Trabajo Práctico N°3
Perceptrón Simple y Multicapa

Grupo 5

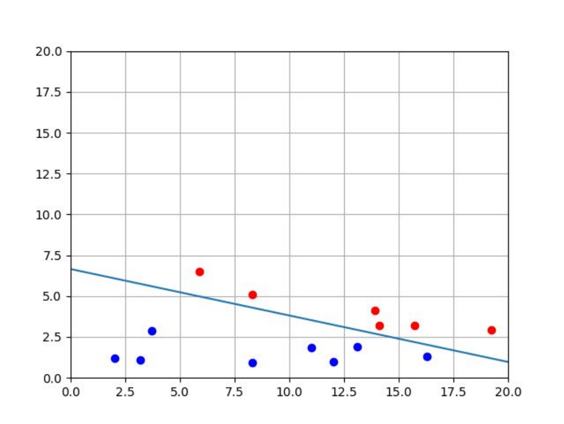
Gonzalo Baliarda
Franco Nicolás Estevez
Ezequiel Agustin Perez
Leandro Ezequiel Rodriguez
Lucas Agustín Vittor

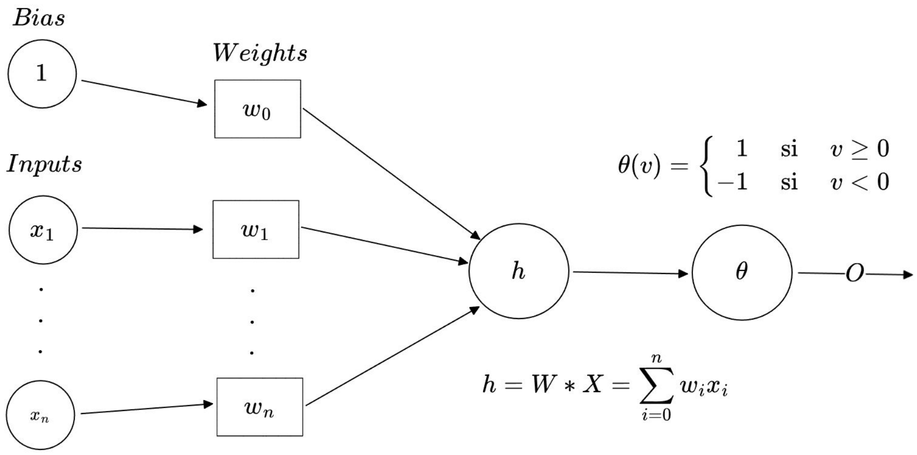
Ejercicio 1

Perceptrón Simple Escalón

Perceptrón Escalón

Permite resolver problemas de **clasificación binaria**, en los que los grupos sean **linealmente separables**.





AND y XOR: arquitectura

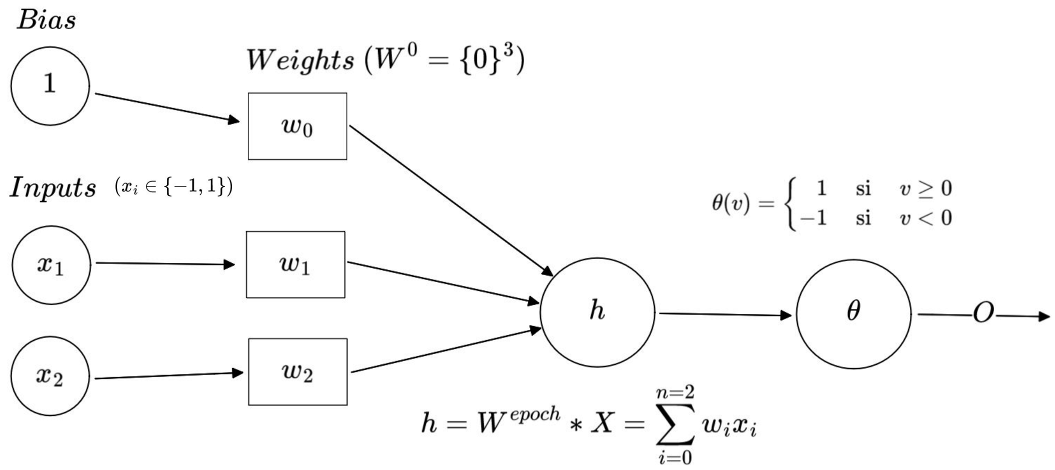
$$E(O) = \sum_{\mu=0}^{p-1} |\zeta^{\mu} - O^{\mu}|$$

