(20242Q) 72.27 - Sistemas de Inteligencia Artificial -Comisión: S

Grupo 9

Pellegrini Jorge Orlando

Perceptrón simple con función de activación escalón

X1	X2	AND(X1,X2)
-1	-1	-1
-1	1	-1
1	-1	-1
1	1	1

Tabla de verdad

$$y_{modelo} = \begin{cases} 1 & si X_1. W_1 + X_2. W_2 + W_3 \ge 0 \\ -1 & si X_1. W_1 + X_2. W_2 + W_3 < 0 \end{cases}$$

$$W_{nuevo} = W_{viejo} + \eta (y_{dato} - y_{modelo}) X_{dato}$$

Se actualiza los pesos en orden aleatorio

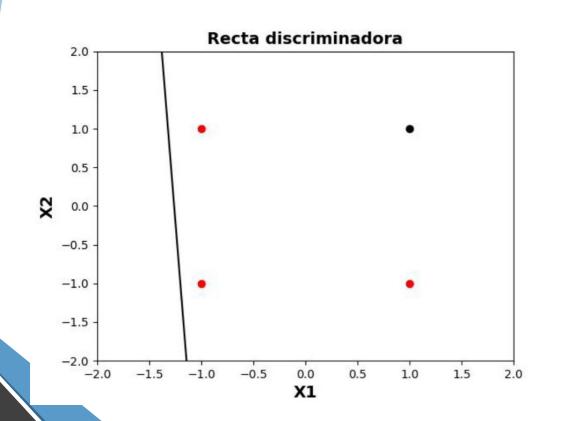
 $E_{iteración} = Puntos \ clasificados \ incorrectamente$

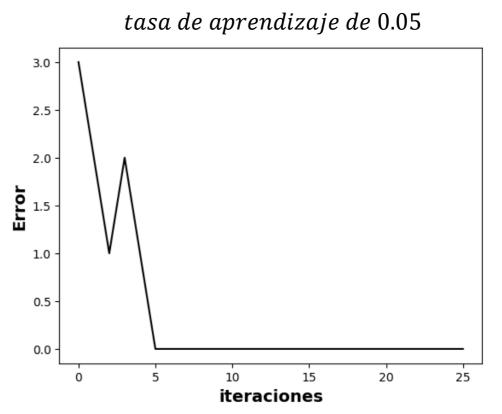
Ecuación de la frontera de decisión

$$x_1 w_1 + x_2 w_2 + 1 w_3 = 0$$

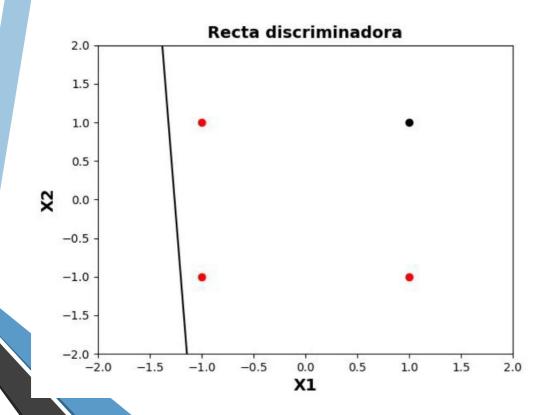
$$x_2 = -\frac{w_3}{w_2} - x_1 \frac{w_1}{w_2}$$

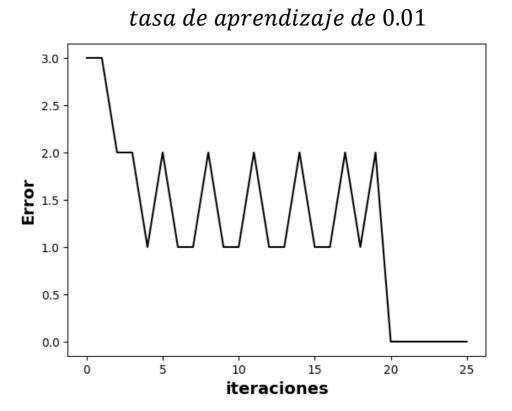
Perceptró**n simple** con función de activación **escal**ó**n**<u>Función AND</u>





Perceptró**n simple** con función de activación **escal**ó**n**<u>Función AND</u>





Perceptrón simple con función de activación escalón

X1	X2	XOR(X1,X2)
-1	-1	-1
-1	1	1
1	-1	1
1	1	-1

Tabla de verdad

$$y_{modelo} = \begin{cases} 1 & si X_1. W_1 + X_1. W_1 + W_3 \ge 0 \\ -1 & si X_1. W_1 + X_1. W_1 + W_3 < 0 \end{cases}$$

$$W_{nuevo} = W_{viejo} + \eta (y_{dato} - y_{modelo}) X_{dato}$$

Se actualiza los pesos en orden aleatorio

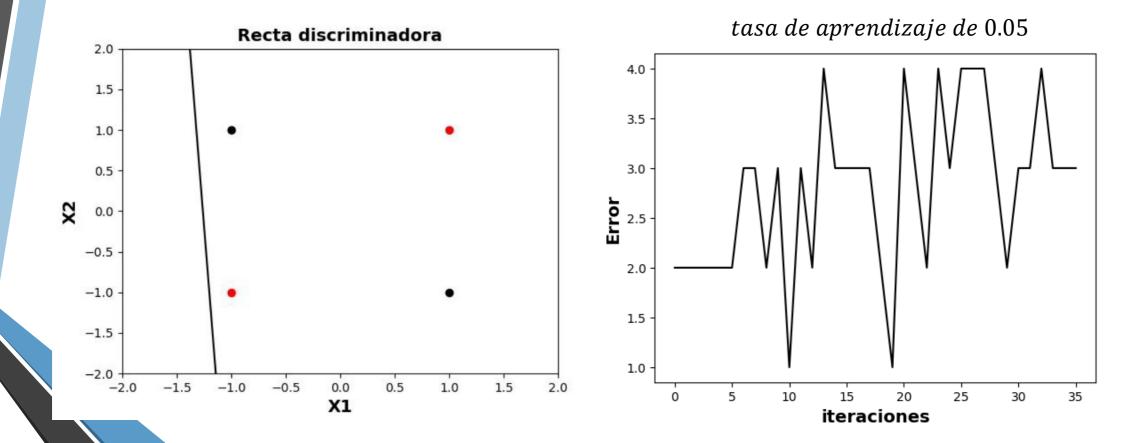
 $E_{iteración} = Puntos clasificados incorrectamente$

Ecuación de la frontera de decisión

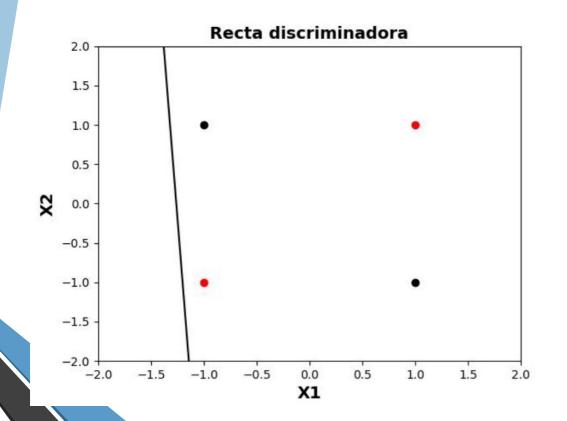
$$x_1 w_1 + x_2 w_2 + 1 w_3 = 0$$

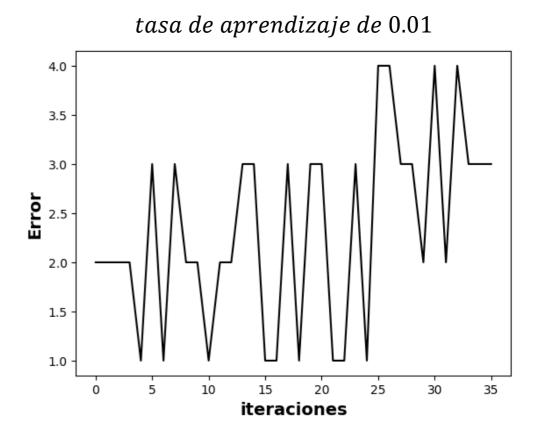
$$x_2 = -\frac{w_3}{w_2} - x_1 \frac{w_1}{w_2}$$

Perceptró**n simple** con función de activación **escal**ó**n** <u>Función XOR</u>

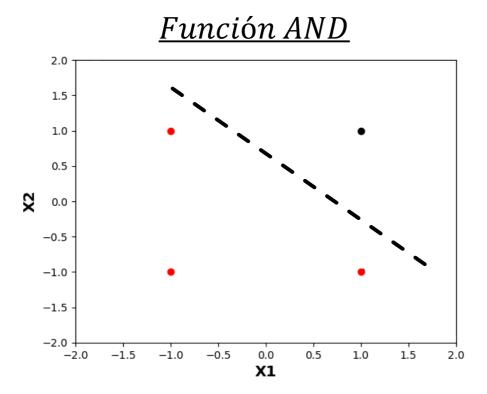


Perceptró**n simple** con función de activación **escal**ó**n** <u>Función XOR</u>





