



# Perceptrón simple y multicapa - TP3

Grupo 11



# ¿Cómo se construyó?

## Perceptron Multicapa

- Cero o mas capas ocultas
- Tasa de aprendizaje adaptativa.
- Momentum.
- Función g y su derivada.
- Iteración por lotes o incremental.

## Implementación de Algoritmos

- Clase con definición de funciones y sus derivadas.

## Inicialización de pesos

- $$\frac{1}{(layer+1)^{1/2}} * random(0, 1)$$

1

# Implementacion



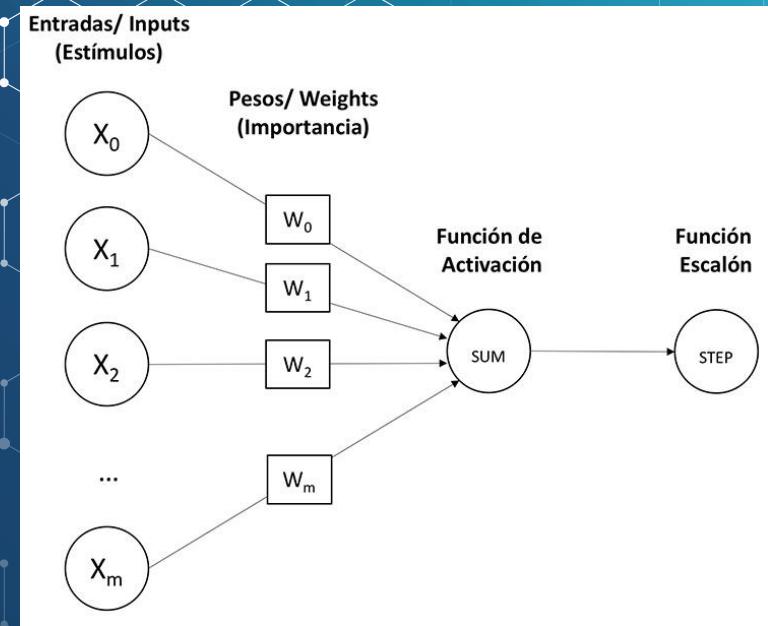
# Perceptron

- Normalización de los datos de entrada y salida.
- Momentum arbitrario.
- Cálculo de  $\eta$ 
  - ◆ minimizando función de error
  - ◆ incremento/decremento respetando un minimo

2

# Ejercicios

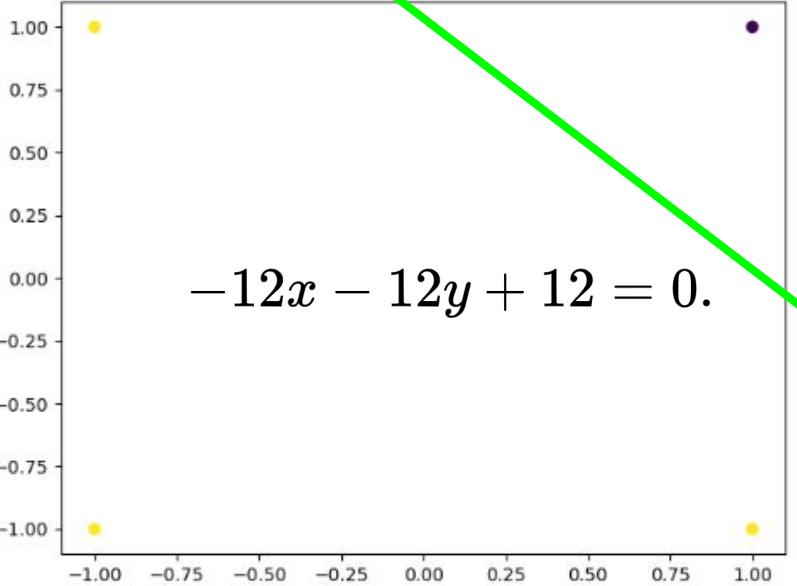
# Ejercicio 1



# Resolución de Problemas

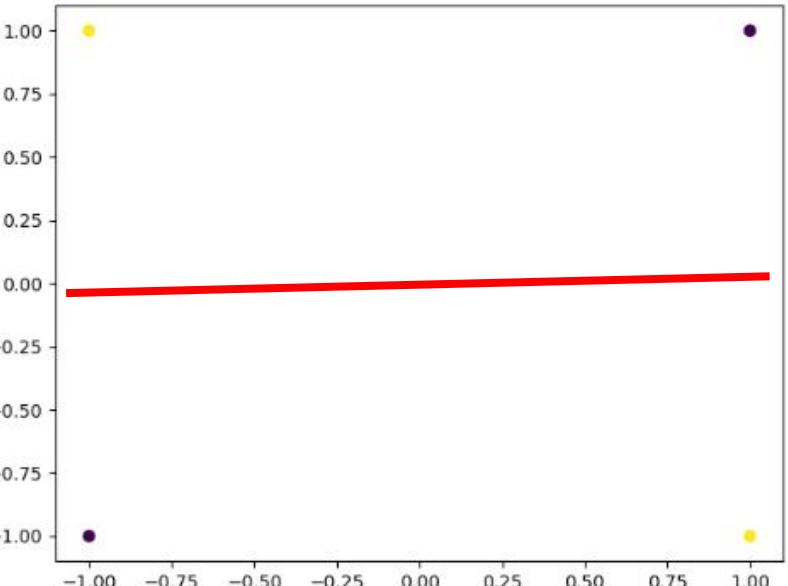
Función Y

$$-12x - 12y + 12 = 0.$$



Inicializando  $w$  en 0

Función O Exclusivo





# O Exclusivo: ¿Por qué no?

Perceptrón Simple Escalón  $\rightarrow$  Linealmente Separables

$$\begin{cases} -w_1 + w_2 = 1 - w_0 \\ w_1 - w_2 = 1 - w_0 \\ -w_1 - w_2 = -1 - w_0 \\ w_1 + w_2 = -1 - w_0 \end{cases}$$