$$convergencia \iff E = 0$$

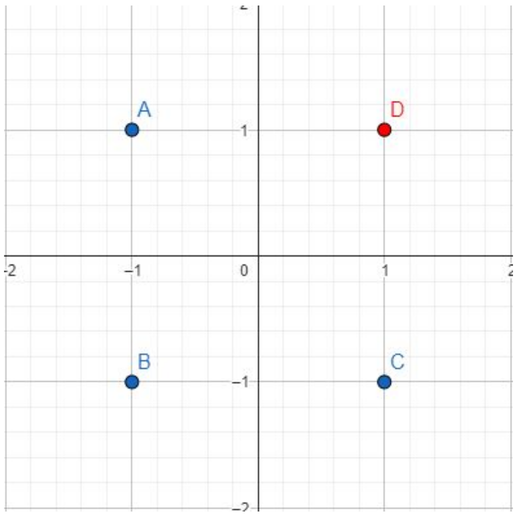
$$y = -\frac{w_1 x + w_0}{w_2}$$

$$optimization = false$$

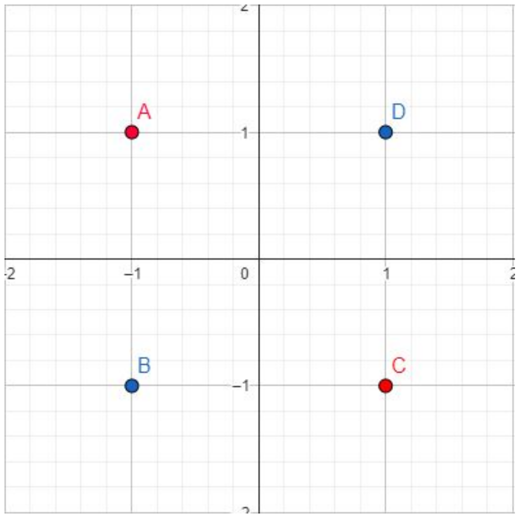
$$\eta = 0.01$$



AND y XOR: resultado esperado

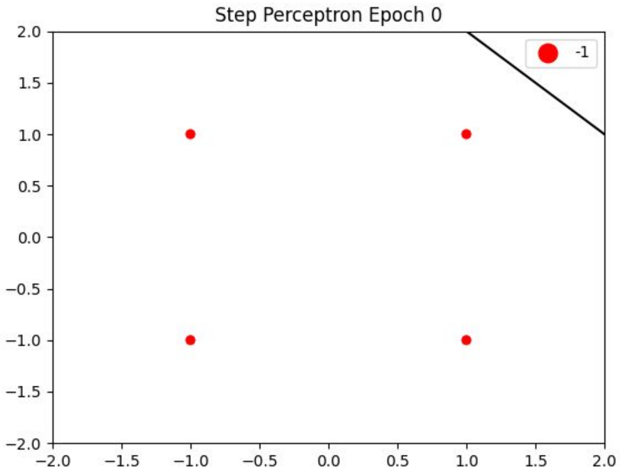


AND



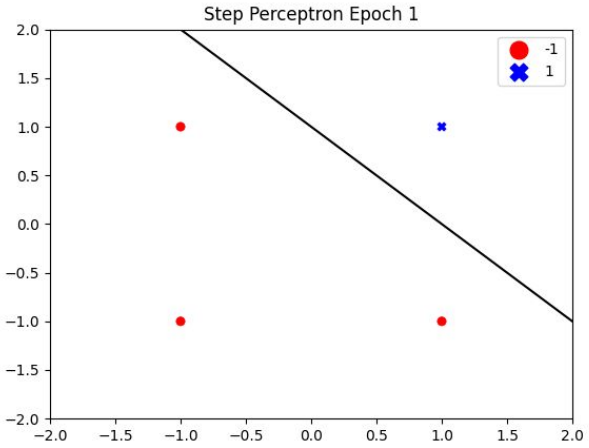
XOR

AND



Step Perceptron Epoch 0
-1
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

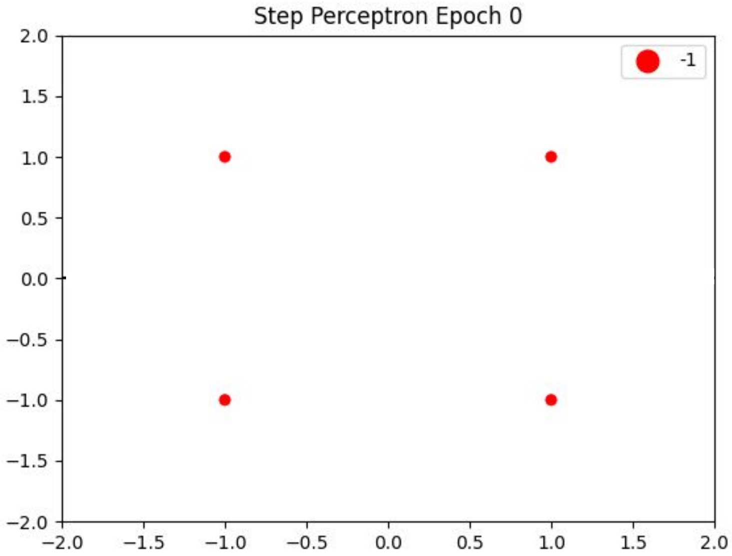
$$y = 3 - x$$



Step Perceptron Epoch 1
-1
1
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

$$y = 1 - x$$

XOR



$$W^{2k} = [0, 0, 0]$$

$$W^{2k+1} = [-0.04, 0, 0]$$

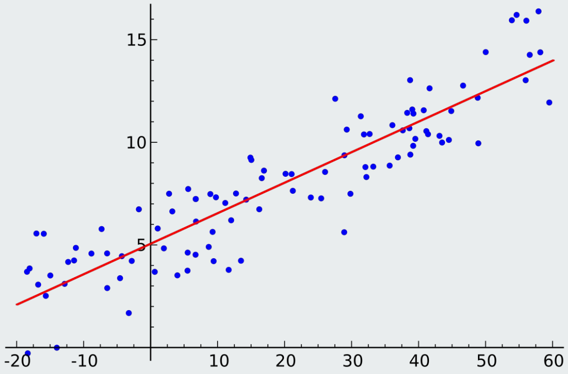
$$\forall k : E(O) = 4$$

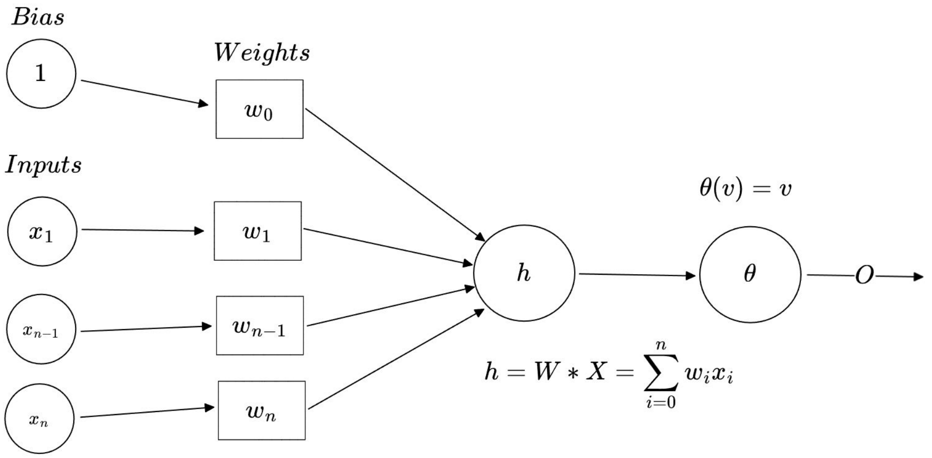
Ejercicio 2

Perceptrón Simple Lineal y No-Lineal

Perceptrón Lineal

Permite resolver problemas de regresión, en los que haya una **relación cuasi lineal entre inputs y outputs**.