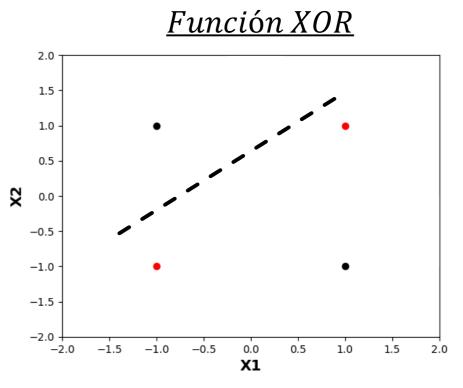
Perceptrón simple con función de activación escalón



Sistema linealmente separable

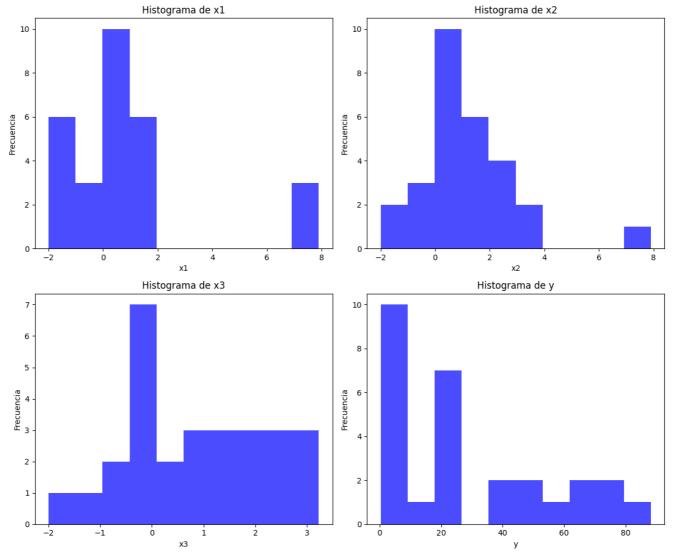


La frontera de decisión puede ser una recta



Sistema no linealmente separable

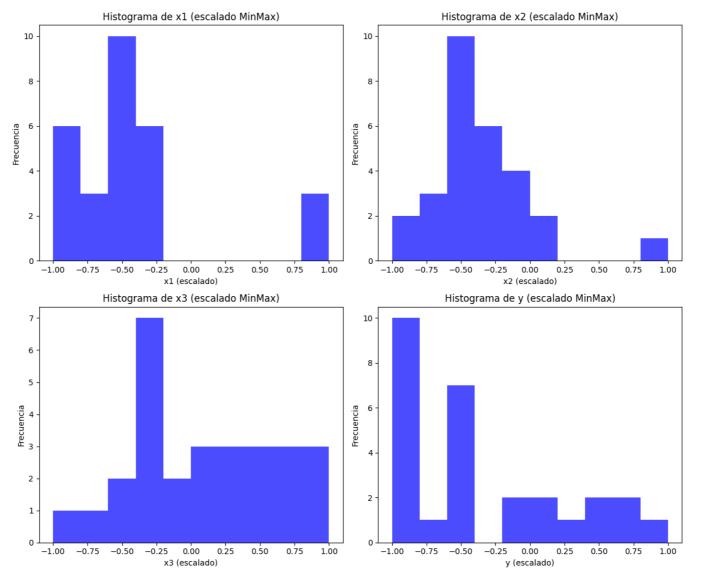
Histogramas



28 datos

$$x^* = 2\frac{x - minX}{maxX.minX} - 1$$

Escalado MinMax



28 datos

Perceptrón **simple lineal** – función de activación identidad

$$y_{modelo} = X_1.W_1 + X_2.W_2 + X_3.W_3 + W_4$$

 $W_{nuevo} = W_{viejo} + \eta (y_{dato} - y_{modelo}) X_{dato}$ Se actualiza los pesos en orden aleatorio

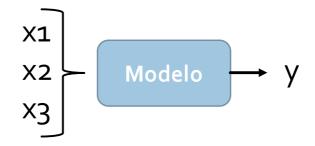
$$E_{iteración} = \frac{1}{2} \sum_{patrones} (y_{dato} - y_{modelo})^2$$

Perceptrón **simple lineal** – función de activación identidad

Capacidad de aprendizaje



Se emplea todo el set de datos

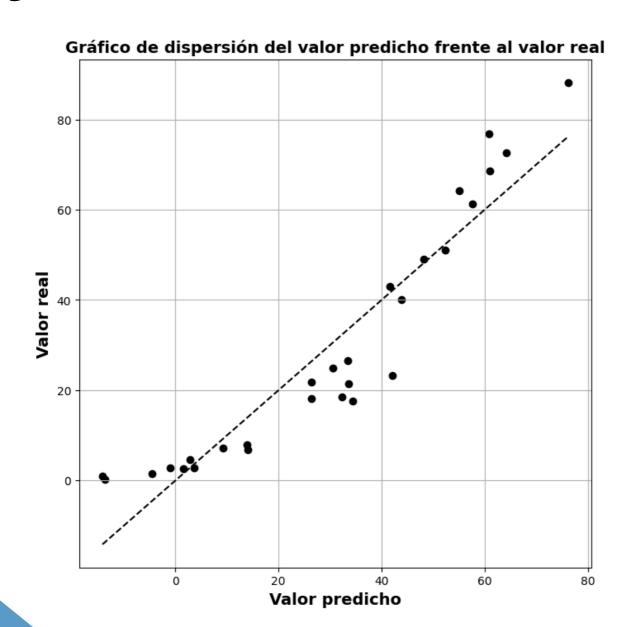


Perceptrón simple lineal

Influencia de la tasa de aprendizaje

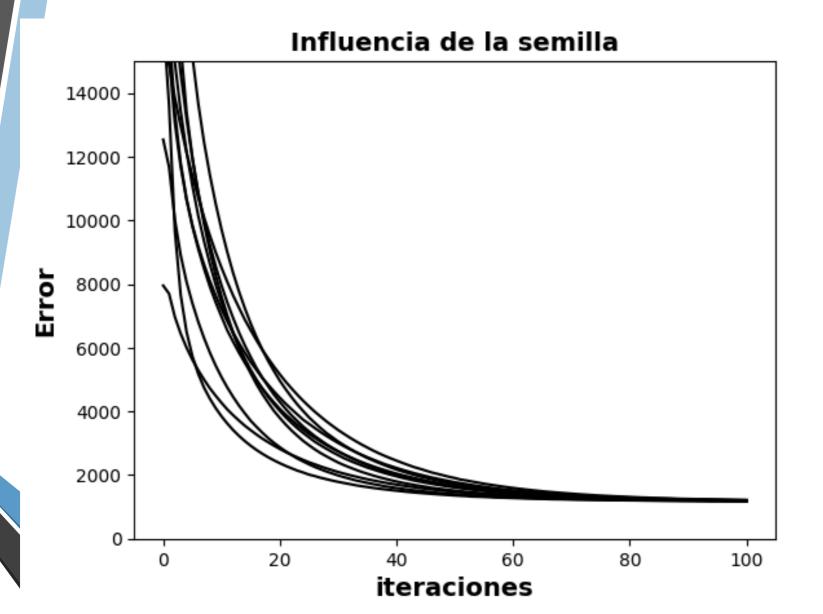
$$\eta$$
=0.01 η 00 η

Perceptrón simple lineal



$$\eta$$
=0.01

Perceptrón simple lineal



$$\eta$$
=0.01

Perceptrón simple no lineal – función de activación tanh

$$y_{modelo} = \tanh(beta * (X_1.W_1 + X_2.W_2 + X_3.W_3 + W_4))$$

$$W_{nuevo} = W_{viejo} + \eta(y_{dato} - y_{modelo}) * g * X_{dato}$$

$$g = beta * (1 - (tanh(beta * (X_1.W_1 + X_2.W_2 + X_3.W_3 + W_4)))^2)$$

Se actualiza los pesos en orden aleatorio

$$E_{iteración} = \frac{1}{2} \sum_{patrones} (y_{dato} - y_{modelo})^2$$

Perceptrón simple no lineal función de activación tanh

Influencia de la tasa de aprendizaje beta=1

$$\eta$$
=0.05 η =0.01

Perceptrón simple no lineal función de activación tanh

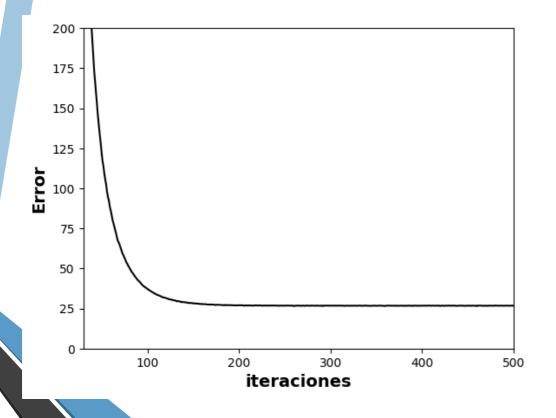
Influencia de la tasa de aprendizaje beta=1

