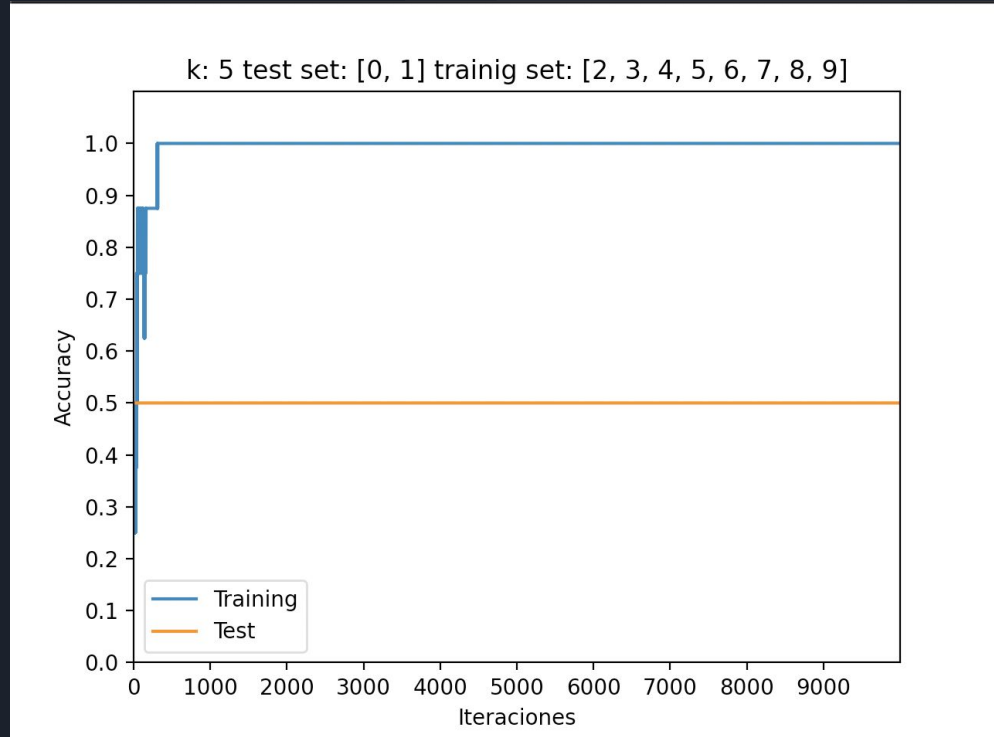
Conjunto de números del 0 al 9 con $k = 5$

→ Entrenamiento:

2 ~ par
3 ~ impar
4 ~ par
5 ~ impar
6 ~ par
7 ~ impar
8 ~ par
9 ~ impar

→ Testeo:

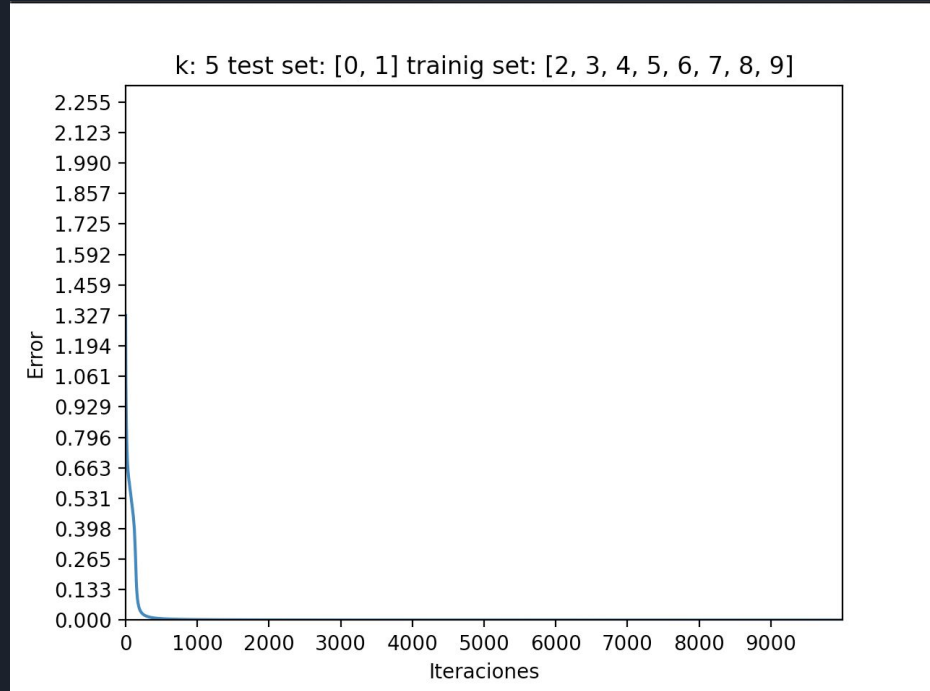
0 ~ impar
1 ~ impar



Conjunto de números del 0 al 9 con $k = 5$

→ Error mínimo de entrenamiento:
0.0001339084330777915

→ Error mínimo de testeo:
1.0071833378829564



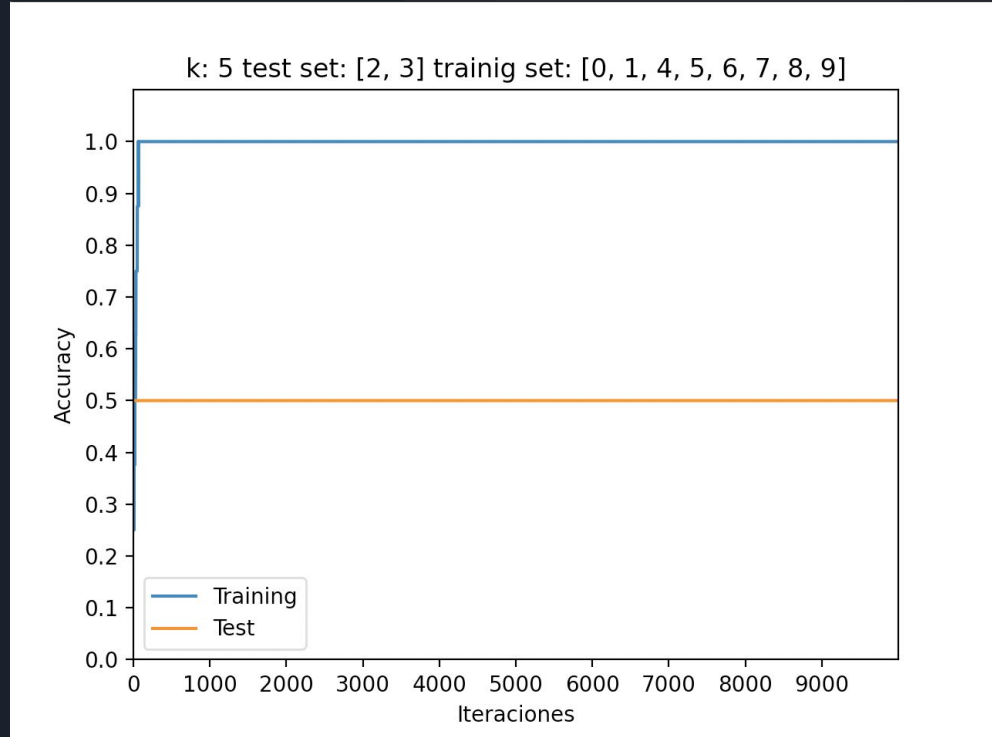
Conjunto de números del 0 al 9 con $k = 5$

→ Entrenamiento:

0 ~ par
1 ~ impar
4 ~ par
5 ~ impar
6 ~ par
7 ~ impar
8 ~ par
9 ~ impar

→ Testeo:

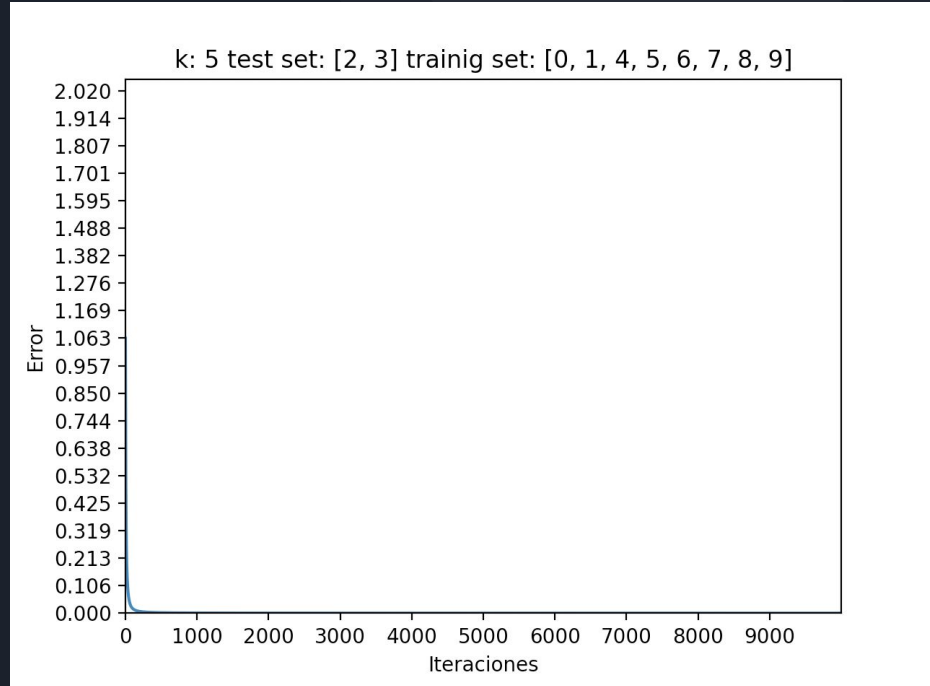
2 ~ impar
3 ~ par



Conjunto de números del 0 al 9 con $k = 5$

→ Error mínimo de entrenamiento:
6.960838194208072e-05

→ Error mínimo de testeo:
0.7148369395752032



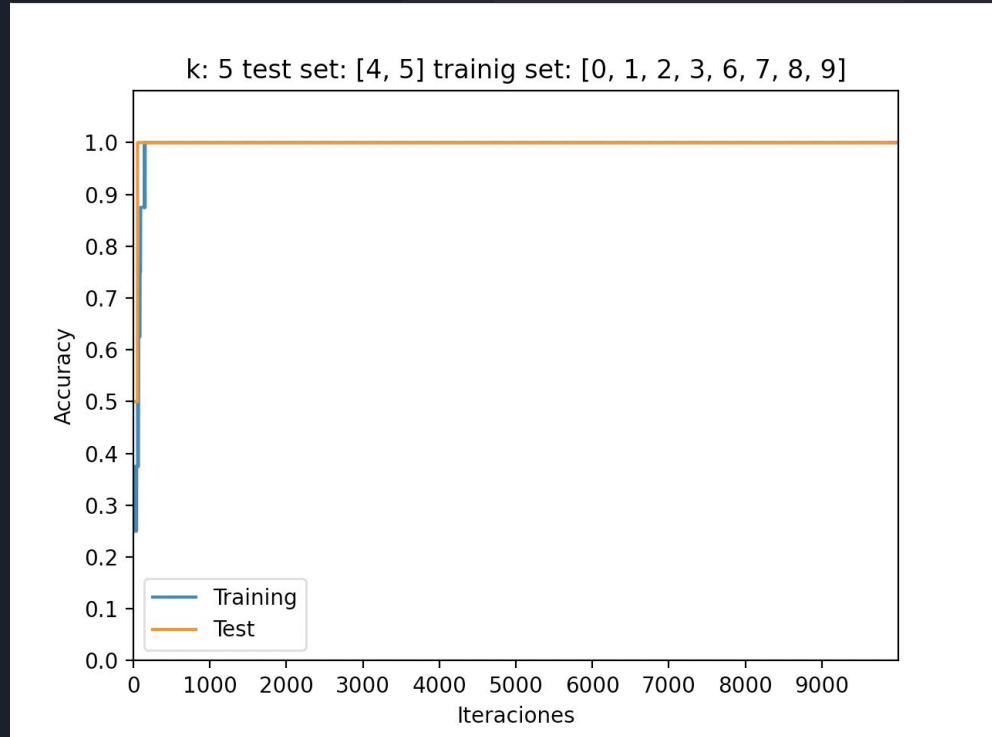
Conjunto de números del 0 al 9 con $k = 5$

→ Entrenamiento:

0 ~ par
1 ~ impar
2 ~ par
3 ~ impar
6 ~ par
7 ~ impar
8 ~ par
9 ~ impar

→ Testeo:

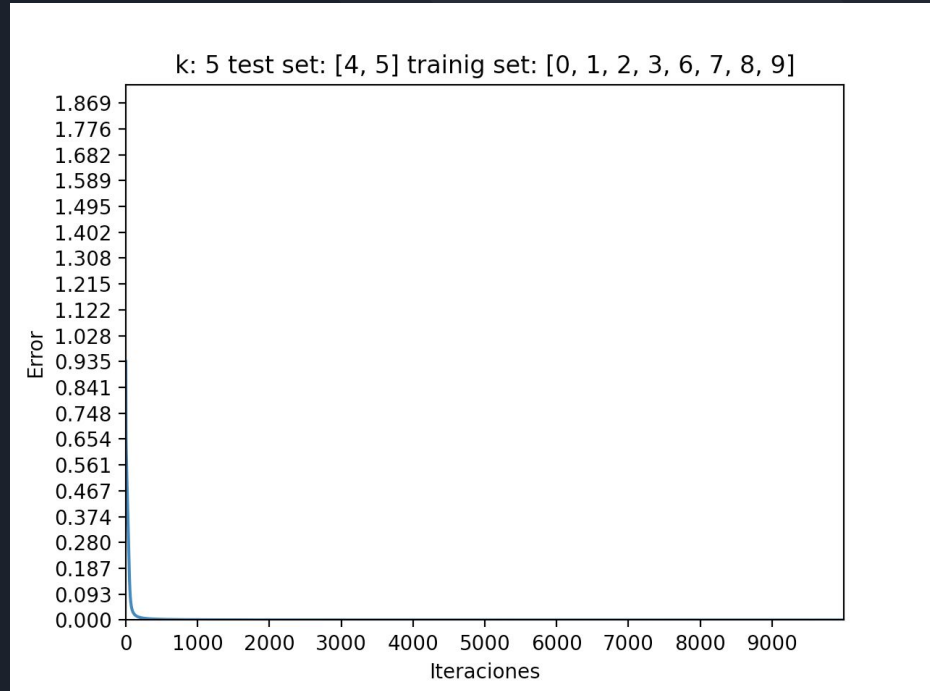
4 ~ par
5 ~ impar



Conjunto de números del 0 al 9 con $k = 5$

→ Error mínimo de entrenamiento:
8.622194207397718e-05

→ Error mínimo de testeo:
1.3139679779959954



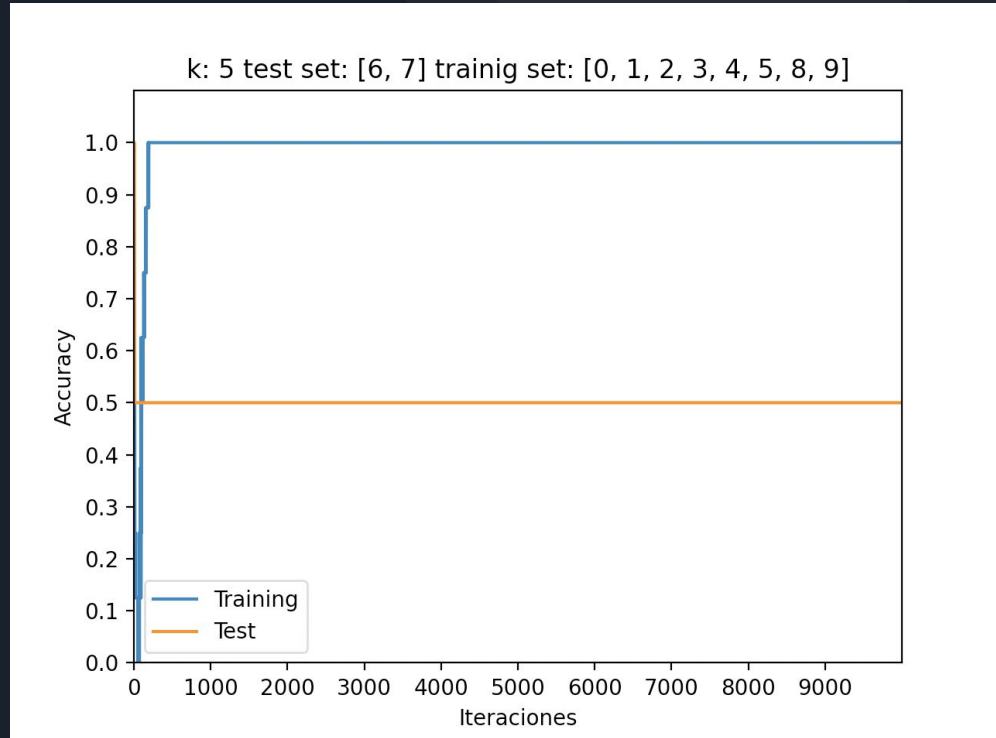
Conjunto de números del 0 al 9 con $k = 5$