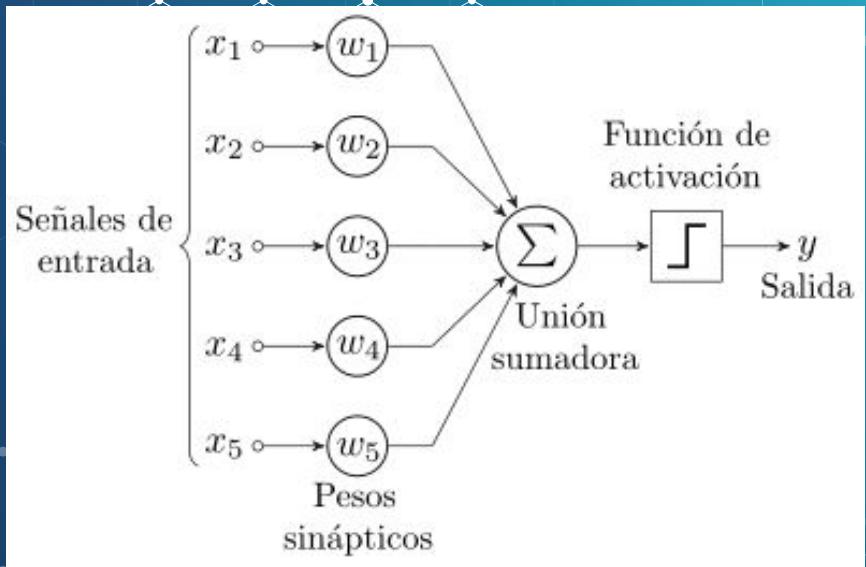
$$+ \longrightarrow 0 = 2 - 2 * w_0$$

$$+ \longrightarrow 0 = -2 - 2 * w_0$$

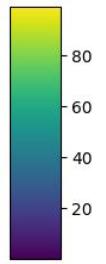
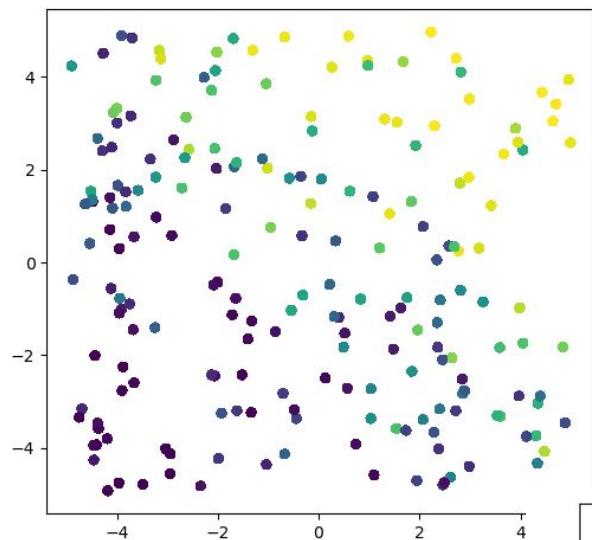
---

$$0 = 4. Abs!$$

# Ejercicio 2



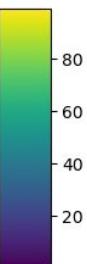
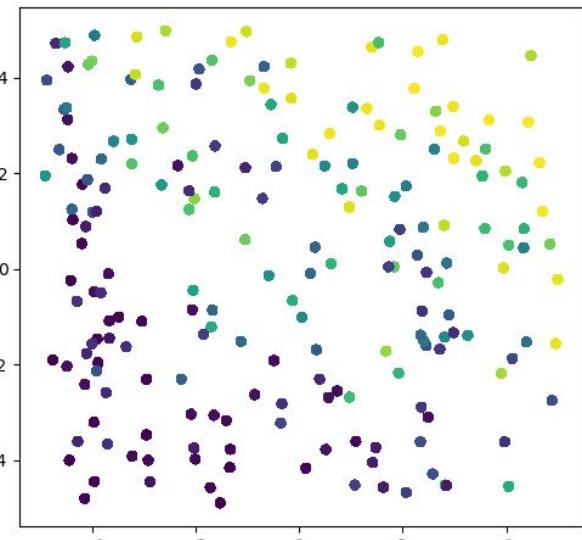
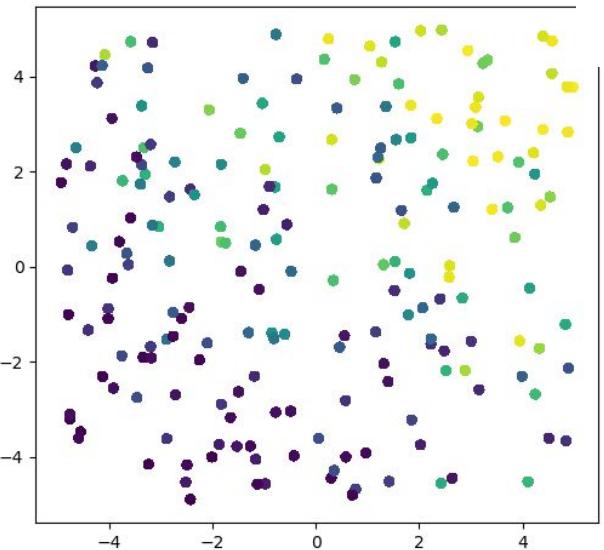
(X,Z)



(X,Y)