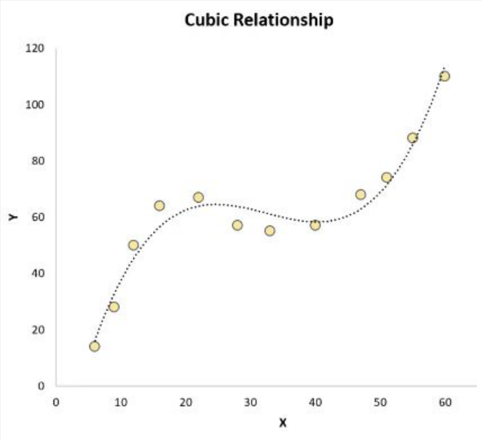


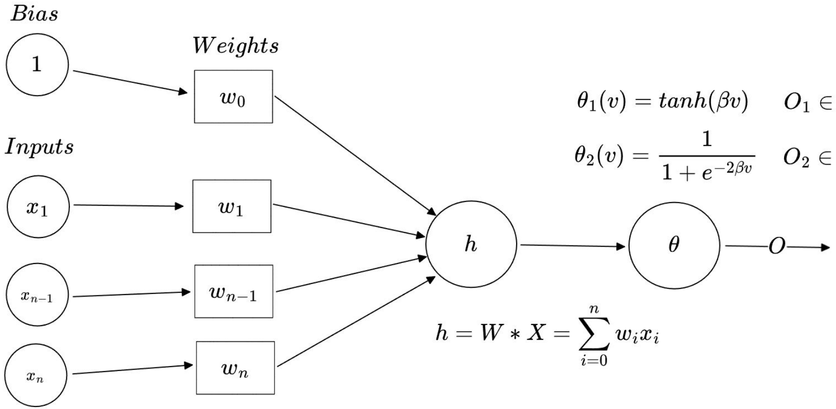
$$O \in \mathbb{R}$$

$$E(O) = MSE = \frac{1}{p} \sum_{\mu=0}^{p-1} (\zeta^\mu - O^\mu)^2$$

Perceptrón No Lineal

Permite resolver problemas de regresión, en los que haya una **relación no lineal entre inputs y outputs**.





$$\theta_1(v) = tanh(\beta v) \qquad O_1 \in [-1, 1]$$

$$\theta_2(v) = \frac{1}{1 + e^{-2\beta v}} \qquad O_2 \in [0, 1]$$

$$h = W * X = \sum_{i=0}^{n} w_i x_i$$

$$E(O) = MSE$$

Los outputs esperados se escalan según:

$$\zeta' = \frac{\zeta - Z_{min}}{Z_{max} - Z_{min}} (\theta_{max} - \theta_{min}) + \theta_{min}$$

Problema de Regresión

x_1	x_2	x_3	ζ
1.200	-0.800	0.000	21.755
1.200	0.000	-0.800	7.176
1.200	-0.800	1.000	43.045
0.000	1.200	-0.800	2.875
$\vdots$	$\vdots$	$\vdots$	$\vdots$