→ Entrenamiento:

0 ~ par
1 ~ impar
2 ~ par
3 ~ impar
4 ~ par
5 ~ impar
8 ~ par
9 ~ impar

→ Testeo:

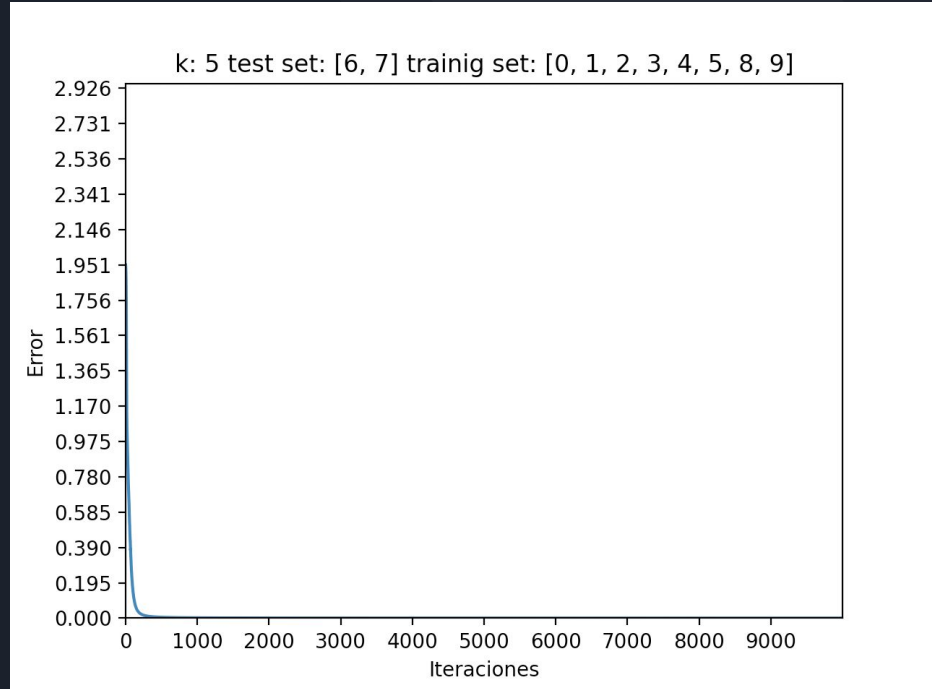
6 ~ impar
7 ~ impar



Conjunto de números del 0 al 9 con $k = 5$

→ Error mínimo de entrenamiento:
0.00013562455504357112

→ Error mínimo de testeo:
1.0324398192462567



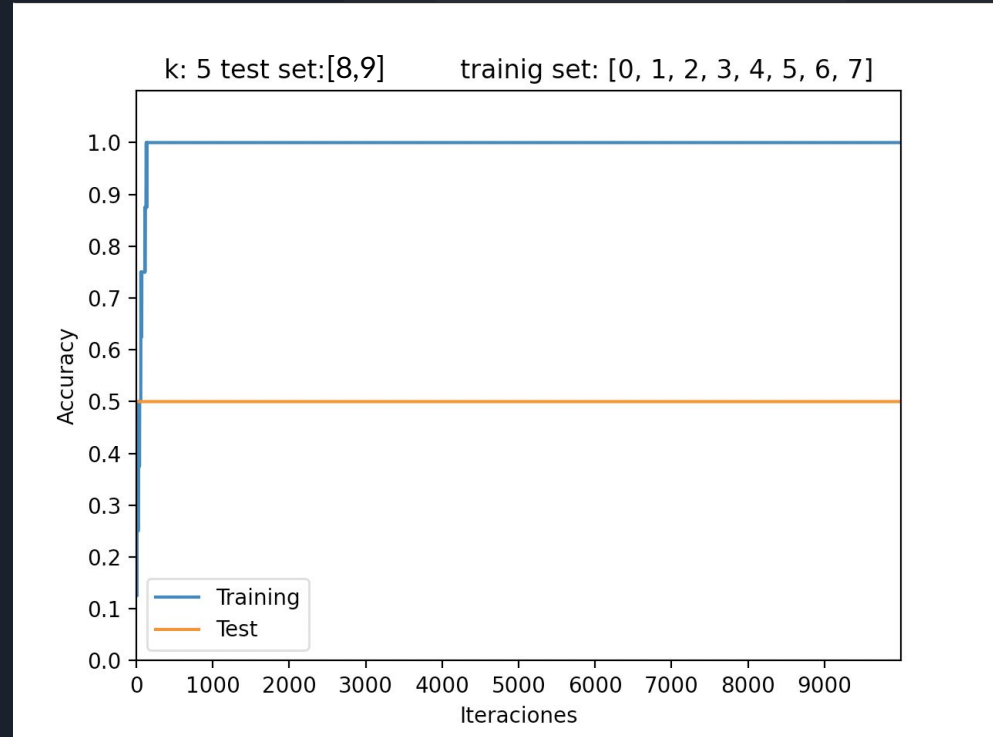
Conjunto de números del 0 al 9 con $k = 5$

