

# TP2 - Red Neuronal



Grupo 12

Pedro Balaguer - 55795

Alexander Dryselius - 60649

Agustín Izaguirre - 57774

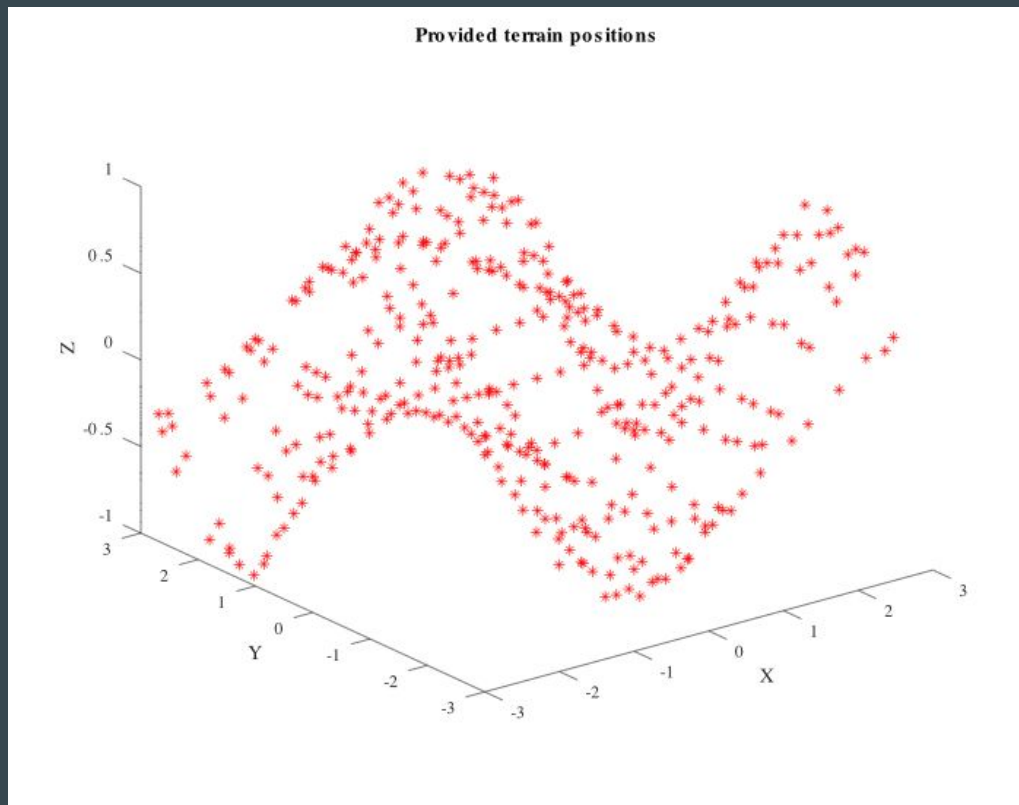
Juan Li Puma - 55824

# Generalidades

Red neuronal feed-forward:

- Los pesos se inicializan con valor aleatorio en el intervalo  $[-n, +n]$
- Funciones de activación lineal, *exp* y *tanh* (excepto para capa de salida)
- La función de costo corresponde al error cuadrático medio entre el valor obtenido y la salida esperada de los patrones
- Normalización de los datos de entrada ( $v = v/norm(v)$ )

# Superficie a Aprender



# Error tolerable

El codominio de la función del terreno es  $[-1 ; 1]$  y todos los valores de entrada tienen una precisión de  $10^{-5}$  por lo tanto consideramos un error aceptable el de:

$$\text{epsilon} = 0.050$$

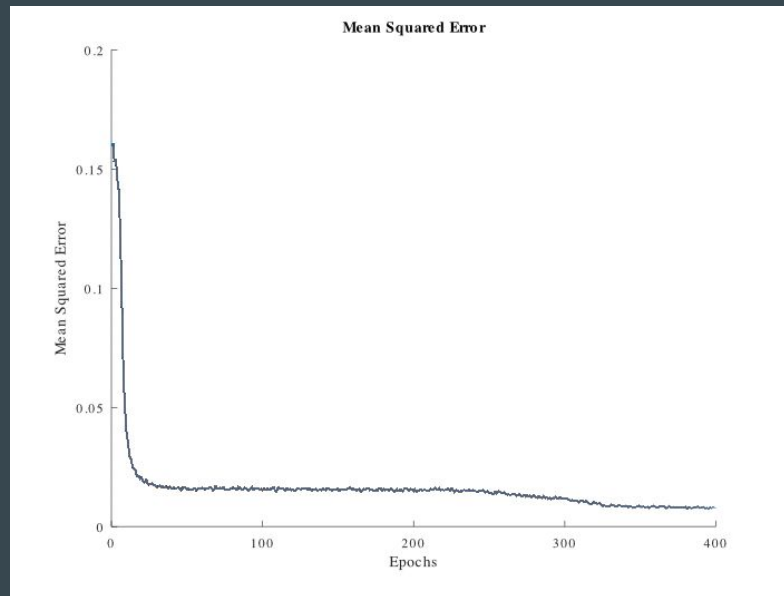
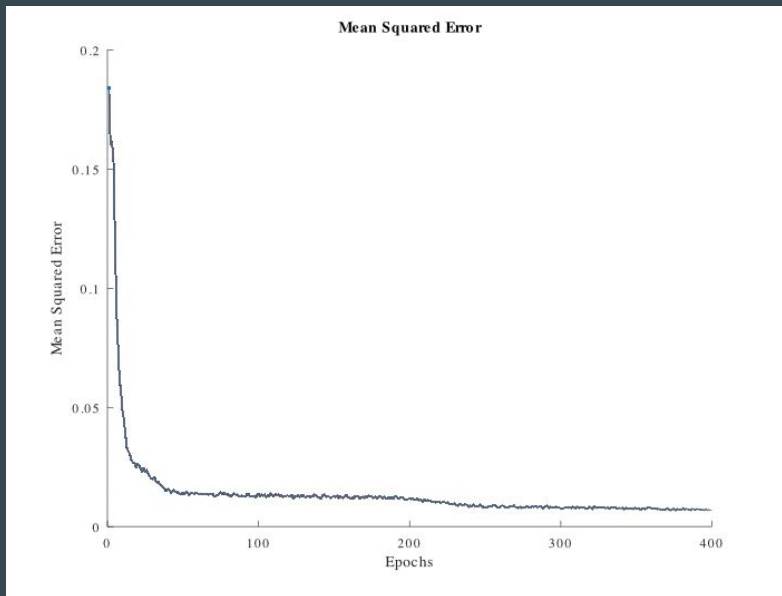
Por lo tanto:

$$\text{ECM} = \text{epsilon}^2/2 = 0.00125$$

# Máxima cantidad de épocas

Determinado empíricamente: la red no aprende significativamente después de **400** épocas.

Ejemplos:



# Optimizaciones: momentum

$$W_{\text{nuevo}} = W_{\text{actual}} + \Delta W_{\text{actual}} + \alpha * \Delta W_{\text{anterior}}$$

Permite tomar en cuenta en cierto grado la actualización de pesos previa.

NOTA:  $\alpha$  es configurable

# Optimizaciones: $\eta$ adaptativo

- Si el ECM aumenta de una época a otra entonces:

$$\eta = \eta - b\eta \text{ (b configurable)}$$

Además se cambia el valor  $\alpha$  del momentum a 0 (hasta que el error decremente) y se vuelven a los pesos de la época anterior.

- Si por K épocas consecutivas el error disminuye (K configurable):