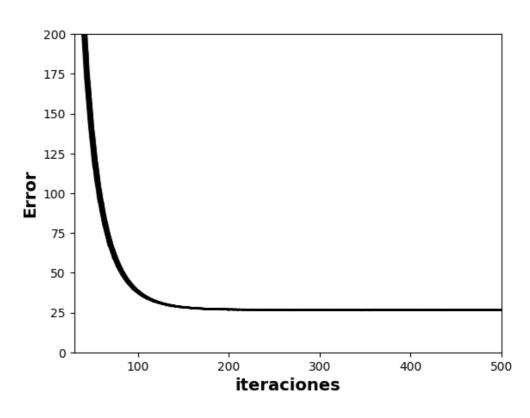
$$\eta$$
=0.5 η =0.01

Perceptrón simple no lineal función de activación tanh

Influencia de la semilla

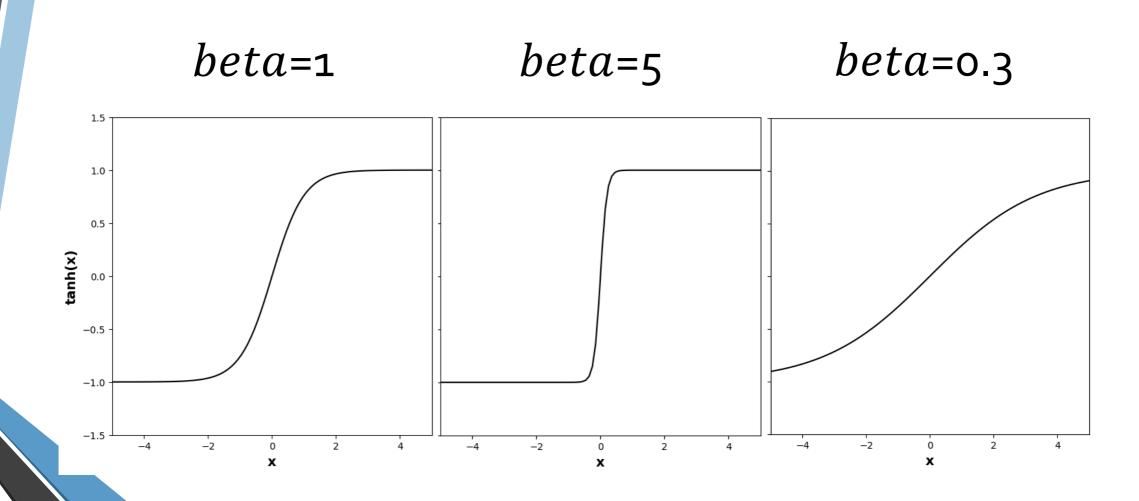
$$\eta = 0.05$$





Perceptrón simple no lineal función de activación tanh

Influencia de Beta

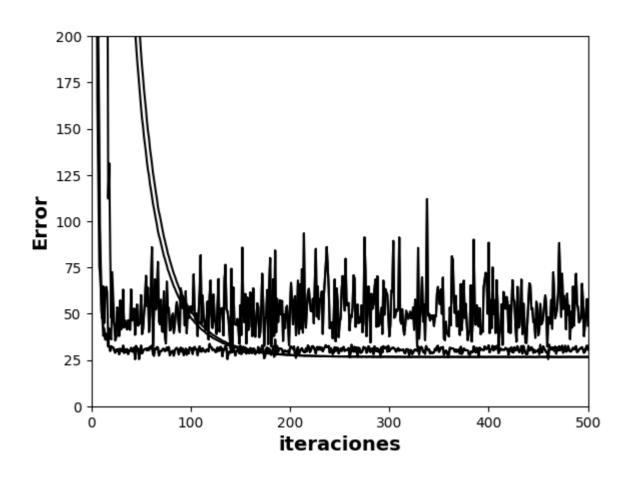


Perceptrón simple no lineal función de activación tanh

Influencia de Beta

$$Dominio_{beta} = [-8;8]$$

10 puntos



Perceptrón simple no lineal función de activación tanh



$$beta = 1$$

Perceptrón simple no lineal función de activación tanh

¿Cómo elegir el mejor conjunto de entrenamiento?

El mejor conjunto de entrenamiento es el que no se elije

¿Qué efecto tiene dicha elección en la capacidad de generalización?

No debe tener efecto



Si lo tiene, puede ser un indicio de que el set de datos no contiene la información para predecir el fenómeno

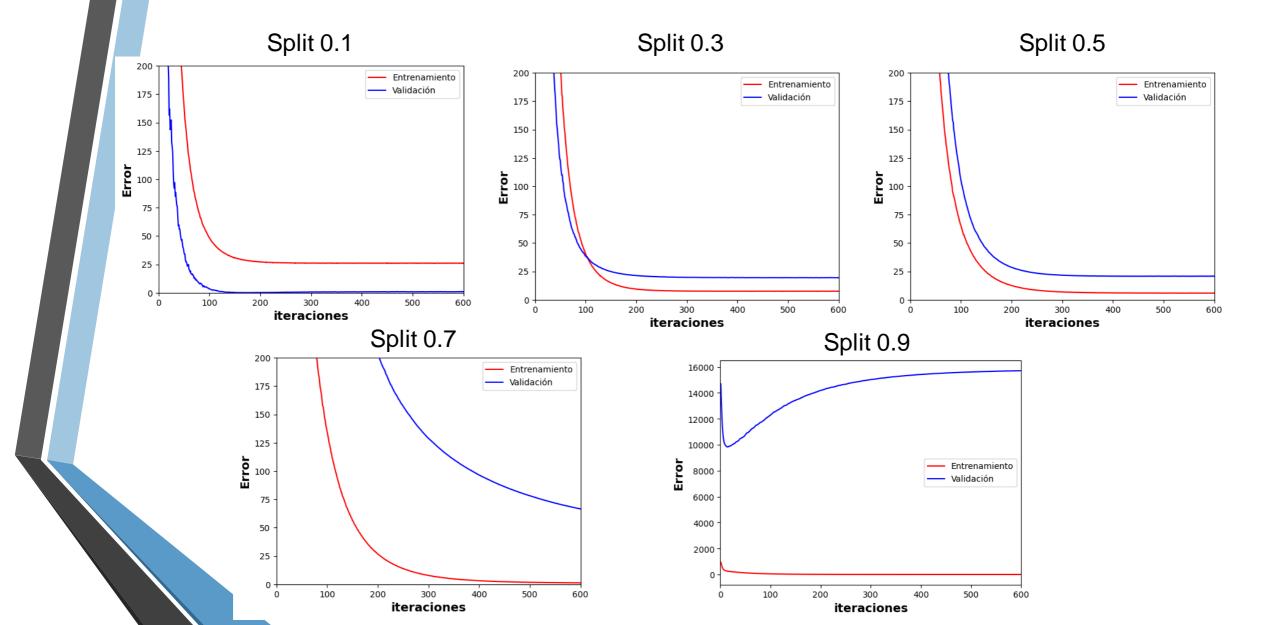


Por falta de datos y/o por la naturaleza de los mismos

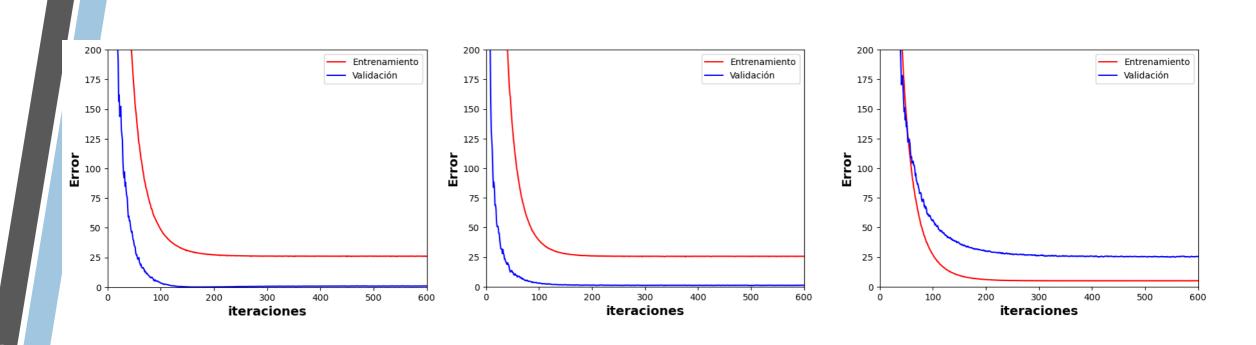
Perceptrón simple no lineal función de activación tanh