→ Entrenamiento:

0 ~ par
1 ~ impar
2 ~ par
3 ~ impar
4 ~ par
5 ~ impar
6 ~ par
7 ~ impar

→ Testeo:

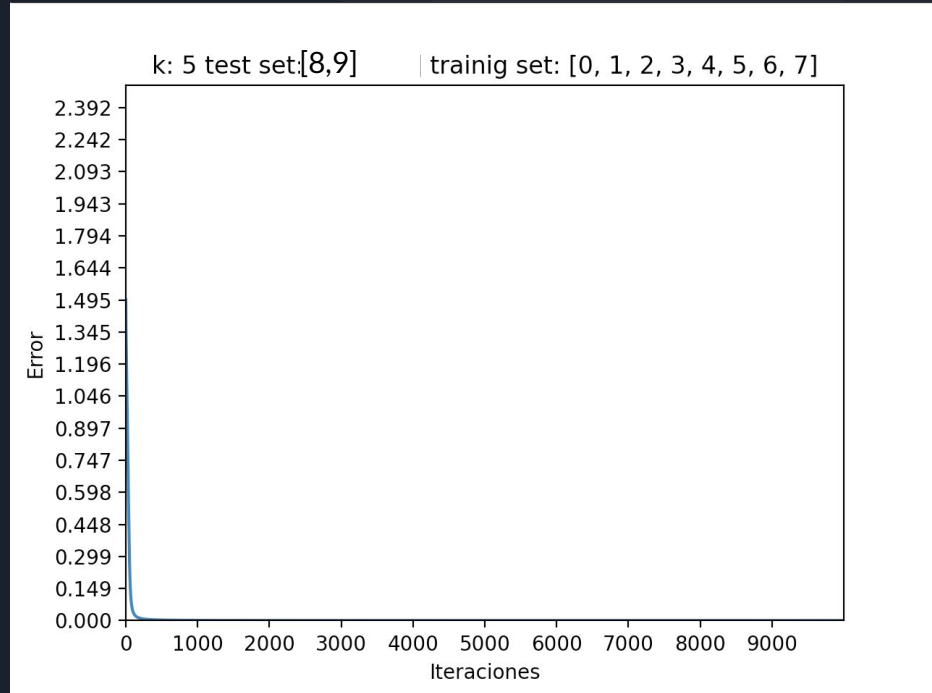
8 ~ par
9 ~ impar



Conjunto de números del 0 al 9 con $k = 5$

→ Error mínimo de entrenamiento:
7.612876333382601e-05

→ Error mínimo de testeo:
0.10193508693951717





¿Que podría decir acerca de la capacidad
para generalizar de la red?