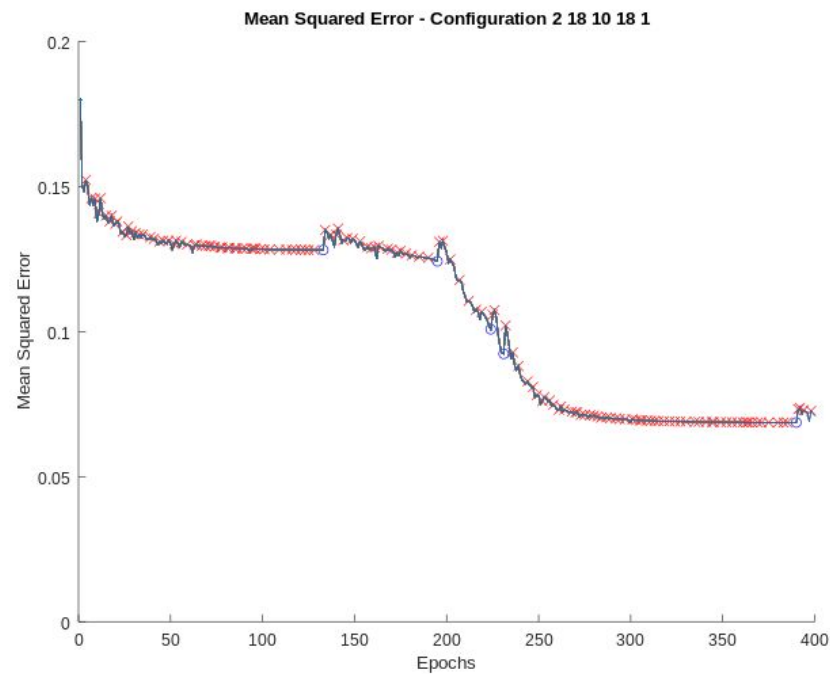
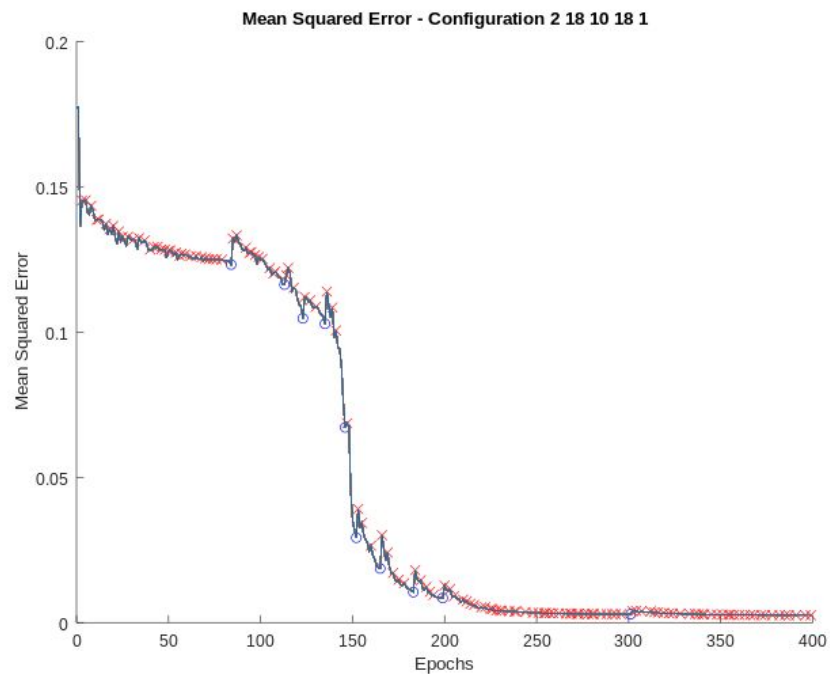
$$\eta = \eta + a \text{ (a configurable)}$$

# Optimizaciones: $\eta$ adaptativo sin memoria de pesos

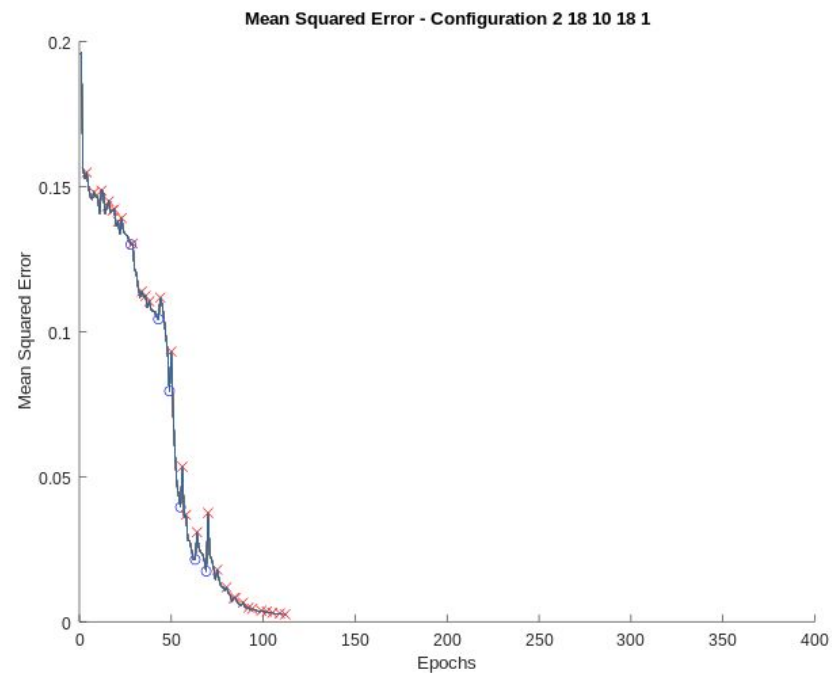
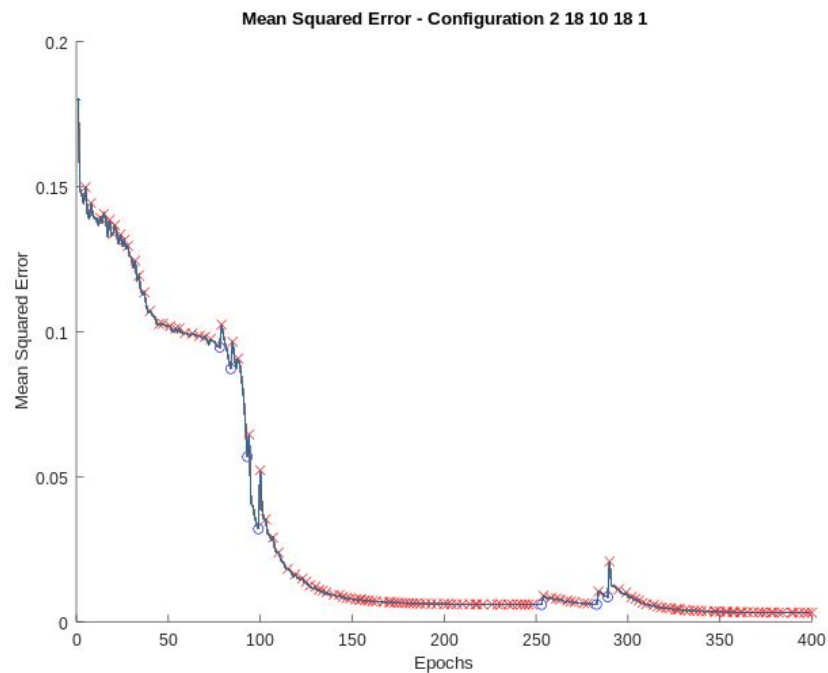
Otra variante de la optimización de  $\eta$  adaptativo. La única diferencia es que no se vuelve a los pesos del estado anterior cuando el error aumenta. Esto permite salir de **mínimos locales** en algunas situaciones. Es un primer acercamiento a una implementación de *simulated annealing*.