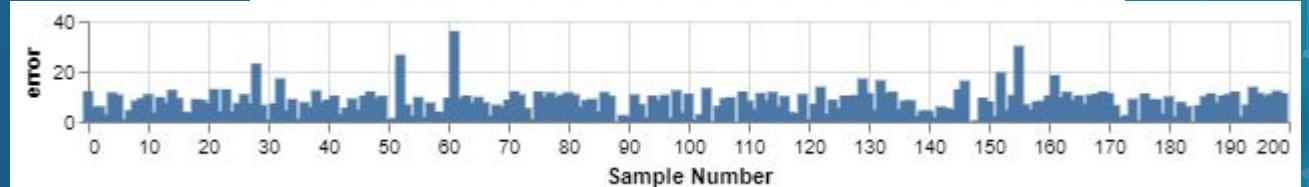
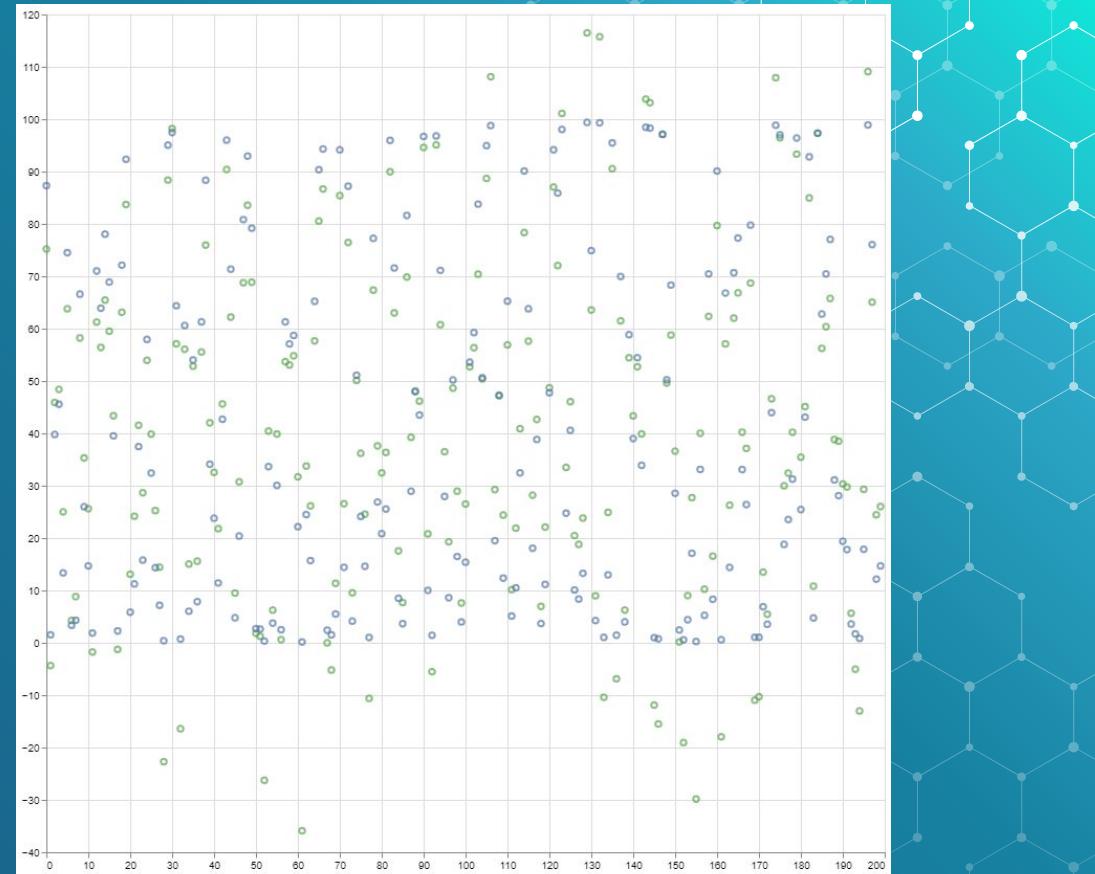
(Y,Z)



# Lineal

$$f(x, y, z) = x * 42.862 + y * 6.4177 + z * 6.5983 + 6.8412$$

Finalización por iteraciones.  
Estancamiento (epochs >500)