Conjunto de números del 0 al 9 con 10 unidades de salida

Esperado	Obtenido
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	0
8	3
9	3

- Entrenamiento: 0 a 6
- Testeo : 7 a 9
- Capas ocultas: 1
- Nodos de las capas ocultas: 10

Conjunto de números del 0 al 9 con 10 unidades de salida

Esperado	Obtenido
0	0
1	1
2	2
3	3
4	4
5	0
6	3
7	2
8	3
9	3

- Entrenamiento: 0 a 4
- Testeo : 5 a 9
- Capas ocultas: 1
- Nodos de las capas ocultas: 10

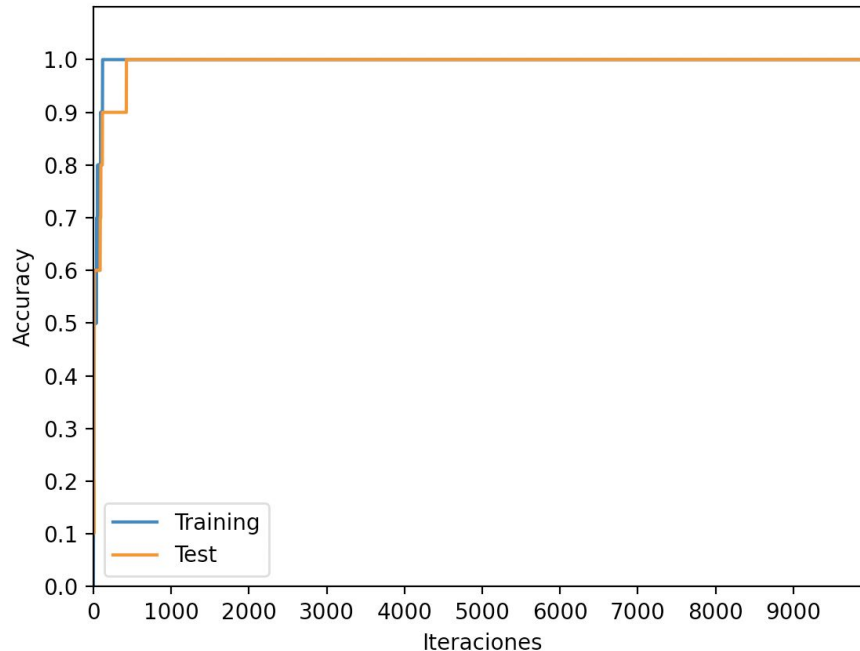
Conjunto de números del 0 al 9 con 10 unidades de salida con ruido

Testeo

Esperado	Obtenido
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

- Ruido: 0.02
- Capas ocultas: 1
- Nodos de las capas ocultas: 10

Conjunto de números del 0 al 9 con 10 unidades de salida con ruido



- Entrenamiento: 0 a 9
- Ruido: 0.02
- Capas oculta: 1
- Nodos de las capas ocultas: 10