# Memoria Activada



# Memoria Desactivada



# Problemas Encontrados: Batch

En muchos casos el valor de los deltas aumenta o disminuye exponencialmente, llegando a  $\pm\infty$  por falta de precisión de punto flotante. Esta situación se da por un producto escalar de una dimensión muy grande (que depende de la cantidad de patrones de entrada).

# Resultados: Arquitectura Óptima

Ejecuciones con 1, 2 y 3 capas ocultas. Para determinar qué red es más óptima se tomaron en cuenta tres criterios en el orden de importancia siguiente:

- 1) Menor error cuadrático medio
- 2) Velocidad de reducción del ECM por época
- 3) Cantidad de unidades en la red

# Resultados: Arquitectura Óptima

Épocas	Error	Cantidad de neuronas en capa oculta #1
400	0.006301	30
400	0.006372	28
400	0.006437	24
400	0.006493	22
400	0.006725	18

1 capa oculta

# Resultados: Arquitectura Óptima

Épocas	Error	Cantidad de neuronas en capa oculta #1	Cantidad de neuronas en capa oculta #2
400	0.0050090	22	6
282	0.0024108	14	10
244	0.0023631	14	6
400	0.0042462	18	6
361	0.0024436	10	22

2 capas ocultas

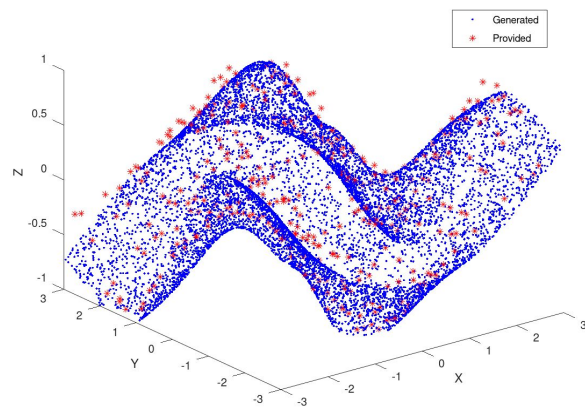
# Resultados: Arquitectura Óptima

Épocas	Error	Cantidad de neuronas en capa oculta #1	Cantidad de neuronas en capa oculta #2	Cantidad de neuronas en capa oculta #3
159	0.0022577	18	10	18
166	0.0024389	26	26	10
189	0.0022340	10	10	18
179	0.0023109	18	18	18
199	0.0024438	18	18	10

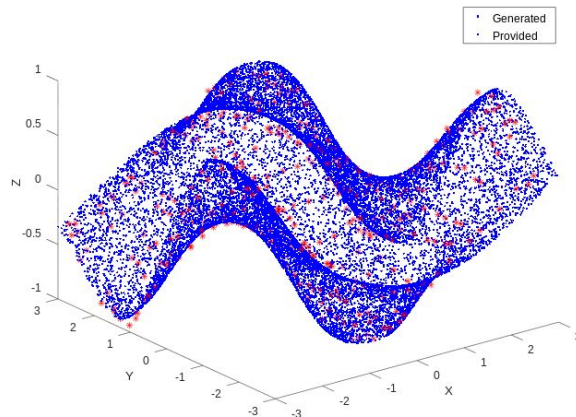
3 capas ocultas

# Resultados: Arquitectura Óptima

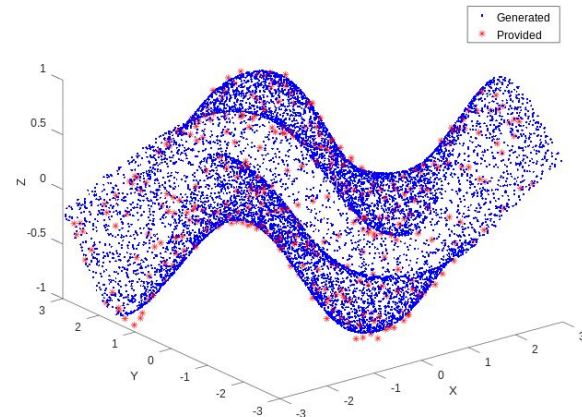
Both terrain positions - Configuration 2 30 1