Capacidad generalizadora

Split: validación/entrenamiento



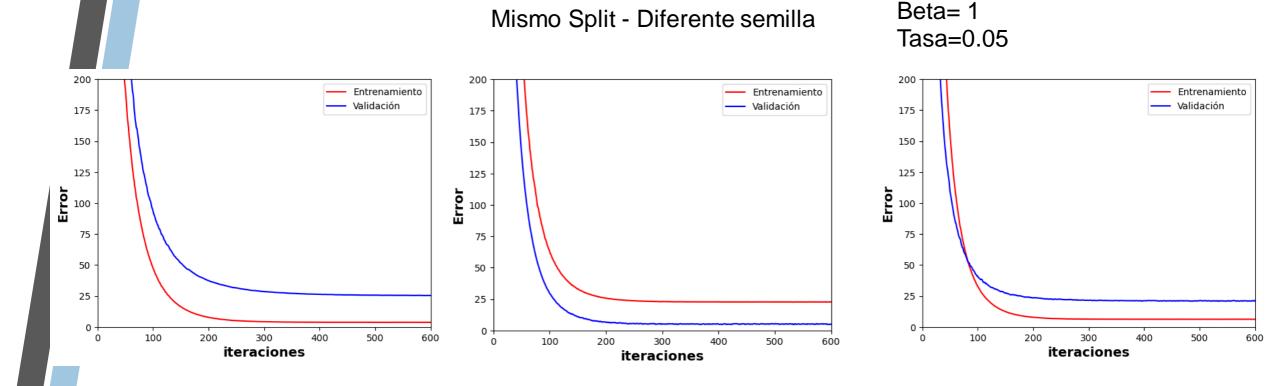
Perceptrón simple no lineal función de activación tanh Capacidad generalizadora



Mismo Split - Diferente semilla

Split 0.1

Perceptrón simple no lineal función de activación tanh Capacidad generalizadora



Split 0.3

No se puede minimizar la validación ya que depende del Split

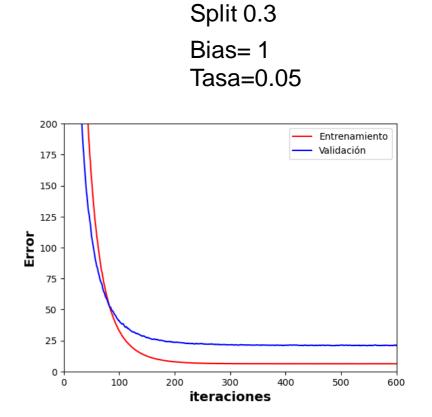


Set con pocos datos

Perceptrón simple no lineal función de activación tanh

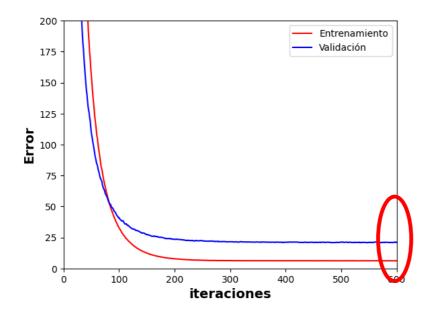
Capacidad generalizadora



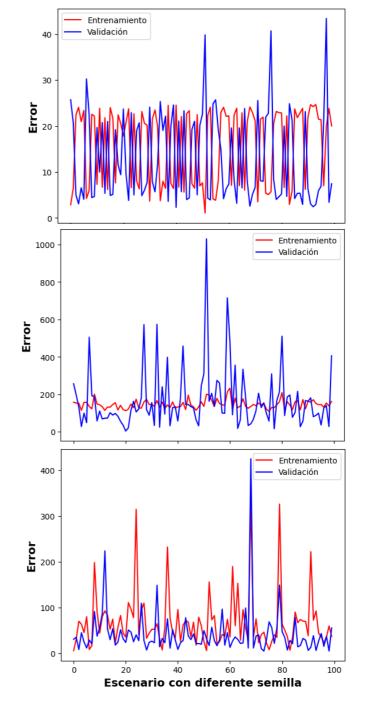


Se evalúa si existe un beta que atenúe el efecto para un numero de semillas con un Split fijo

Perceptrón simple no lineal Capacidad generalizadora



Se evalúa los extremos para diferentes semillas

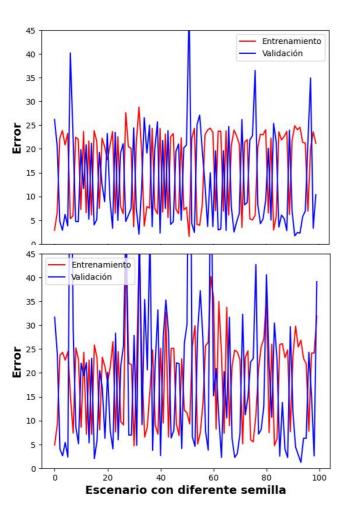


Split 0.3 Beta= 1 Tasa=0.05

Split 0.3 Beta= 0.3 Tasa=0.05

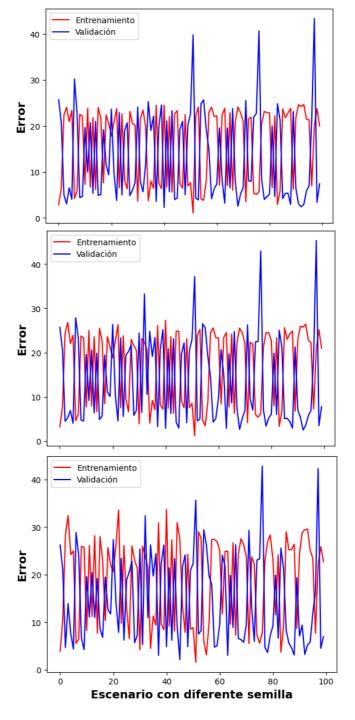
Split 0.3 Beta= 5 Tasa=0.05

Perceptrón simple no lineal Capacidad generalizadora



Split 0.3 Beta= 0.8 Tasa=0.05

Split 0.3 Beta= 0.6 Tasa=0.05

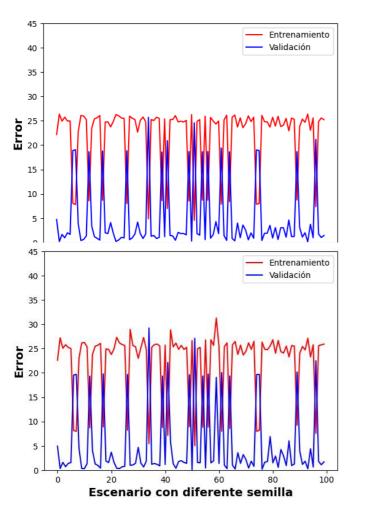


Split 0.3 Beta= 1 Tasa=0.05

Split 0.3 Beta= 2 Tasa=0.05

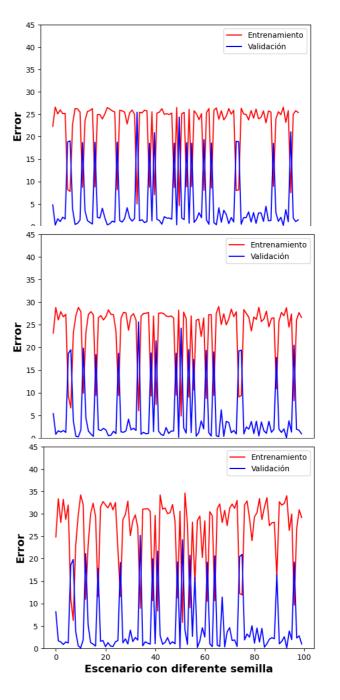
Split 0.3 Beta= 3 Tasa=0.05

Perceptrón simple no lineal Capacidad generalizadora



Split 0.1 Beta= 0.8 Tasa=0.05

Split 0.1 Beta= 0.6 Tasa=0.05



Split 0.1 Beta= 1 Tasa=0.05

Split 0.1 Beta= 2 Tasa=0.05

Split 0.1 Beta= 3 Tasa=0.05

Perceptrón simple no lineal

Capacidad generalizadora

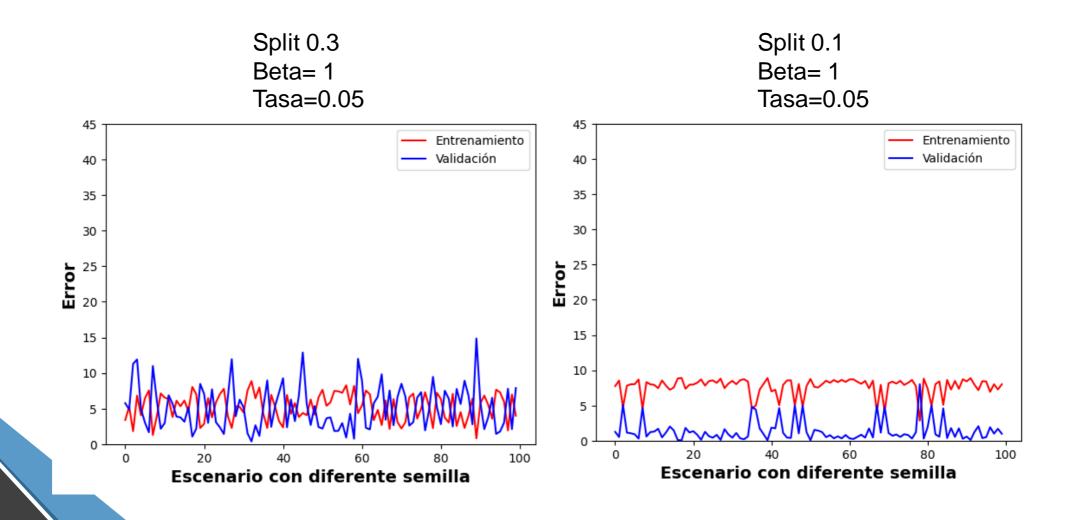
Si se baja el Split, el conjunto de validación no representa el comportamiento de nuevos datos

Se puede plantear que el punto extremo no representa el comportamiento a predecir del set de datos y por lo tanto se lo descarta.