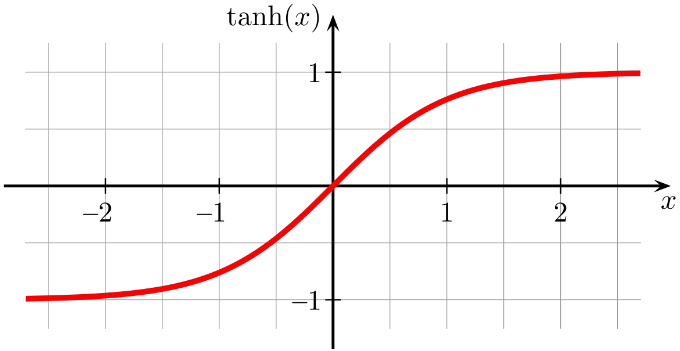
$$X = (x_1, x_2, x_3)$$

$$Z = (\zeta)$$

$$\eta = 0.001$$

$$convergencia \iff MSE \leq \frac{q}{100}(Z_{max} - Z_{min})$$

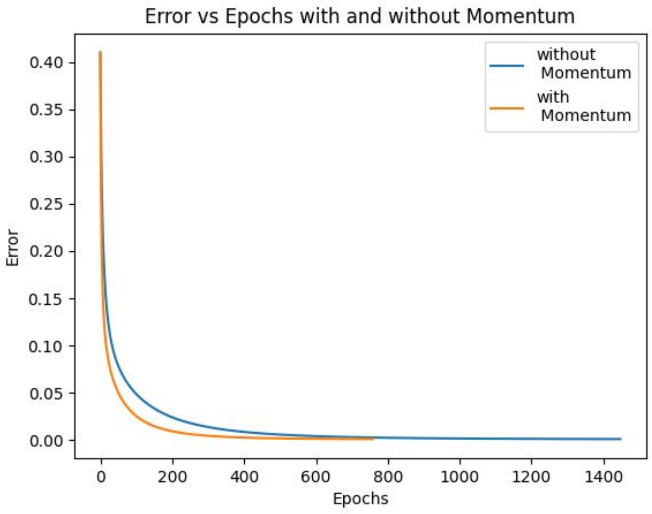
Regresión: perceptrón no lineal

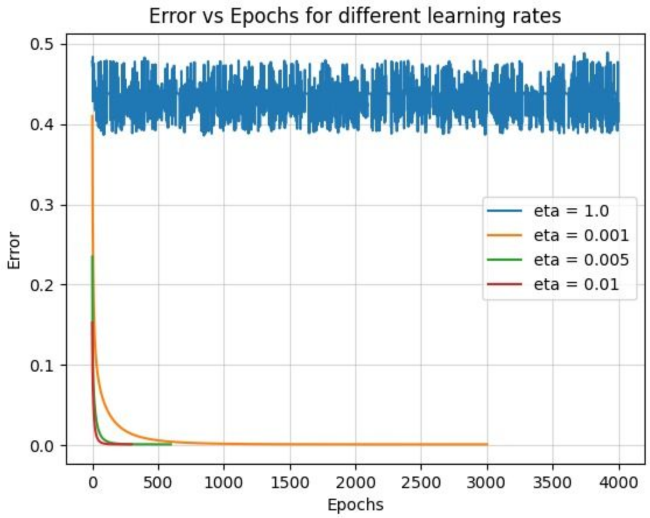


$$\theta(v) = tanh(v)$$

$$convergencia \iff MSE \leq \frac{0.05}{100}(Z'_{max} - Z'_{min})$$

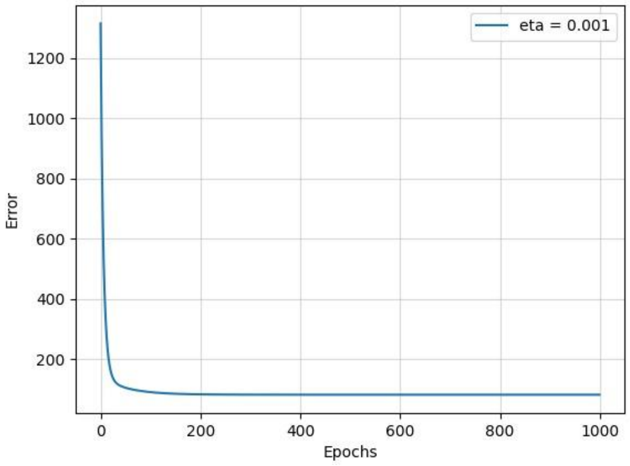
Regresión: perceptrón no lineal

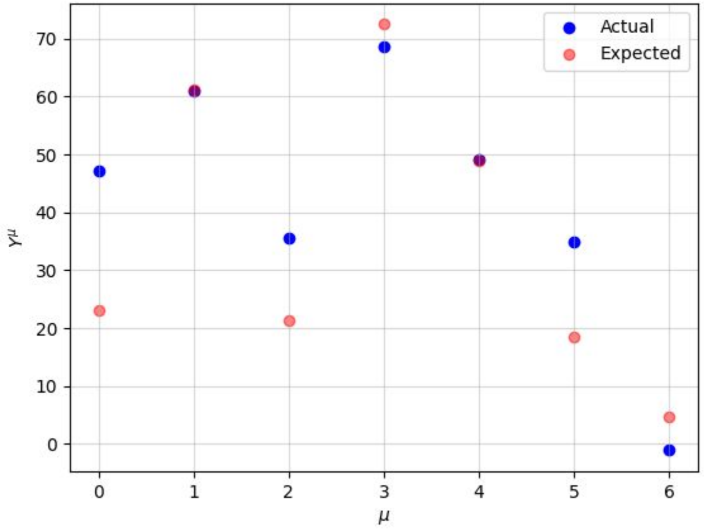




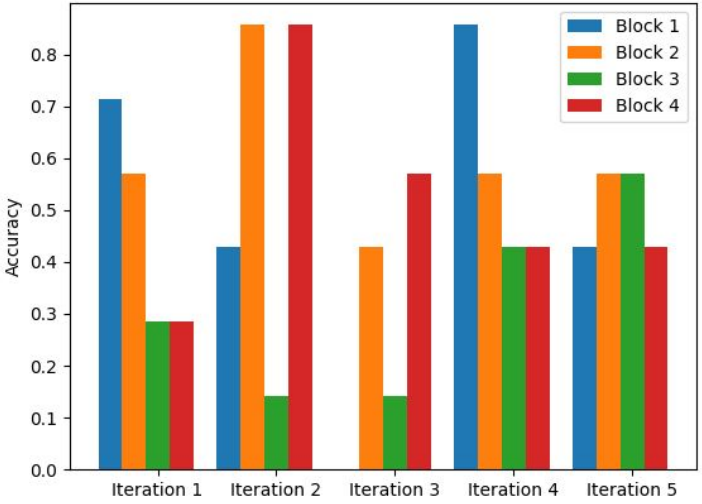
Regresión: perceptrón lineal

Como es esperable, no logra aproximar bien todos los puntos, a diferencia del no lineal que sí.