Both terrain positions - Configuration 2 10 10 1



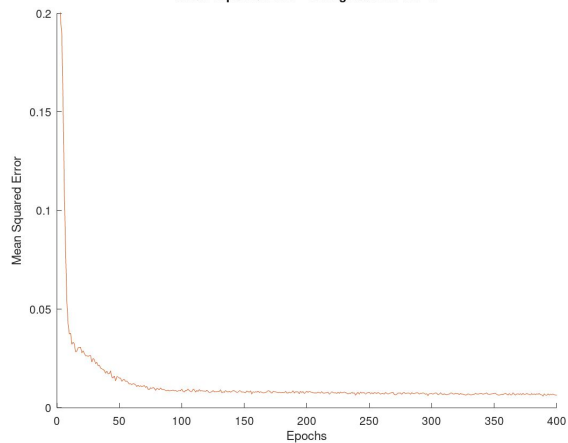
Both terrain positions - Configuration 2 18 10 18 1



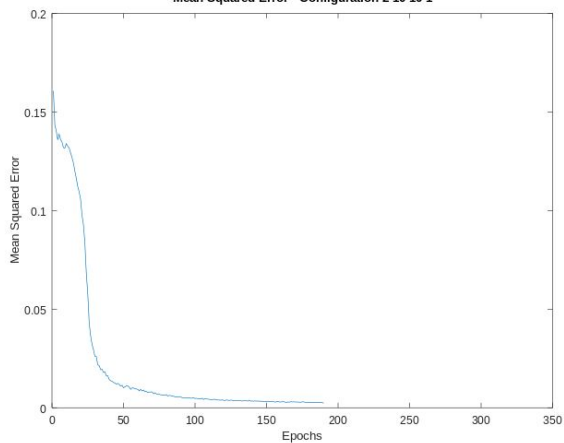


# Resultados: Arquitectura Óptima

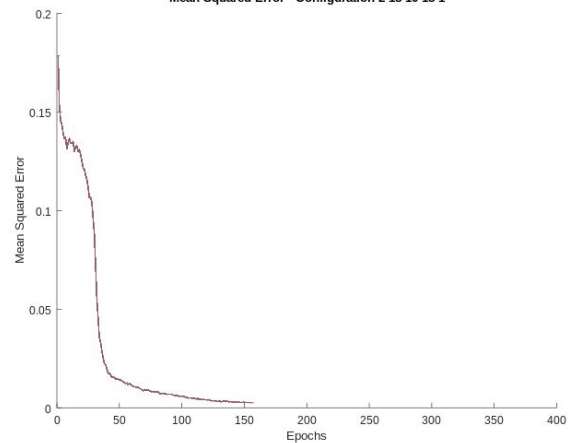
Mean Squared Error - Configuration 2 28 1



Mean Squared Error - Configuration 2 10 10 1



Mean Squared Error - Configuration 2 18 10 18 1



# Resultados: Arquitectura Óptima

La configuración que minimiza el error es una de tres capas ocultas con **18, 10 y 18** unidades respectivamente.

# Resultados: Factor de Aprendizaje Inicial

Factor de aprendizaje inicial	Error
0.1	0.002357
0.2	0.002262
0.3	0.002325
0.4	0.002357
0.5	0.004361
0.6	0.006950
0.7	0.005100
0.8	0.134363
0.9	0.210761

# Resultados: Factor de Aprendizaje Inicial

Factor de aprendizaje inicial	Error
0.21	0.002426
0.22	0.002387
0.23	0.002417
0.24	0.002433
0.25	0.002162
0.26	0.002414
0.27	0.002370
0.28	0.002249
0.29	0.002229

# Resultados: $\alpha$ del momento

Alfa del Momento	Error
0.1	0.001493
0.2	0.001439
0.3	0.001489
0.4	0.001688
0.5	0.001847
0.6	0.001605
0.7	0.003923
0.8	0.003447
0.9	0.236886

# Resultados: $\alpha$ del momento

Alfa del momento	Error
0.2	0.001439
0.21	0.001447
0.22	0.001456
0.23	0.001463
0.24	0.001471
0.25	0.001478
0.26	0.001483
0.27	0.001488
0.28	0.001490
0.29	0.001489