Perceptrón simple no lineal

Capacidad generalizadora

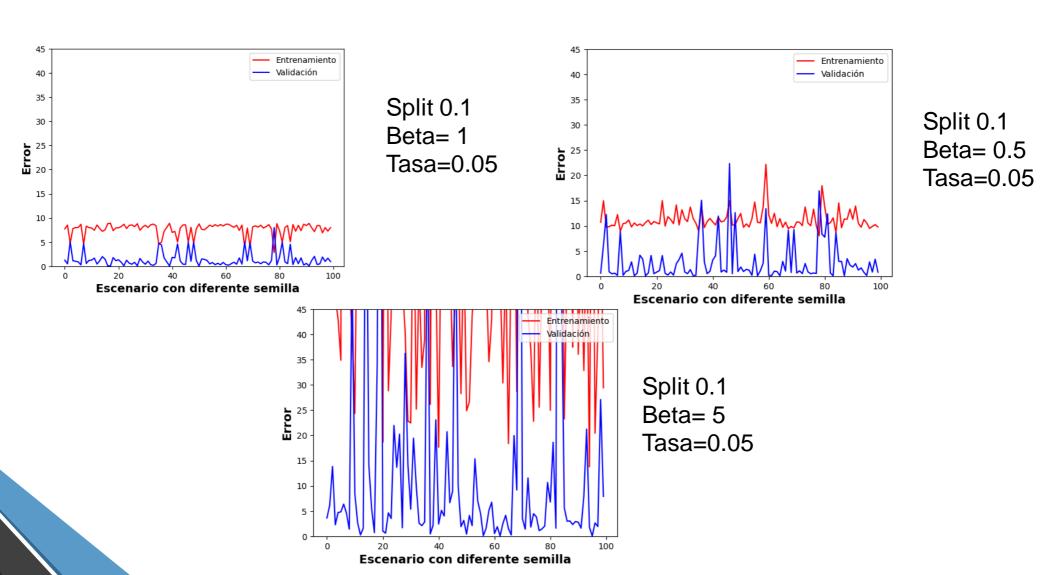
Se puede plantear que el punto extremo no representa el comportamiento a predecir del set de datos y por lo tanto se lo descarta.



Perceptrón simple no lineal

Capacidad generalizadora

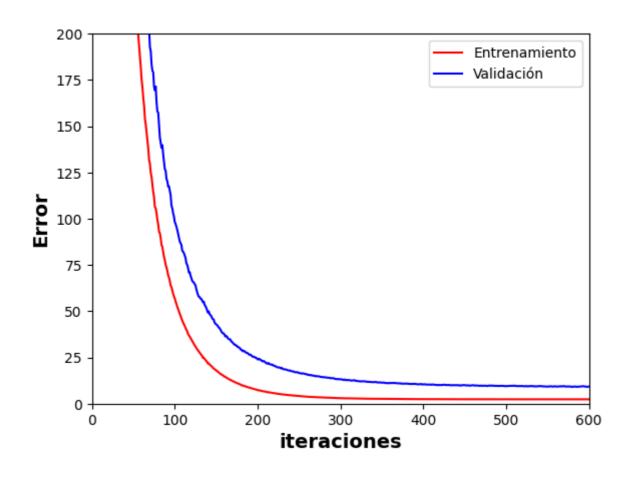
Se elije un Split de 0.1 por ser reproducible



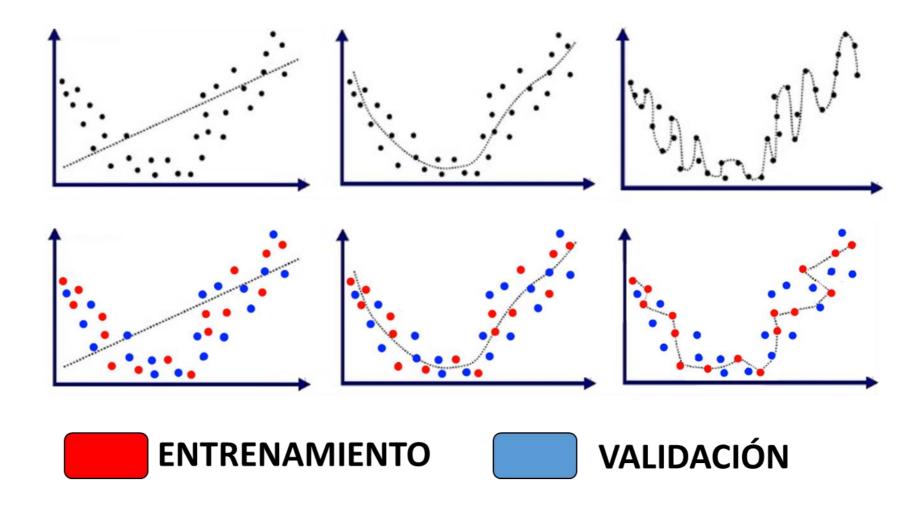
Perceptrón simple no lineal

Capacidad generalizadora

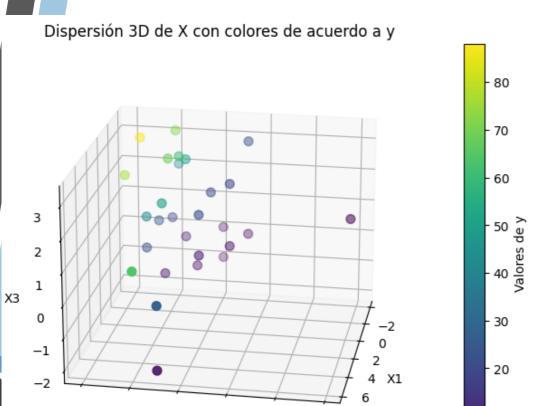
Se elije un Split de 0.1 por ser reproducible, Beta=1, Tasa=0.05



Perceptrón simple no lineal
Capacidad generalizadora
Se esperaba:



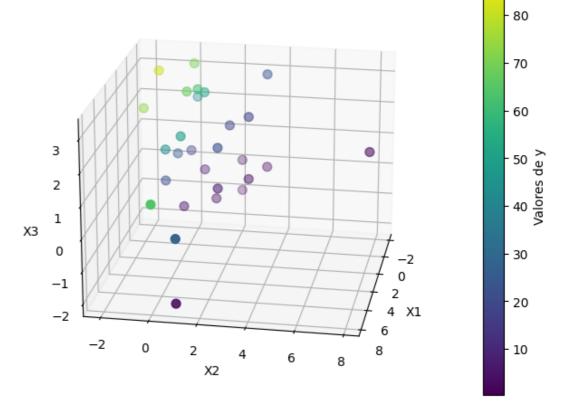
Perceptrón simple no lineal Capacidad generalizadora Realidad:



X2

- 10

Dispersión 3D de X con colores de acuerdo al modelo



Perceptrón multicapa

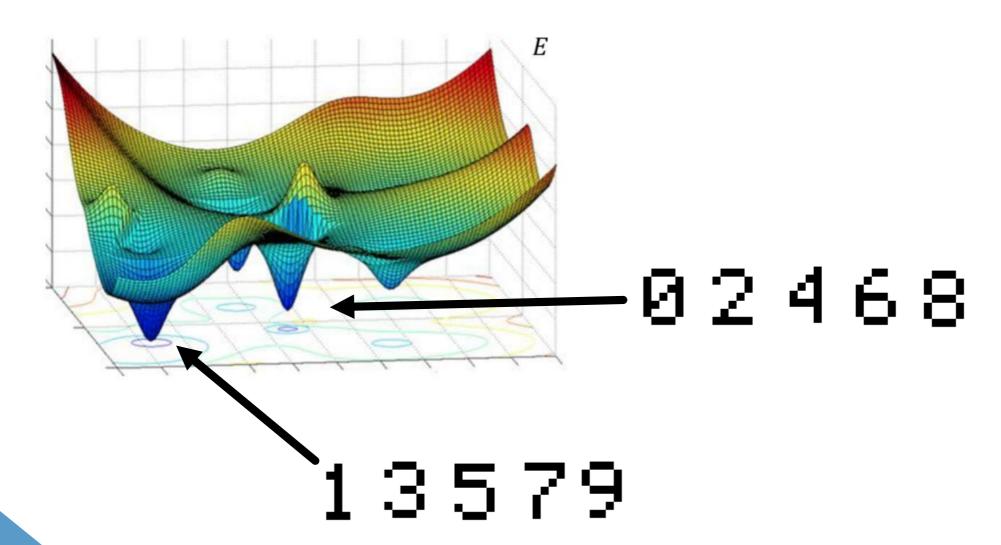
Ø 1 2 3 4Dígitos5 6 7 8 9

Números pares: 02468 Etiqueta 1

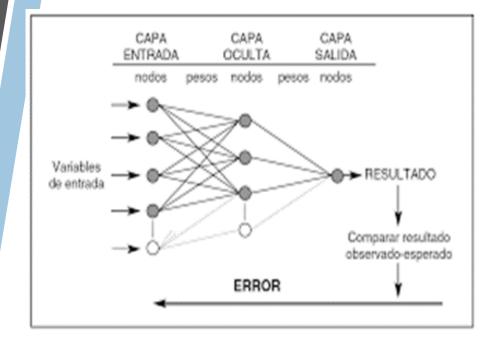
Números impares: 13579 Etiqueta 0

Perceptrón multicapa

Planteo del problema



Esquema



Perceptrón multicapa

- •1) 35 nodos de entrada, n neuronas en capa oculta y una salida.
- •2) Se inicia el ciclo de cálculo. Presentando el patrón
- •3) Se calcula la salida del sistema (*feedforward*) (Y_{Output}). Utilizando la función logsig como función de transferencia.

$$f(x) = \frac{1}{1 + e^{-x}}$$

Backpropagation

•4) Se computa el error $e(y_d - y_{output})$) y f' para la salida de la red.

$$f(x) = \frac{1}{1 + e^{-x}} \Rightarrow f'(x) = f(x)(1 - f(x))$$

•5) Se calcula la señal delta para la salida:

$$\delta = e_{output} y_{output} (1 - y_{output}) = (y_d - y_{output}) y_{output} (1 - y_{output})$$

•6) Actualización de los pesos sinápticos de las conexiones de las neuronas "k" de la capa oculta a la salida.

$$w_{k(i+1)} = w_{k(i)} + \Delta w_{k(i)} = w_{k(i)} + n\delta y_{k-Salida}$$

•7) Actualización de los pesos sinápticos de las conexiones "Wek" desde las entradas (e) a las neuronas k de la capa oculta.

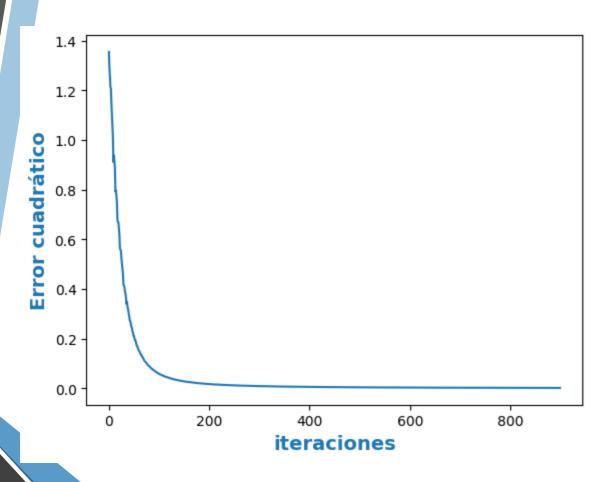
$$w_{ek(i+1)} = w_{ek(i)} + \Delta w_{ek(i)} = w_{ek(i)} + n \left[\sum_{k} \delta w_{kj(i)} \right] y_{k-Salida} (1 - y_{k-Salida}) x_e$$

- •Se repiten los pasos 2 a 7 para todos los patrones a aprender.
- •Se efectúa el cálculo del Error Cuadrático a todo el set de patrones "e" como:

$$E = \frac{1}{2} \sum_{e} (y_d - y_{k(e)})^2$$

Perceptrón multicapa

Resultado



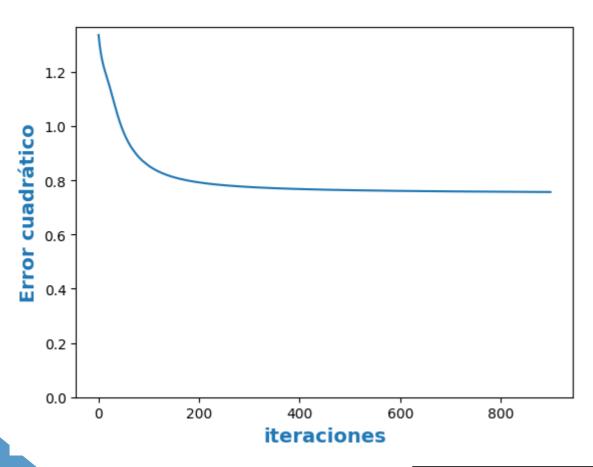
Iteraciones=900 Tiempo=3s +/- 0.5 Neuronas capa oculta= 20

Salida:

Tasa = 0.25

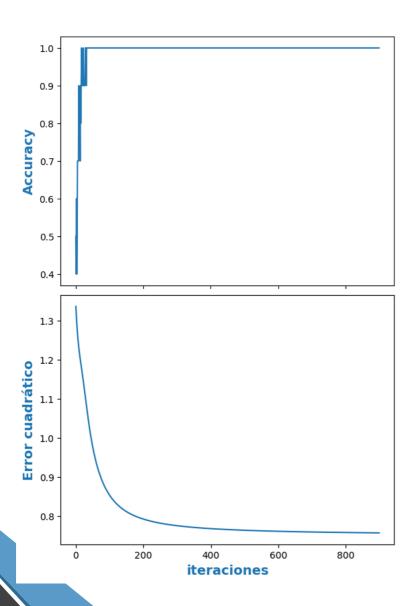
Perceptrón multicapa

Resultado



Iteraciones=900
Tiempo=3s +/- 0.5
Neuronas capa
oculta= 1
Tasa= 0.25

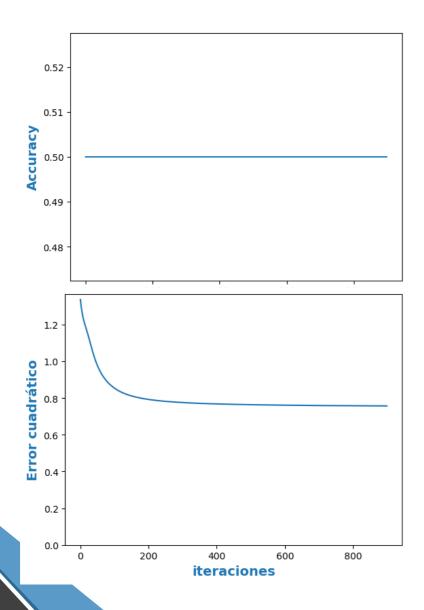
Salida:



$$Accuracy = \frac{Predicciones\ correctas}{total\ de\ predicciones}$$

Iteraciones=900
Tiempo=3s +/- 0.5
Neuronas capa
oculta= 20

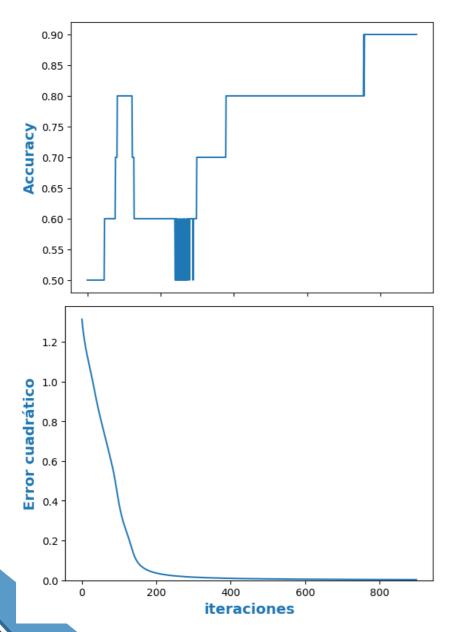
Tasa= 0.25



$$Accuracy = \frac{Predicciones\ correctas}{total\ de\ predicciones}$$

Iteraciones=900
Tiempo=3s +/- 0.5
Neuronas capa
oculta= 1

Tasa = 0.25



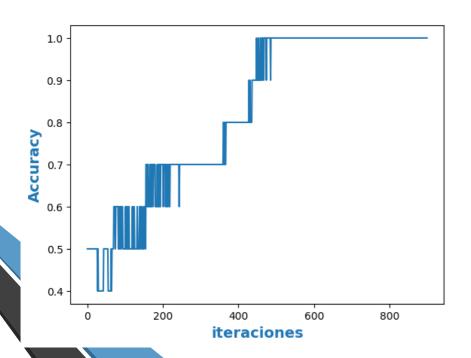
$$Accuracy = \frac{Predicciones\ correctas}{total\ de\ predicciones}$$