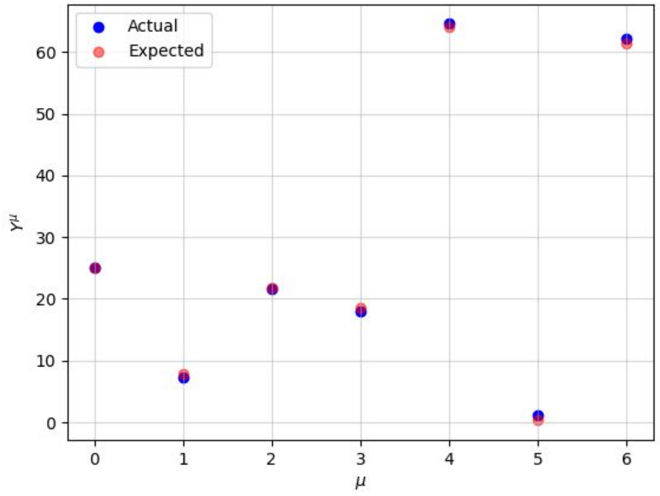
Método Cross-Validation



Regresión: capacidad de generalización

Perceptrón no lineal

- MSE Training = 0.0011
- MSE Testing = 0.3532



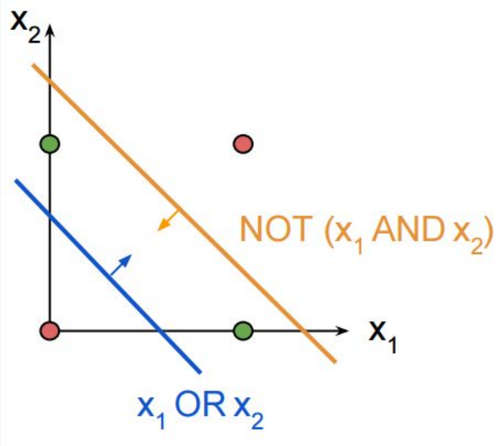
Predicciones para training set

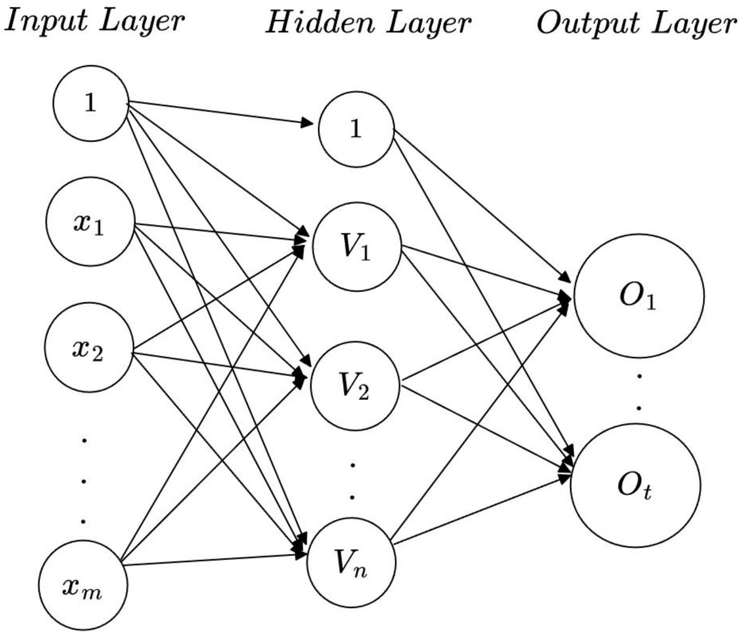
Ejercicio 3

Perceptrón Multicapa

Perceptrón Multicapa

Permite resolver problemas de **clasificación en muchos grupos**.