# Resultados: Constantes de $\eta$ Adaptativo

Épocas	Error	Constante de aumento	Factor de decrecimiento
194	0.002446	0.0875	0.1
196	0.002332	0.125	0.1
400	0.003255	0.125	0.25
400	0.002628	0.05	0.1
400	0.004069	0.0875	0.25

Constante de Aumento = 0.125 y Factor de Decrecimiento = 0.1

# Resultados: Funciones de Activación

Función exponencial:

Beta	Épocas	Error
0.25	400	0.137400
0.5	400	0.006709
1	264	0.001237
2	400	0.002074

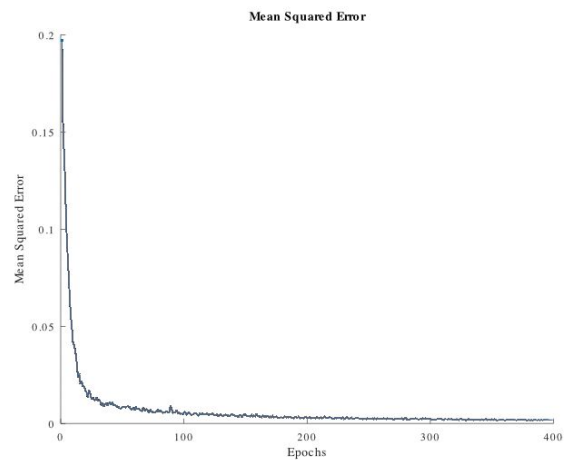
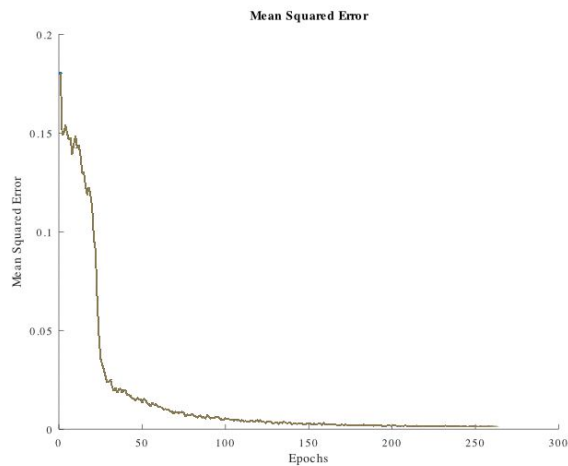
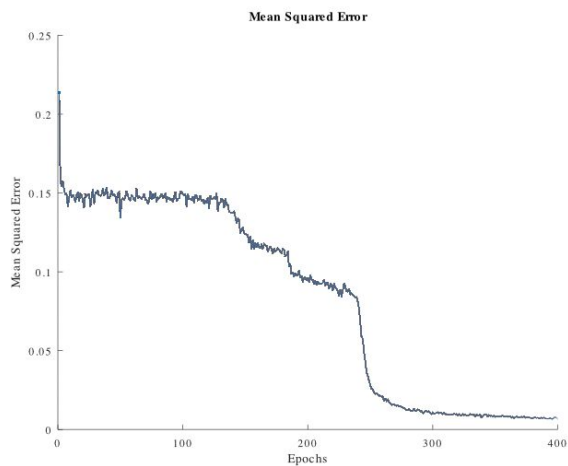
Función tangente hiperbólica:

Beta	Épocas	Error
0.25	400	0.009320
0.5	400	0.002442
1	400	0.002837
2	-	-



# Resultados: Funciones de Activación

Función exponencial:



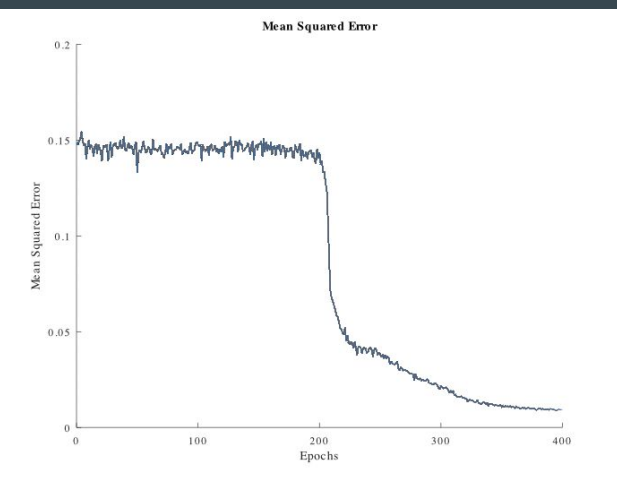
Beta = 0.5

Beta = 1

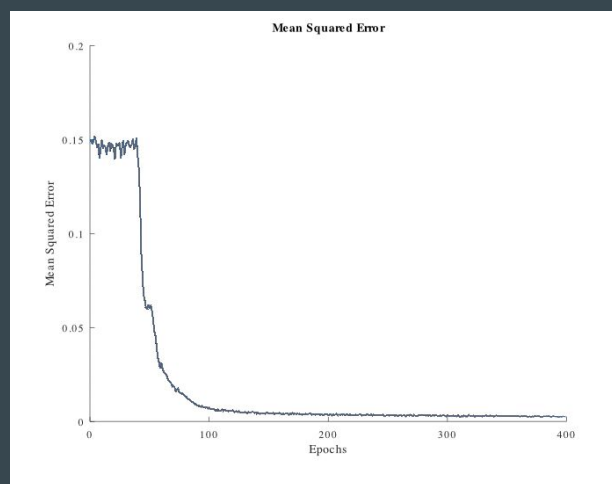
Beta = 2

# Resultados: Funciones de Activación

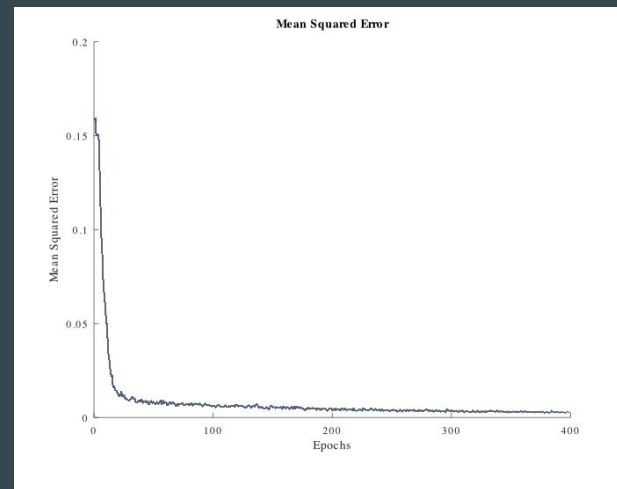
Función tangente hiperbólica:



$\beta = 0.25$



$\beta = 0.5$



$\beta = 1$

# Resultados: Funciones de Activación

En ambos casos, tanto para *exp* como para *tanh*, un valor de beta más bajo desacelera la bajada del error.

Para nuestro problema la función de activación exponencial provee resultados levemente mejores que la tangente hiperbólica.

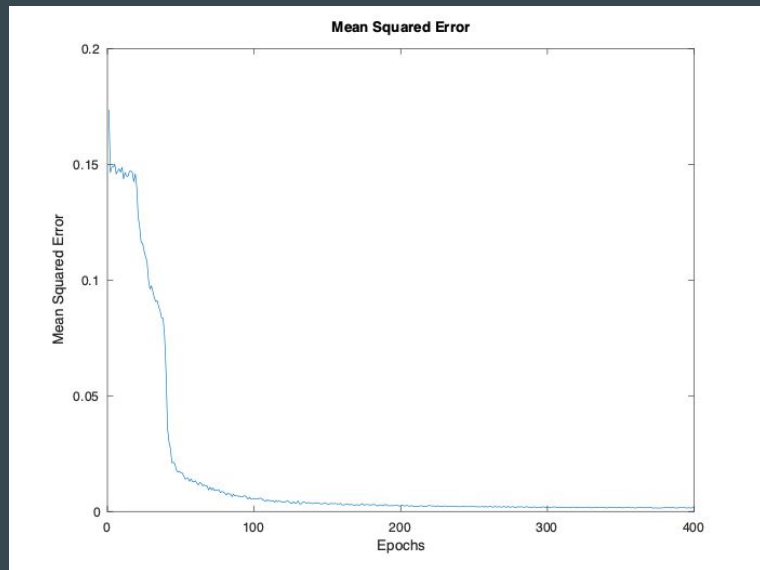
# Resultados: Inicialización de Pesos

	1er Corrida	2da Corrida	3er Corrida	4ta Corrida	5ta Corrida
Error con Inicialización random [-0.5 , 0.5]	0.00147774970116330	0.0015861206137081	0.00189960171157504	0.0023875456431415	0.0013971638065032245
Error con Inicialización con fan-in	0.00147774970116335	0.0015861206137082	0.00189960171157502	0.0023875456431416702	0.001397163806503228