Conjunto de números del 0 al 9 con 10 unidades de salida con ruido

Testeo 1

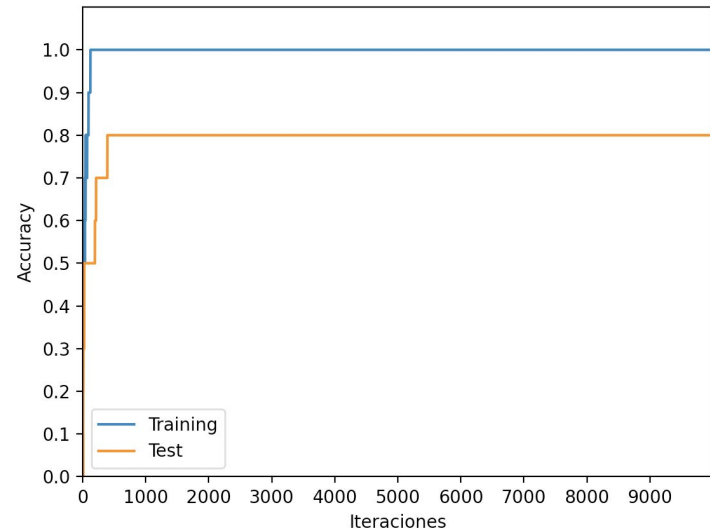
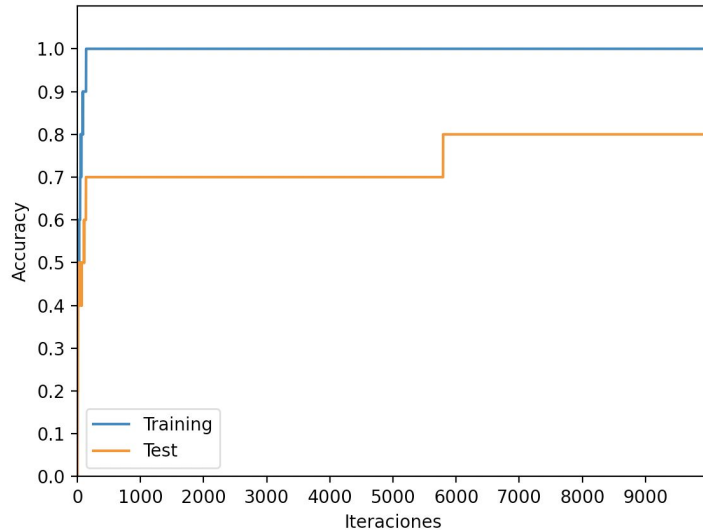
Esperado	Obtenido
0	0
1	1
2	5
3	8
4	4
5	5
6	6
7	7
8	8
9	9

Testeo 2

Esperado	Obtenido
0	0
1	5
2	2
3	3
4	4
5	5
6	5
7	7
8	8
9	9

- Ruido: 0.06
- Capas ocultas: 1
- Nodos de las capas ocultas: 10

Conjunto de números del 0 al 9 con 10 unidades de salida con ruido



- Ruido: 0.06
- Capas ocultas: 1
- Nodos de las capas ocultas: 10

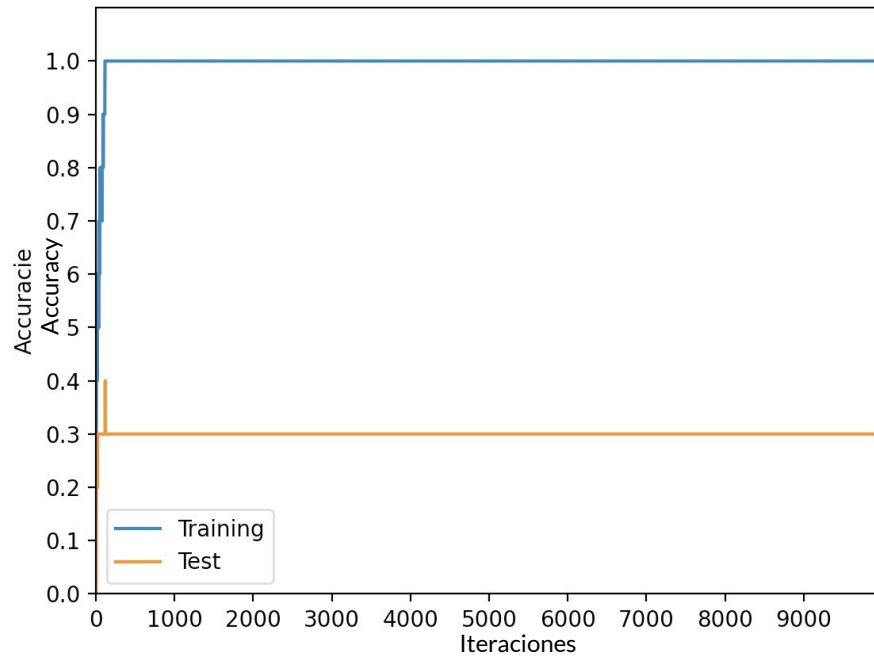
Conjunto de números del 0 al 9 con 10 unidades de salida con ruido

Testeo

Esperado	Obtenido
0	4
1	7
2	7
3	7
4	4
5	5
6	3
7	7
8	7
9	4

- Ruido: 0.1
- Capas ocultas: 1
- Nodos de las capas ocultas: 10

Conjunto de números del 0 al 9 con 10 unidades de salida con ruido



- Ruido: 0.1
- Capas ocultas: 1
- Nodos de las capas ocultas: 10



Problemas



→ Tangente a Logística para el Perceptron No Lineal