Iteraciones=900
Tiempo=3s +/- 0.5
Neuronas capa
oculta= 2

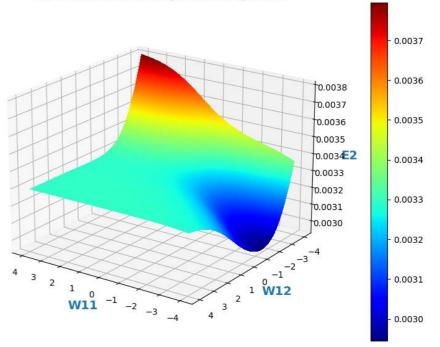
Tasa = 0.25

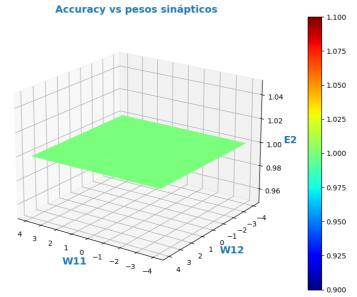
Iteraciones=900 Tiempo=2s +/- 0.5 Neuronas capa oculta= 8

Tasa= 0.25

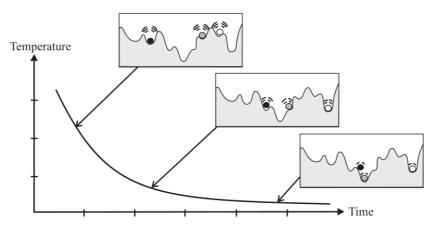








Simulated annealing



Fuente:Ledesma, S., Ruiz, J., & Garcia, G. (2012). Simulated annealing evolution. Simulated Annealing-Advances, Applications and Hybridizations, 210-218.

$$P = p_0 e^{rac{-\Delta E}{T}}$$

 ΔE : Variación de la función objetivo

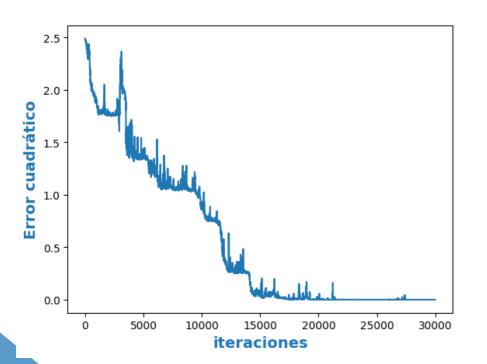
T : *f*(iteraciones)

Se toma un movimiento al azar, se acepta si $\Delta E < 0$ o con una probabilidad P

$$T = \frac{Ti}{iteraciones}$$

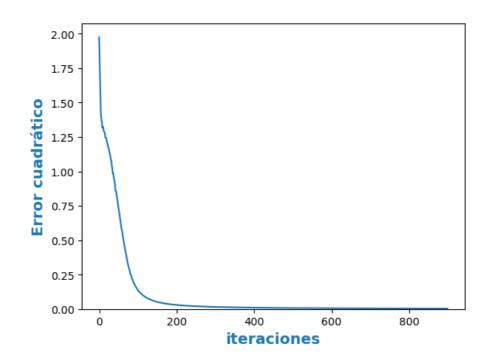
Simulated annealing

Iteraciones=30000
T inicial=1000
P inicial=1
Tiempo=20s +/- 2
Neuronas capa oculta= 8

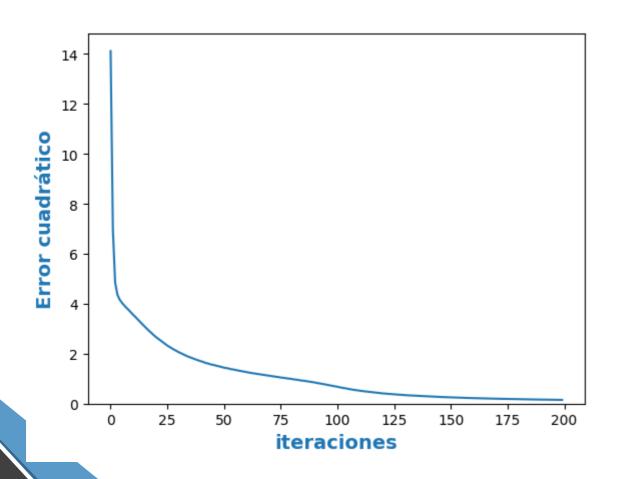


Gradiente descendiente

Iteraciones=900 Tiempo=1s +/- 0.3 Neuronas capa oculta= 8



Ejercicio 3 Discriminador de dígito



Se modifica y

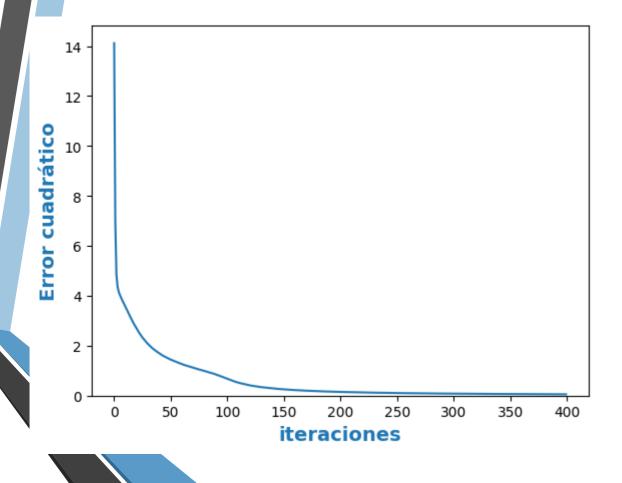
```
[[1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 1. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0.]
```

10 neuronas de salida

20 neuronas de la capa oculta Tiempo= 1s +/- 0.5

Ejercicio 3 Discriminador de dígito

10 neuronas de salida20 neuronas de la capa oculta



```
9.17477936e-01 2.20066623e-03 1.06599463e-02 3.43363648e-02
2.06254397e-02 4.27912856e-02 3.36821661e-02 3.46547775e-02
2.65925641e-02 1.53420008e-02]
[6.05577325e-03 9.36861574e-01 2.64437029e-02 1.15938891e-02
4.37481132e-02 4.33869557e-03 2.76739590e-02 1.17346824e-02
5.63263134e-04 2.65362999e-02]
[1.50078104e-02 3.23946671e-02 9.38615181e-01 2.24838323e-02
7.32586474e-03 1.40610553e-02 2.33345012e-03 3.18617255e-02
3.81530763e-02 2.78690483e-02]
[4.35498673e-02 1.69345114e-02 1.98849743e-02 9.24114247e-01
3.79117887e-03 1.81190690e-02 2.56939026e-02 1.47484156e-02
5.17100427e-02 3.30822487e-02]
[2.89293864e-02 2.95122664e-02 1.89894784e-02 1.46935014e-03
9.38750722e-01 2.68633943e-02 2.09816440e-02 2.42651997e-02
2.98421687e-03 7.92133998e-03]
[3.12885023e-02 1.09966818e-02 1.63775854e-02 3.32292100e-02
2.98647606e-02 9.40434174e-01 1.62326546e-02 3.53379566e-02
3.43494476e-02 3.15057582e-02]
[3.24025317e-02 3.03410923e-02 1.93239523e-03 2.80788645e-02
1.78283581e-02 2.71294948e-03 9.31652015e-01 1.84723375e-02
5.71097625e-02 8.10486064e-03]
[3.75081280e-02 2.64901433e-02 2.82724245e-02 2.91086780e-02
3.15484150e-02 2.07012131e-02 2.12029427e-02 9.33027398e-01
1.19699741e-02 4.68481506e-03]
[2.64998201e-02 1.16572794e-03 2.93315915e-02 3.46151520e-02
3.44447367e-03 9.62397696e-03 3.73497413e-02 3.96974616e-03
9.01460934e-01 2.78444413e-02]
[2.24075059e-02 2.49104425e-02 3.01982868e-02 2.48972888e-02
8.32247281e-03 1.28450276e-02 8.40298206e-03 1.28935002e-03
4.10015927e-02 9.36105998e-01]]
```