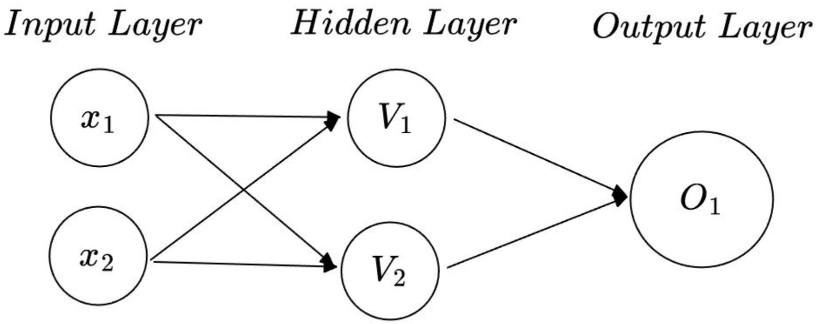




$$\theta(v) = \frac{1}{1+e^{-v}} \qquad E = MSE$$

$$convergencia \iff E \leq \frac{q}{100}(\theta_{max} - \theta_{min})$$

XOR: arquitectura

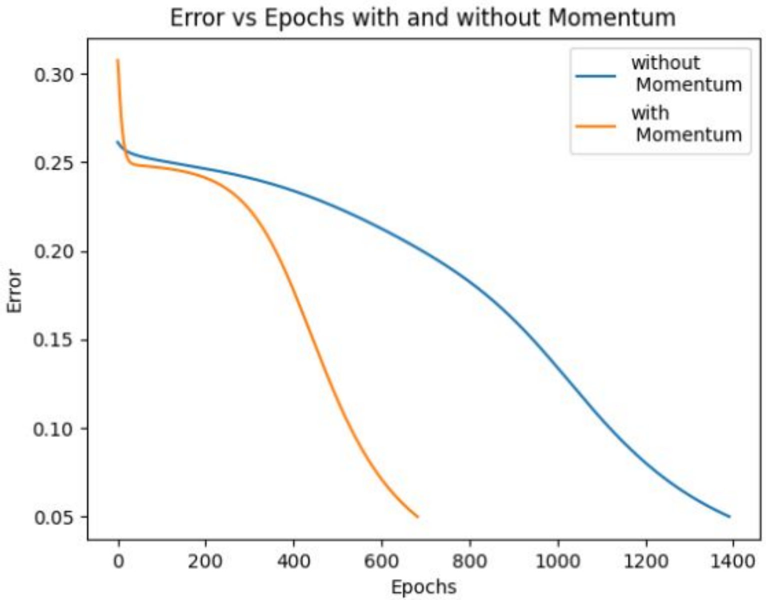


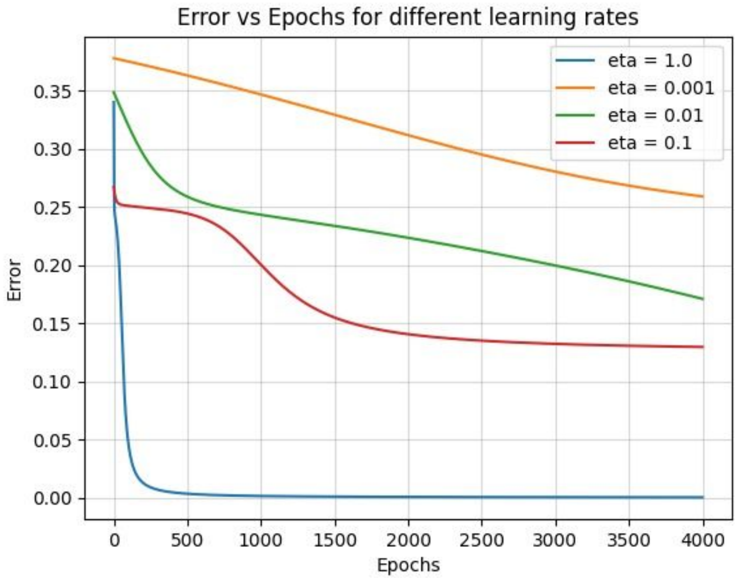
$$convergencia \iff E \leq 0.05$$

X	y
(1, -1)	1
(-1, 1)	1
(1, 1)	-1
(-1, -1)	-1

$$\eta = 0.1$$

XOR: resultados





Dígitos como inputs

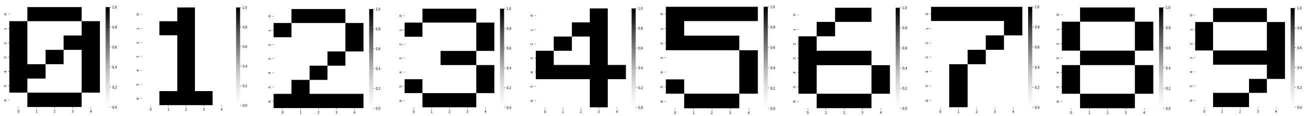
5

7

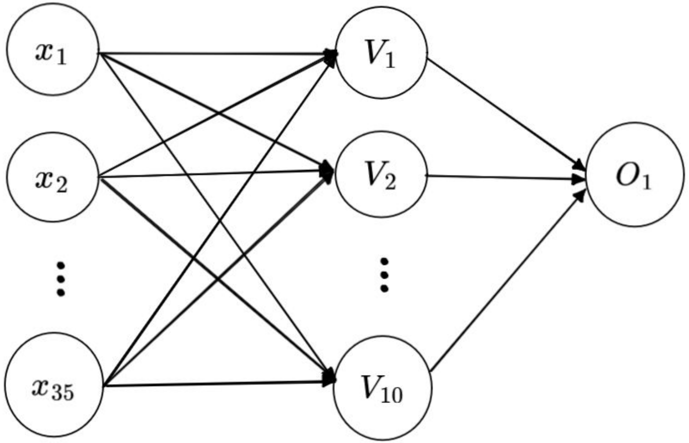
$$\begin{matrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{matrix}$$