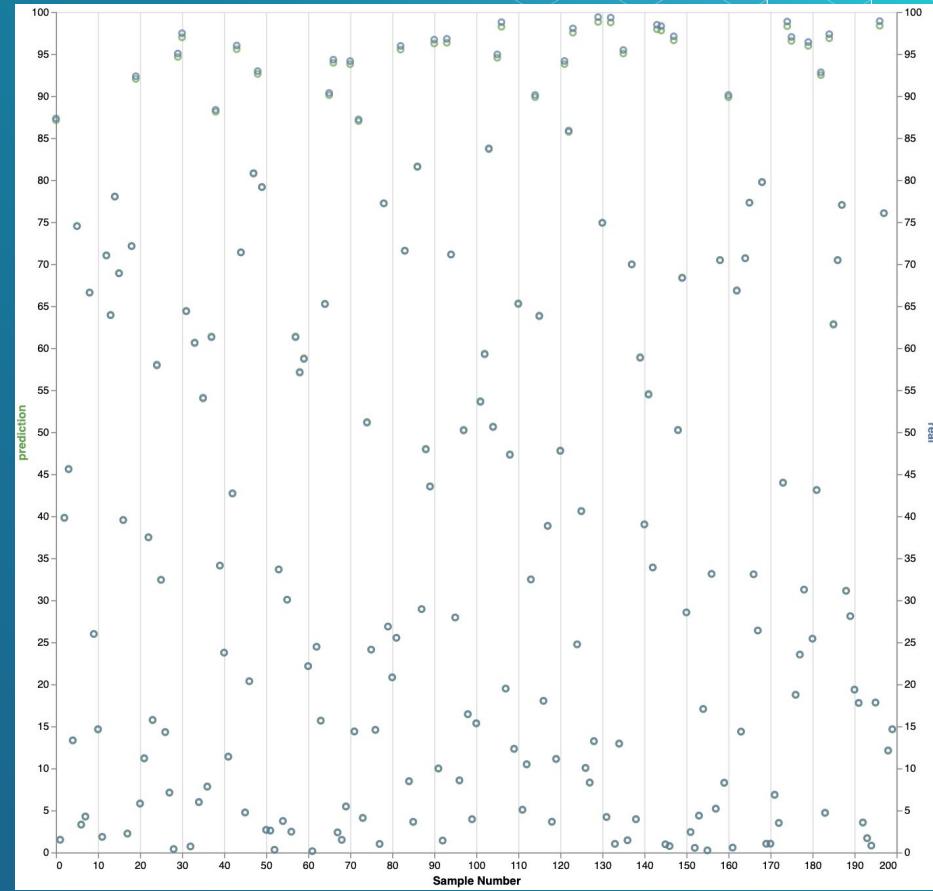


# No Lineal

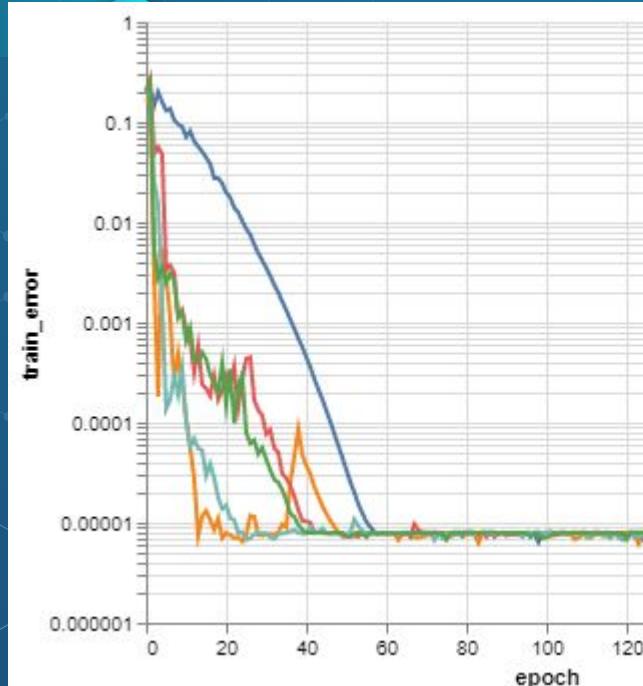
$$f(x, y, z) = g(-x * 8.0480 + y * 5.0399 + z * 5.0415 + 5.0395, 1)$$