Ejercicio 3 Discriminador de dígito

14 12 uadrátic 10

100

150

200

iteraciones

250

300

350

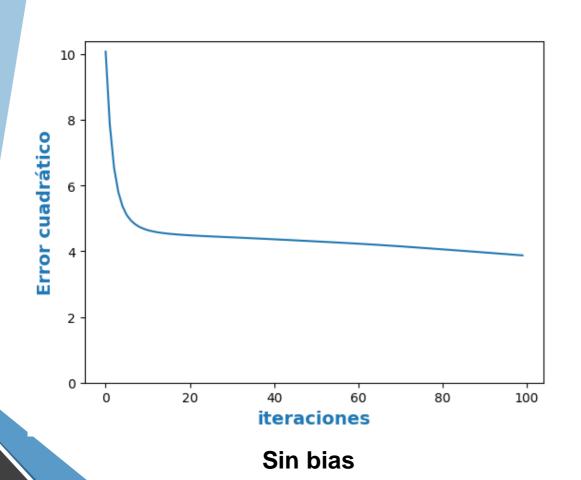
400

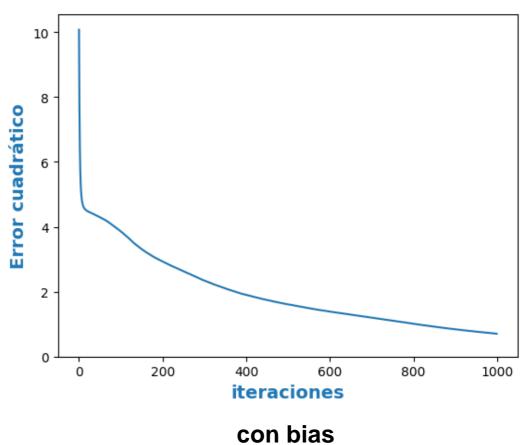
20 neuronas de la capa oculta Se añade bias

```
9.17580798e-01 2.21934388e-03 1.06952147e-02 3.45301910e-02
 2.03858968e-02 4.27528438e-02 3.35060562e-02 3.44872774e-02
2.66757272e-02 1.55173759e-02]
[6.10723163e-03 9.36917060e-01 2.63272786e-02 1.16619762e-02
4.37208495e-02 4.33360642e-03 2.77162132e-02 1.15293886e-02
5.58066069e-04 2.66190894e-02]
[1.51373807e-02 3.24037098e-02 9.38569147e-01 2.26039622e-02
7.32208958e-03 1.41088183e-02 2.31919295e-03 3.17431679e-02
3.80788933e-02 2.77901300e-02]
[4.35102238e-02 1.70444681e-02 1.99220347e-02 9.24006553e-01
3.79026688e-03 1.82181191e-02 2.59034160e-02 1.45907352e-02
5.17849449e-02 3.31540981e-02]
[2.86610773e-02 2.96575453e-02 1.88245687e-02 1.47463335e-03
9.38722645e-01 2.69653642e-02 2.12074821e-02 2.43119372e-02
2.90931433e-03 7.93388645e-03]
[3.11056183e-02 1.09391363e-02 1.62849033e-02 3.31264785e-02
2.96858839e-02 9.40668966e-01 1.60704494e-02 3.53119180e-02
3.42718453e-02 3.14571486e-02]
[3.21898037e-02 3.01353025e-02 1.91047087e-03 2.81177372e-02
1.77739256e-02 2.68606358e-03 9.31777016e-01 1.83735539e-02
5.68901417e-02 8.02553362e-03]
[3.73535941e-02 2.62010942e-02 2.81258911e-02 2.89959044e-02
 3.17550682e-02 2.08872725e-02 2.15043941e-02 9.33047175e-01
1.17464066e-02 4.63842316e-03]
[2.62796699e-02 1.16989253e-03 2.92573383e-02 3.45858550e-02
 3.39009712e-03 9.59444350e-03 3.74000127e-02 3.92968838e-03
 9.01469302e-01 2.79372900e-02]
[2.21306681e-02 2.47978402e-02 3.00872532e-02 2.47175750e-02
 8.07774780e-03 1.26328654e-02 8.29343631e-03 1.26942046e-03
 4.07381490e-02 9.35973657e-01]]
```

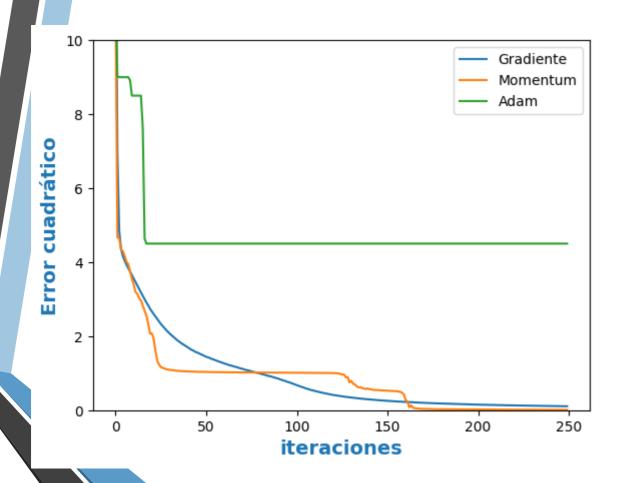
Ejercicio 3 Influencia del bias

4 neuronas de la capa oculta





Ejercicio 3 Influencia de los gradientes



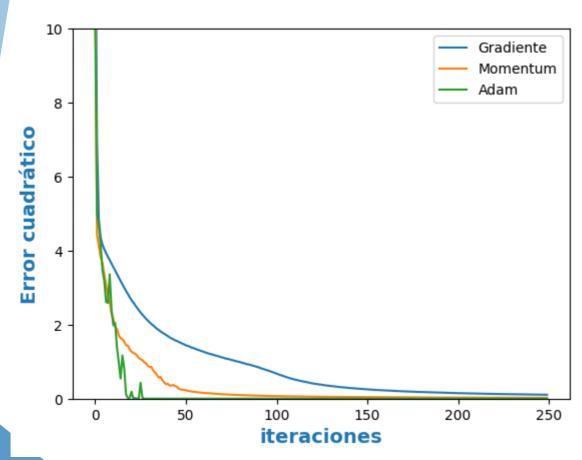
4 neuronas de la capa oculta

Momentum- factor de 0.9

Adam- Beta1=0.9; Beta2=0.9

Tasa de 0.25

Ejercicio 3 Influencia de los gradientes



4 neuronas de la capa oculta

Momentum- factor de 0.7

Adam- Beta1=0.6; Beta2=0.8

Tasa de 0.25

Influencia de cambiar un pixel del primer patrón

[0 0 0 0 0 0 0 0 0]	1 cambio	1 accuracy
[0 0 0 0 0 0 0 8 0]	5 cambios	0.9 accuracy
[2 0 0 7 9 0 0 1 0 9]	10 cambios	0.5 accuracy
[5 1 9 4 2 1 9 9 2 9]	15 cambios	0 accuracy
[4 4 4 4 4 4 4 4 4]	35 cambios	0 accuracy

Influencia de cambiar un pixel del primer patrón

[4 2 1 4 2 2 4 4 2 2]

0 accuracy

30 cambios

```
([[4.34196440e-04, 2.33476182e-01, 1.49261099e-01, 3.31897716e-03,
  2.64094185e-01, 3.26800442e-02, 3.40149876e-03, 6.95191869e-03,
  1.71670242e-02, 7.61939446e-02],
 [4.00968311e-03, 2.59083376e-01, 6.19570791e-01, 8.02666525e-04,
  1.92364317e-01, 2.25431919e-02, 1.80156934e-03, 5.02946399e-03,
  4.62284298e-02, 5.81816949e-01],
 [4.67720563e-03, 3.96410660e-01, 6.79603884e-02, 4.01721350e-04,
  3.09576616e-01, 4.47732476e-02, 2.04235657e-02, 8.88095301e-02,
  7.09551456e-03, 3.41820418e-01],
 [1.60829481e-03, 3.24358844e-01, 1.34536176e-01, 2.92954821e-03,
  5.44105490e-01, 5.25740553e-02, 3.58032088e-03, 7.34155752e-03,
  3.67020585e-03, 3.90900228e-01],
 [8.18064498e-03, 4.46567151e-02, 5.98553745e-01, 5.76951157e-04,
  1.68734715e-01, 3.48669900e-01, 1.93225339e-04, 1.70884968e-02,
  6.46797220e-02, 3.17471036e-01],
 [2.66007653e-03, 3.67230479e-01, 6.29014372e-01, 1.96398889e-03,
  2.09687779e-01, 6.88584444e-02, 1.04473540e-03, 1.66200913e-03,
  5.71121520e-02, 2.48731624e-01],
 [1.48119766e-02, 6.94147224e-02, 2.19521612e-01, 7.77221469e-05,
  3.55630062e-01, 6.75082238e-02, 2.74672488e-03, 2.27507296e-02,
  1.08323834e-01, 5.64470874e-02],
 [6.33918377e-03, 6.19161852e-01, 1.68657537e-01, 1.00590385e-03,
  7.09781431e-01, 1.61946122e-01, 1.41711669e-02, 7.45232774e-03,
  5.12913071e-03, 4.54698070e-01],
 [1.76667410e-03, 1.06646955e-01, 4.22822062e-01, 2.03230029e-03,
  3.93195670e-01, 6.73791782e-02, 1.06197385e-03, 4.85782502e-03,
  3.14639003e-02, 2.65898240e-01],
 [3.07488081e-03, 1.07279140e-02, 3.81498227e-01, 1.82261830e-04,
  3.59054416e-01, 8.14835673e-02, 8.50859167e-03, 4.80215800e-02,
  3.72343039e-01, 1.21333927e-02]])
```

Gracias por la atención