$$array[0,1,1,1,0,1,...,1,0]$$

$$len(array) = 7x5 = 35$$



Dígito par: arquitectura

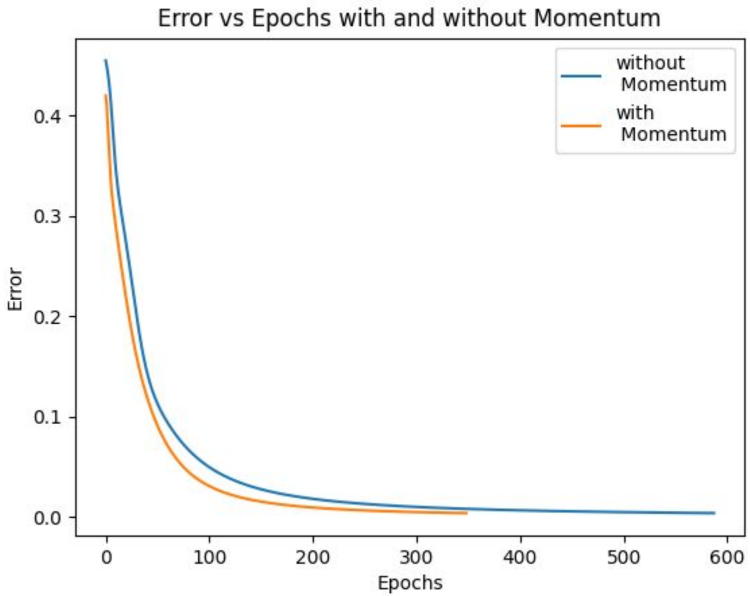


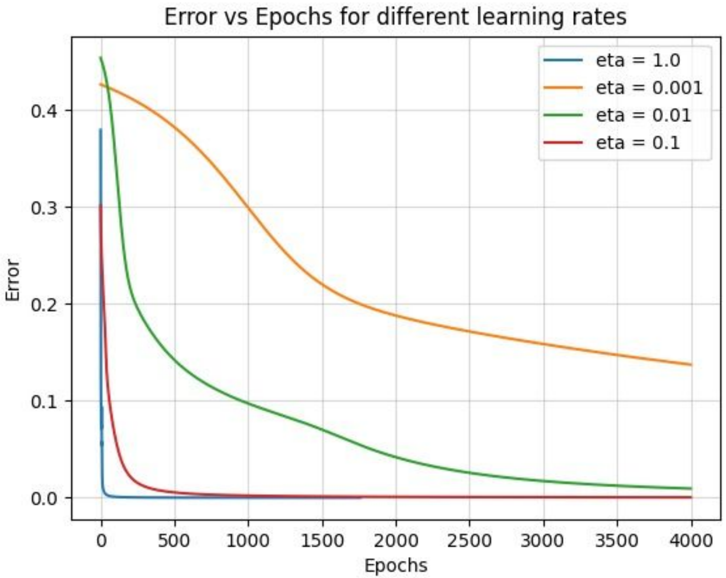
$$\eta = 0.1$$

X	y
arr[35]_0	1
arr[35]_1	-1
……	…..
arr[35]_8	1
arr[35]_9	-1

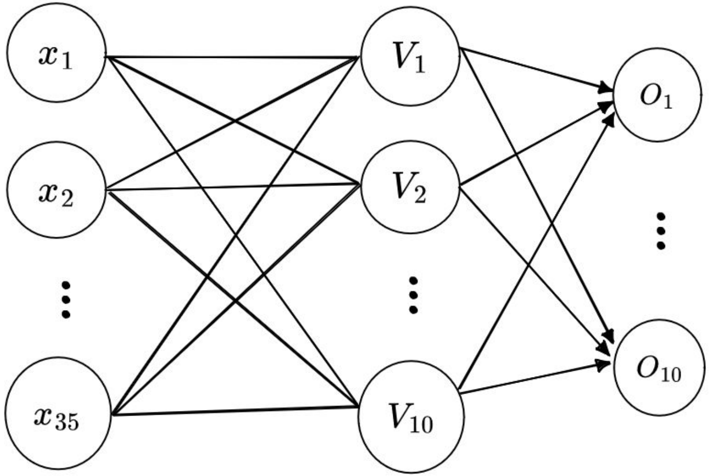
$$convergencia \iff E \leq 0.005$$

Dígito par: resultados





Predecir dígito: arquitectura

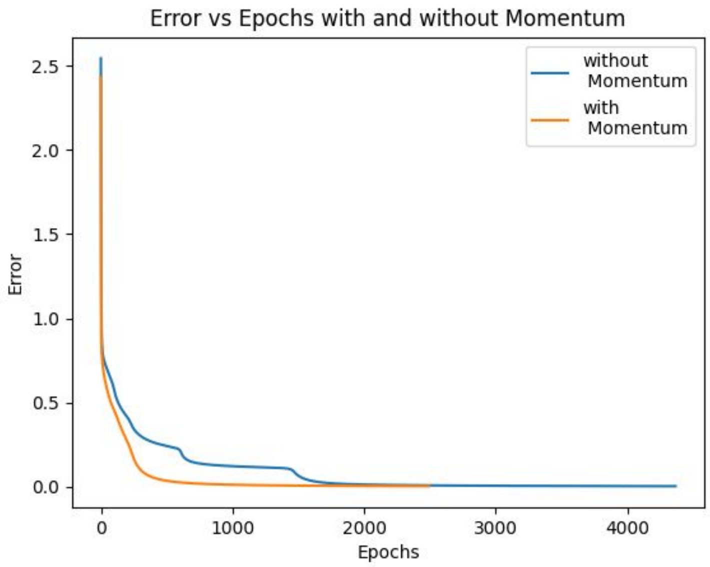


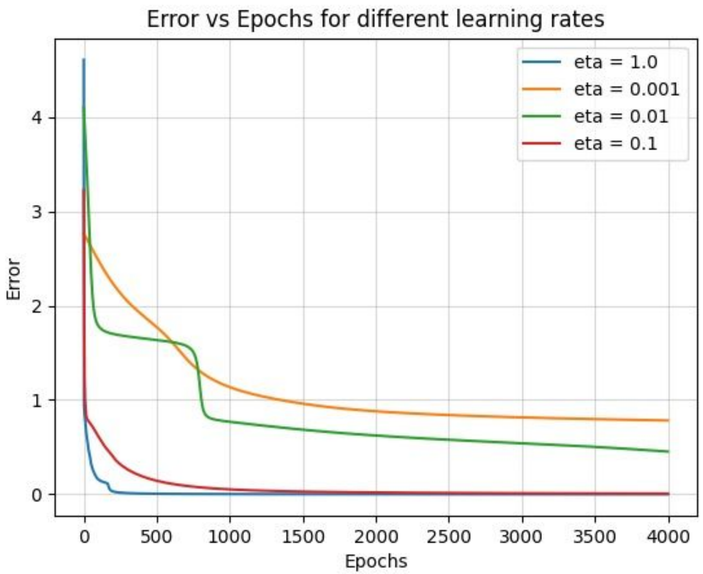
$$\eta = 0.1$$

X	y
arr[35]_0	[1, 0, …, 0, 0]
arr[35]_1	[0, 1, …, 0, 0]
……	…..
arr[35]_8	[0, 0, …, 1, 0]
arr[35]_9	[0, 0, …, 0, 1]