$$g(h, \beta) = \frac{1}{1+e^{-h*\beta}}$$

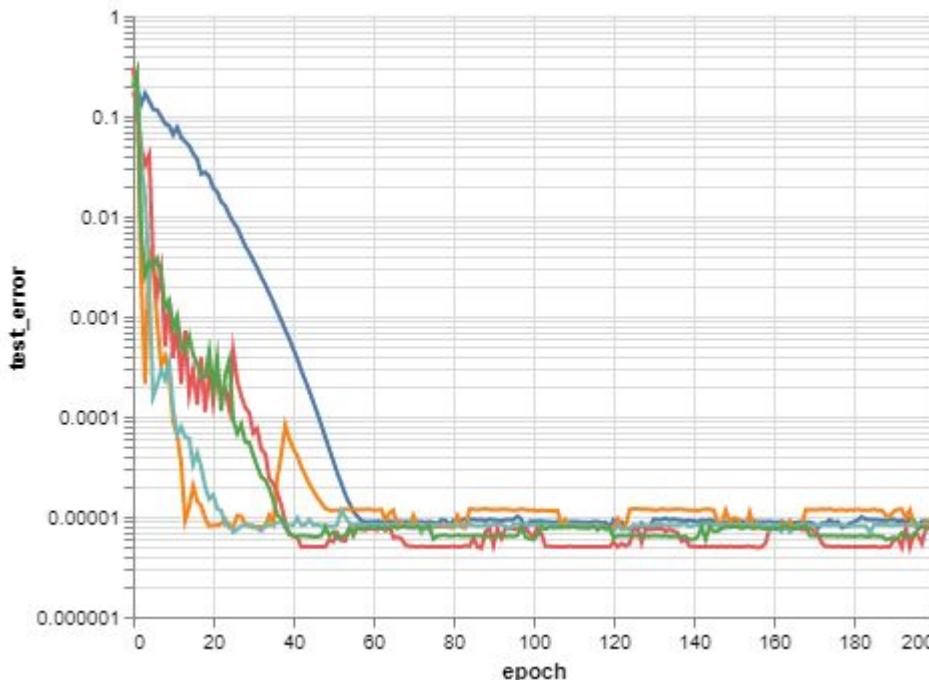
Precisión: 0.00033



# Conjunto Entrenamiento



Train Set Error



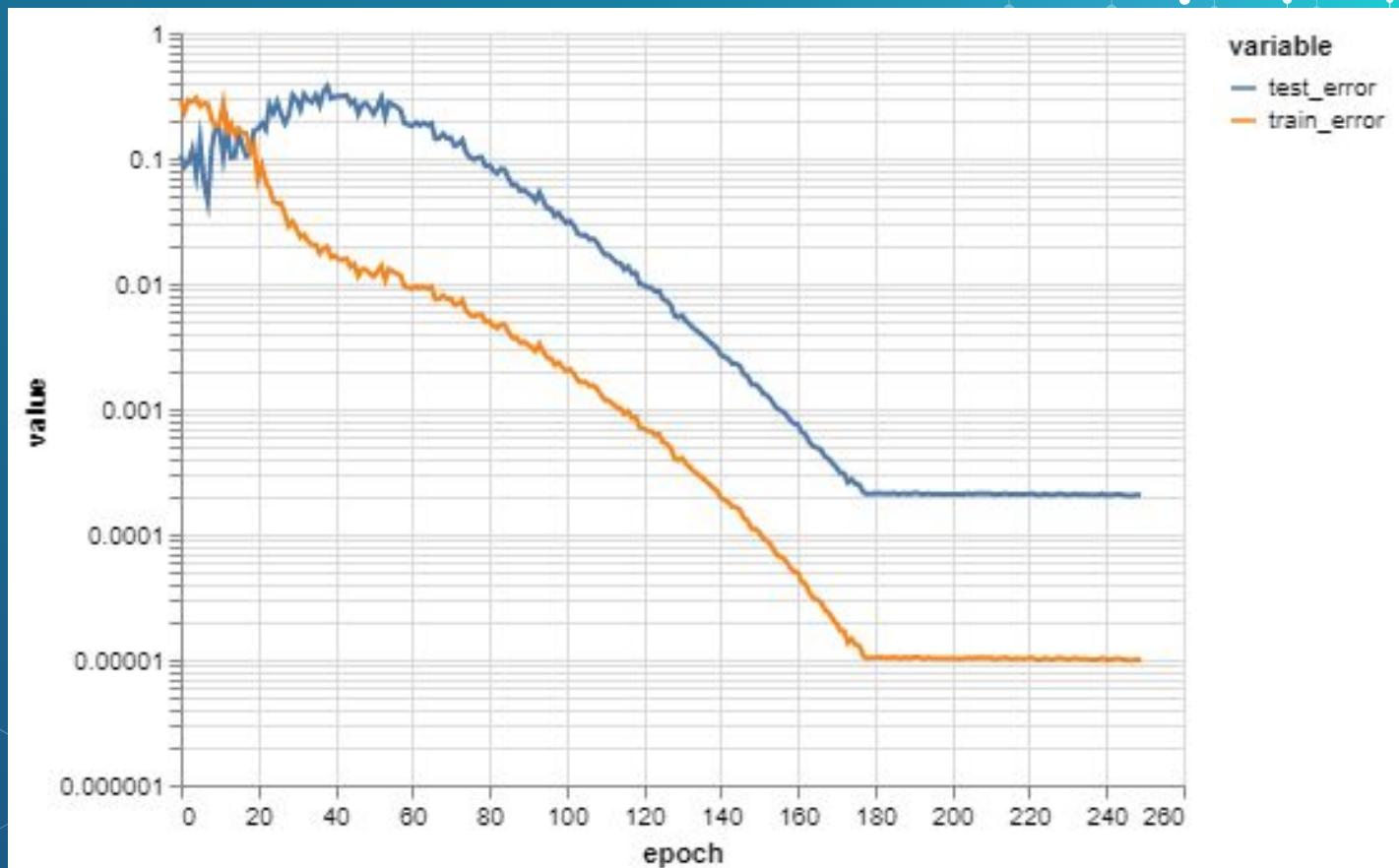
Test Set Error

# Conjuntos Erróneos

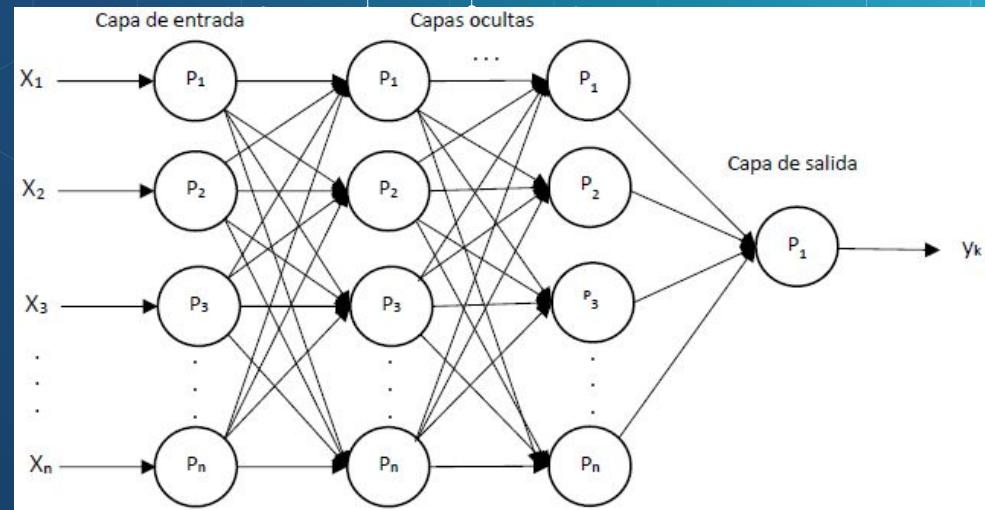
Entrenamiento con los valores de menores que la mitad:  
 $(x,y,z)$  tal que  
 $f(x,y,z) < \langle f \rangle$

Train: 120 (60%)

Test: 80 (40%)