$$convergencia \iff E \leq 0.004$$

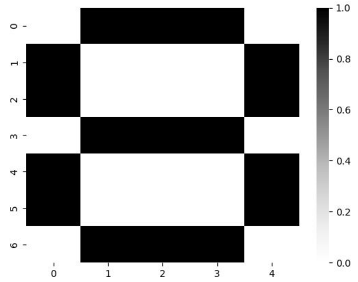
Predecir dígito: resultados

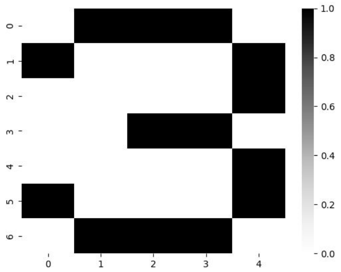


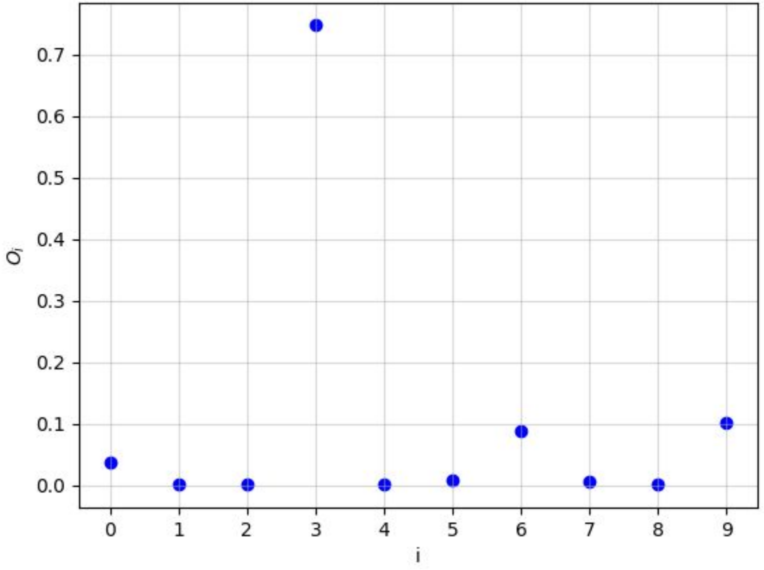


Predecir dígito: capacidad de generalización

Predicción para el 8, entrenando con todos los dígitos excepto el 8.

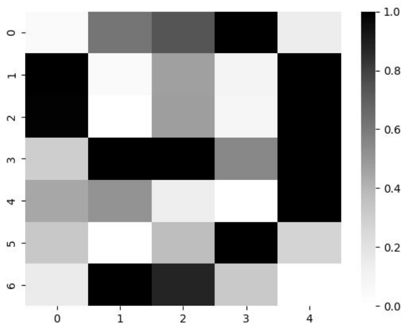




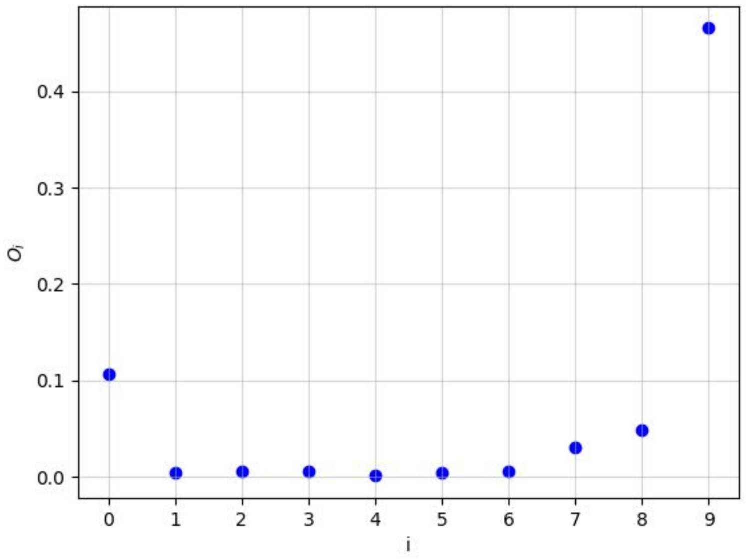


Predecir dígito: ruido

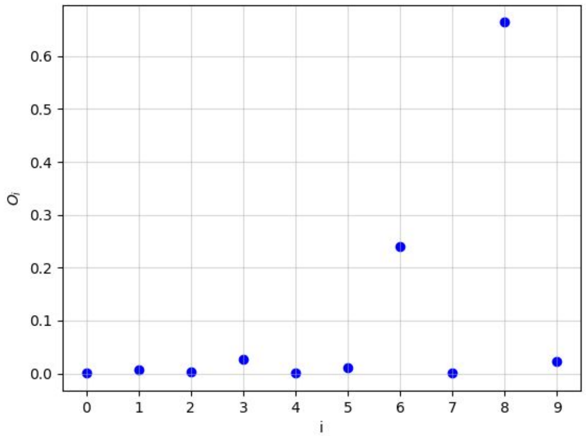
Ruido con distribución normal μ = 0, σ = 0.3

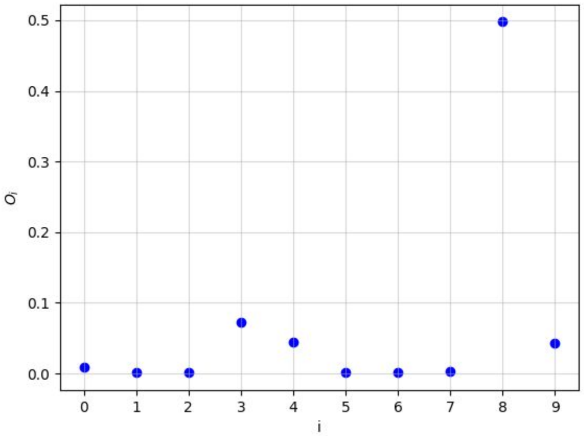


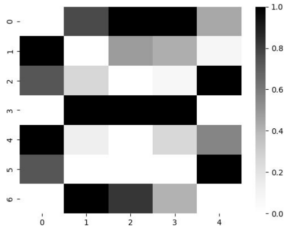
= 9



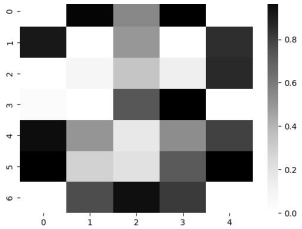
Predecir dígito: ruido







= 8



= 3

Conclusiones

Conclusiones generales

- Gradiente descendente con momentum mejora notablemente las épocas de convergencia, a cerca de la mitad con respecto a la variante sin momentum.
- El valor óptimo de la tasa de aprendizaje varía según cada problema.

Gracias