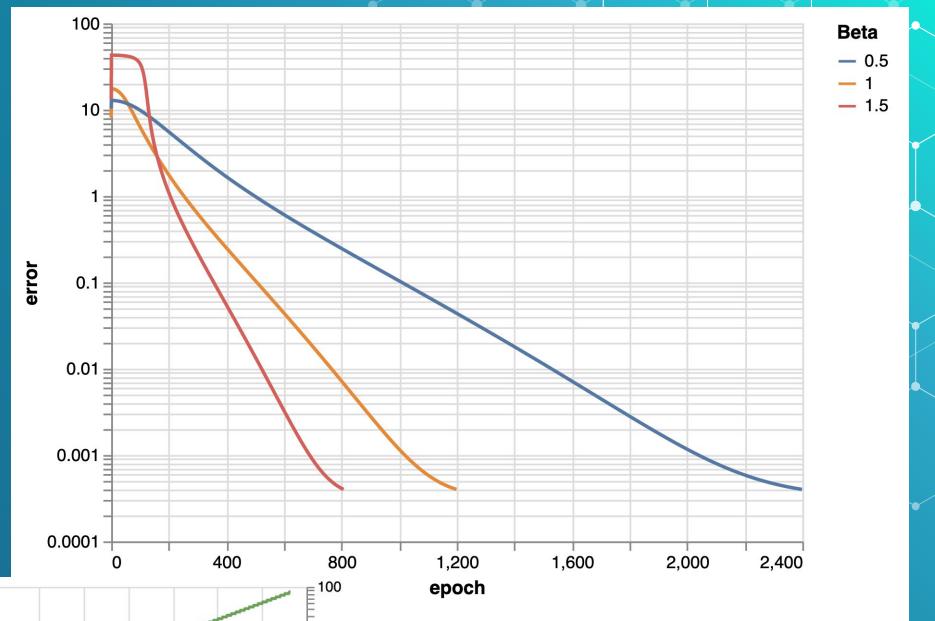
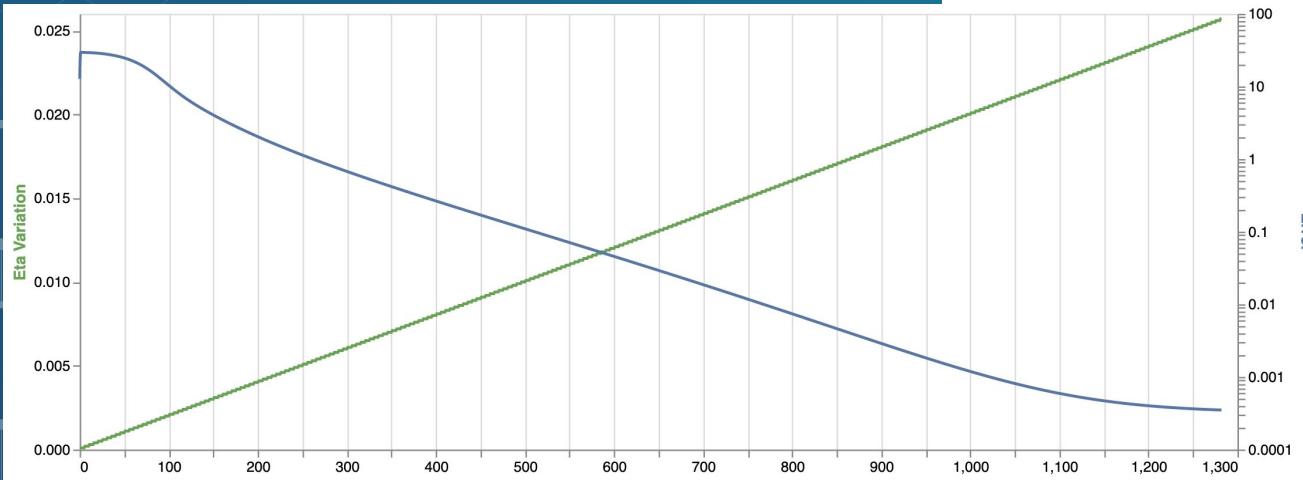
# Ejercicio 3



# Variación de $\beta$

$$g(h, \beta) = \frac{1}{1+e^{-h*\beta}}$$

DataSet Ejercicio 2



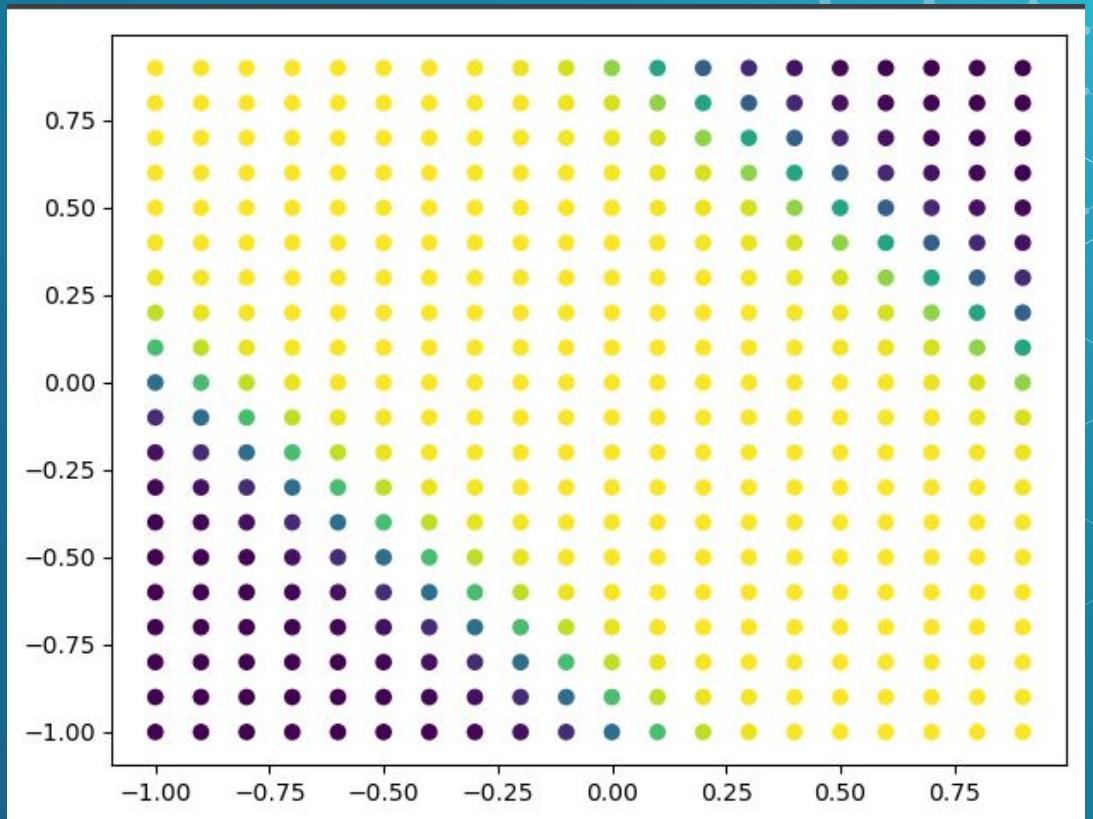
Variación de  $\eta$

# Función O Exclusivo

3 nodos Capa  
Oculta

$$g(h, \beta) = \frac{1}{1+e^{-h*\beta}}$$

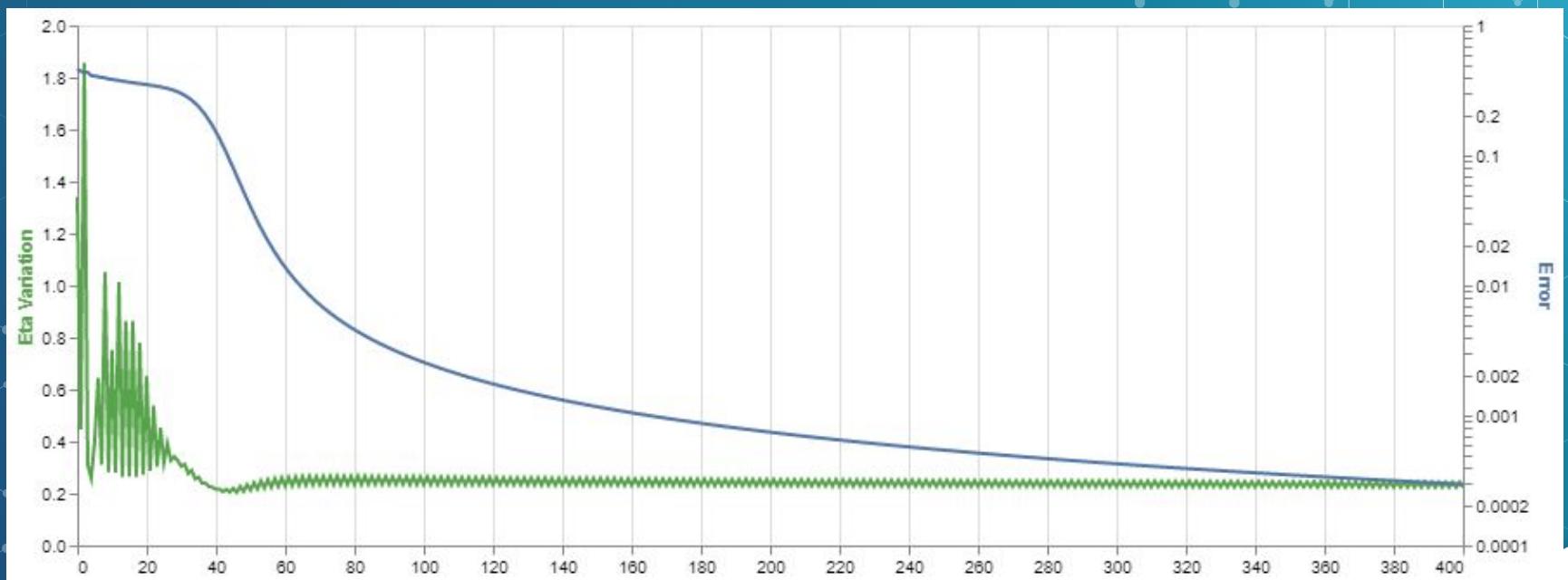
$\beta = 1$



# Cálculo de mejor Eta

## Función O Exclusivo

Capa Oculta con 3 Neuronas. Momentum = 0



# Reconocimiento de Números

10 valores de salida [0,1]

	0	1	2	3	4	5	6	7	8	9
0	0.987	0	0.006	0.009	0.01	0	0.01	0	0	0
1										

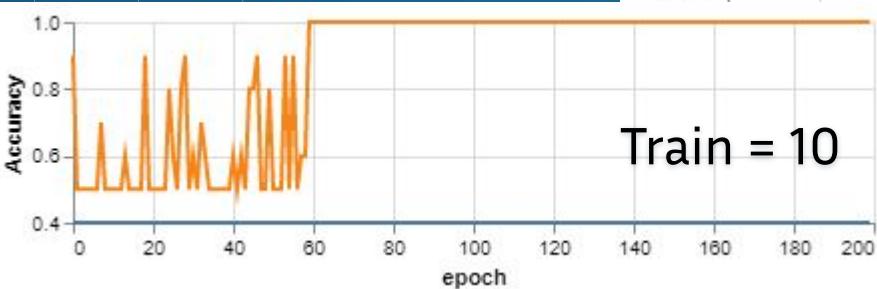
# Reconocimiento de Números Pares

1 valor de salida [0,1]

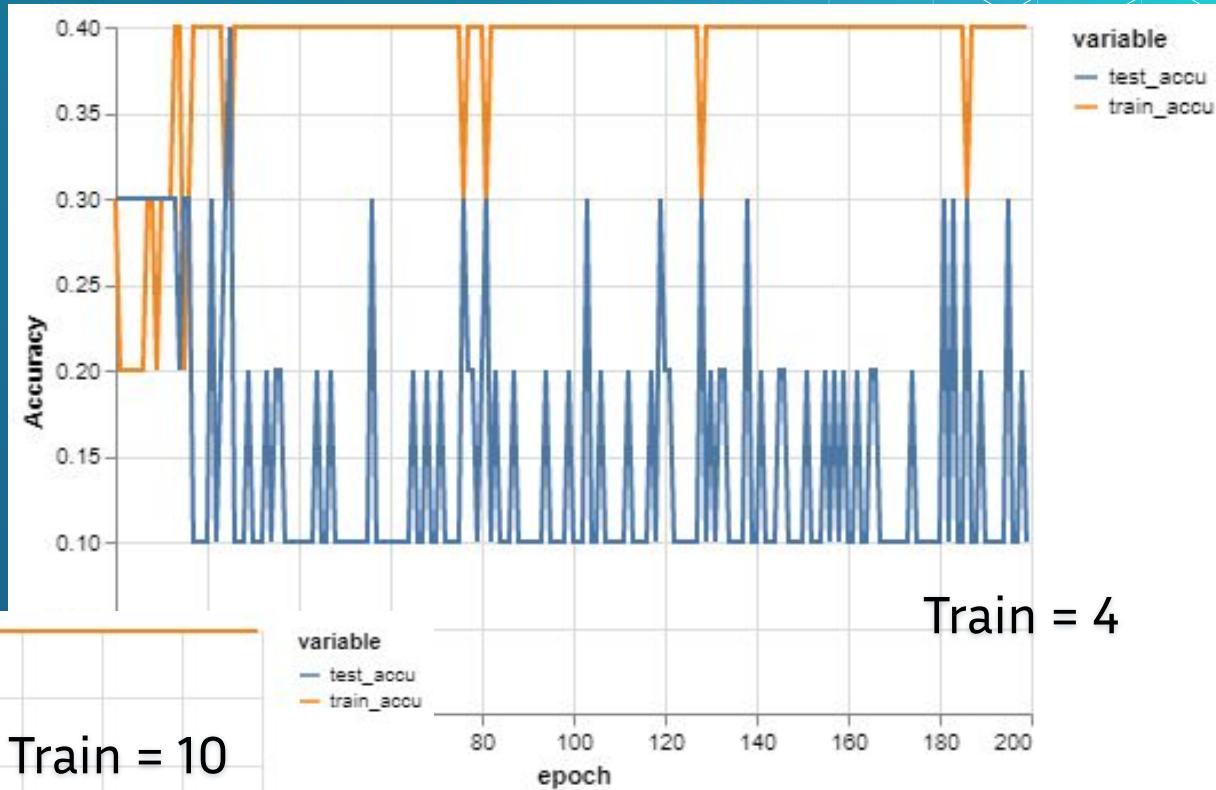
Capa Oculta 3 Neuronas

Es par si  $g(h) < 0.5$ ,

Es impar en otro caso.



Train = 10



Train = 4

# Matriz de Confusión

Positivo es valer 1 -> Impar.

Train = 8	Positive	Negative
Positive	0	1
Negative	0	1

Train = 4	Positive	Negative
Positive	0	3
Negative	2	1

Train = 6	Positive	Negative
Positive	0	2
Negative	2	0

Train = 2	Positive	Negative
Positive	0	4
Negative	0	4

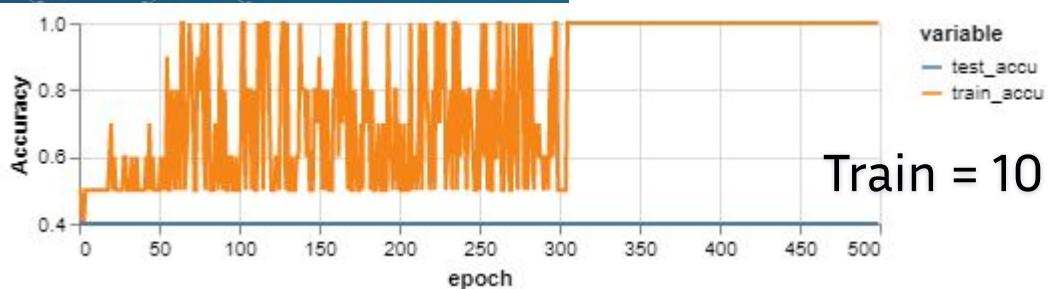
# Reconocimiento de Números Pares

1 valor de salida [0,1]

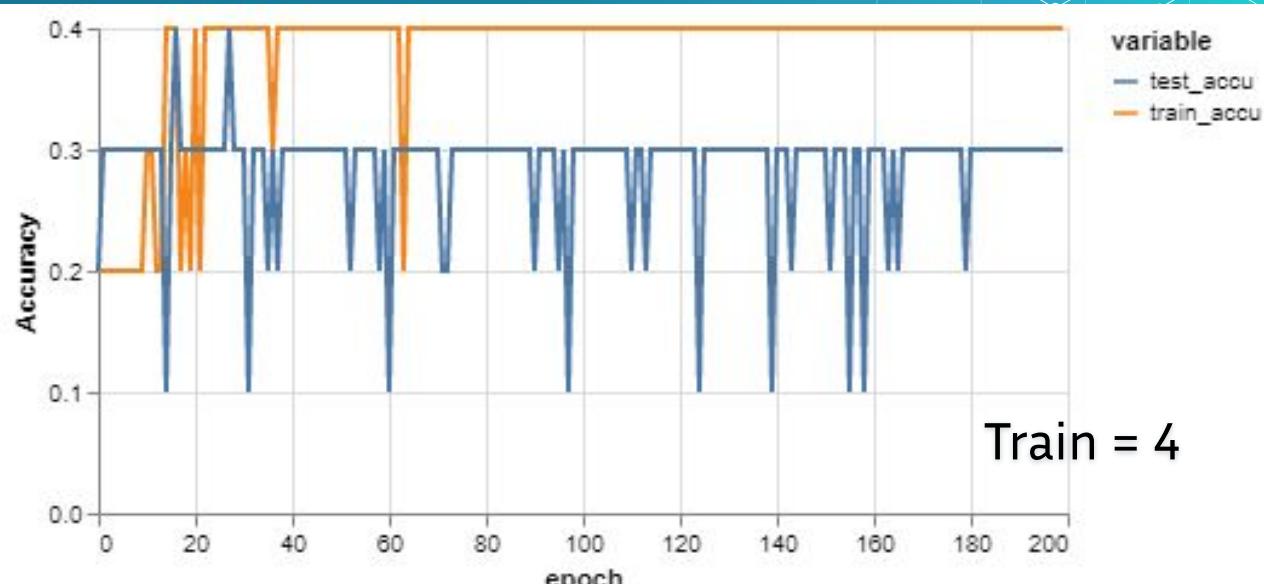
Capa Oculta 3 Neuronas

Es par si  $g(h) < 0.1$ ,

Es impar en otro caso.



Train = 10



Train = 4

# Matriz de Confusión

Positivo es valer 1 -> Impar.

Train = 8	Positive	Negative
Positive	1	0
Negative	1	0

Train = 4	Positive	Negative
Positive	3	0
Negative	3	0

Train = 6	Positive	Negative
Positive	2	0
Negative	2	0

Train = 2	Positive	Negative
Positive	4	0
Negative	4	0



# CONCLUSIONES

El cálculo del  $\eta$  que minimiza el error es costoso computacionalmente. Utilizar solo al principio o cada N pasos.

El momentum ayuda en el estancamiento como en la oscilación.

Ej. 3: la red no puede generalizar ya que no es posible hacerlo con la información de entrada.

El conjunto de entrenamiento debe poseer la mayor diversidad.  
(Contrario a lo realizado en el ej2.)

$\eta$  adaptativo tiene buenos resultados sin compromisos en el cómputo