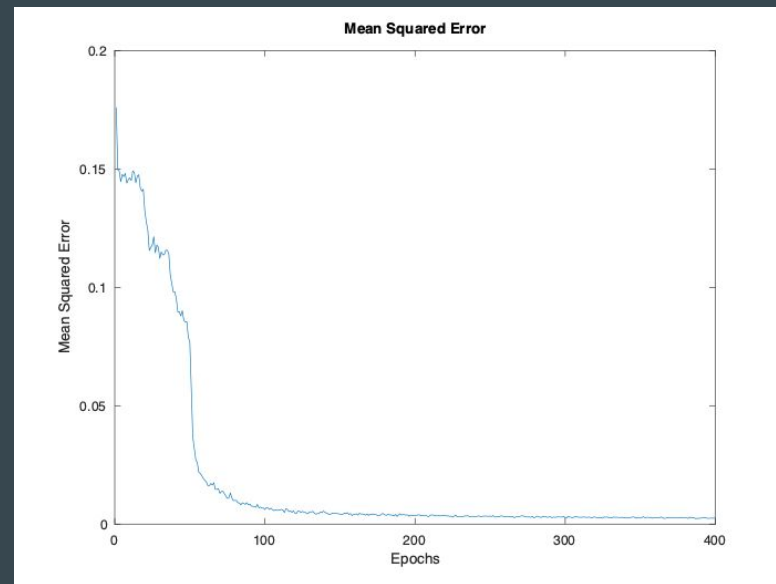
Error obtenido según inicialización de pesos

La diferencia de rendimiento no es apreciable entre una inicialización aleatoria y con fan-in.

# Resultados: Inicialización de Pesos



Inicialización aleatoria



Inicialización utilizando fan-in

No hay tampoco una diferencia apreciable de la evolución del error.

# Resultados: Porcentaje de Entrenamiento

Porcentaje de Entrenamiento (%)	Error
10	0.070233
20	0.003167
30	0.001745
40	0.002406
50	0.002066
60	0.001959
70	0.002308
80	0.001250
90	0.001642

# Resultados: Porcentaje de Entrenamiento

Porcentaje de Entrenamiento (%)	Error
80	0.001250
81	0.001238
82	0.001309
83	0.001427
84	0.001182
85	0.001539
86	0.001324
87	0.001345
88	0.001114
89	0.001129

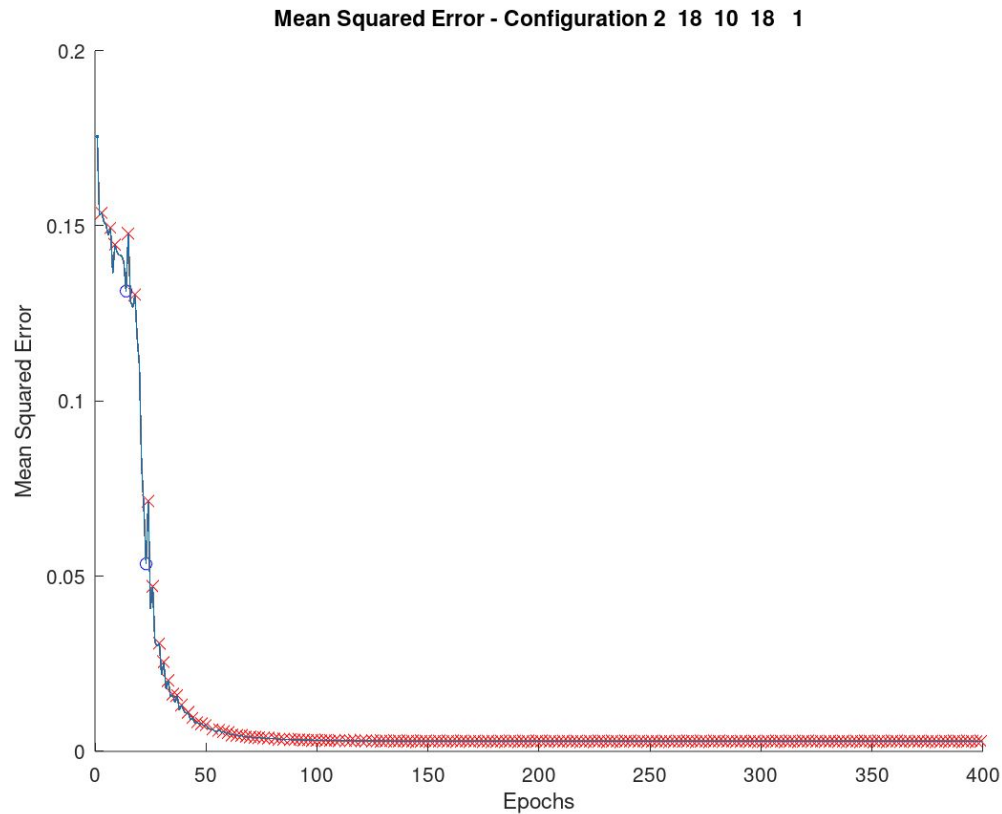
# Configuración Óptima

Tomando los resultados obtenidos podemos crear la configuración óptima de la red:

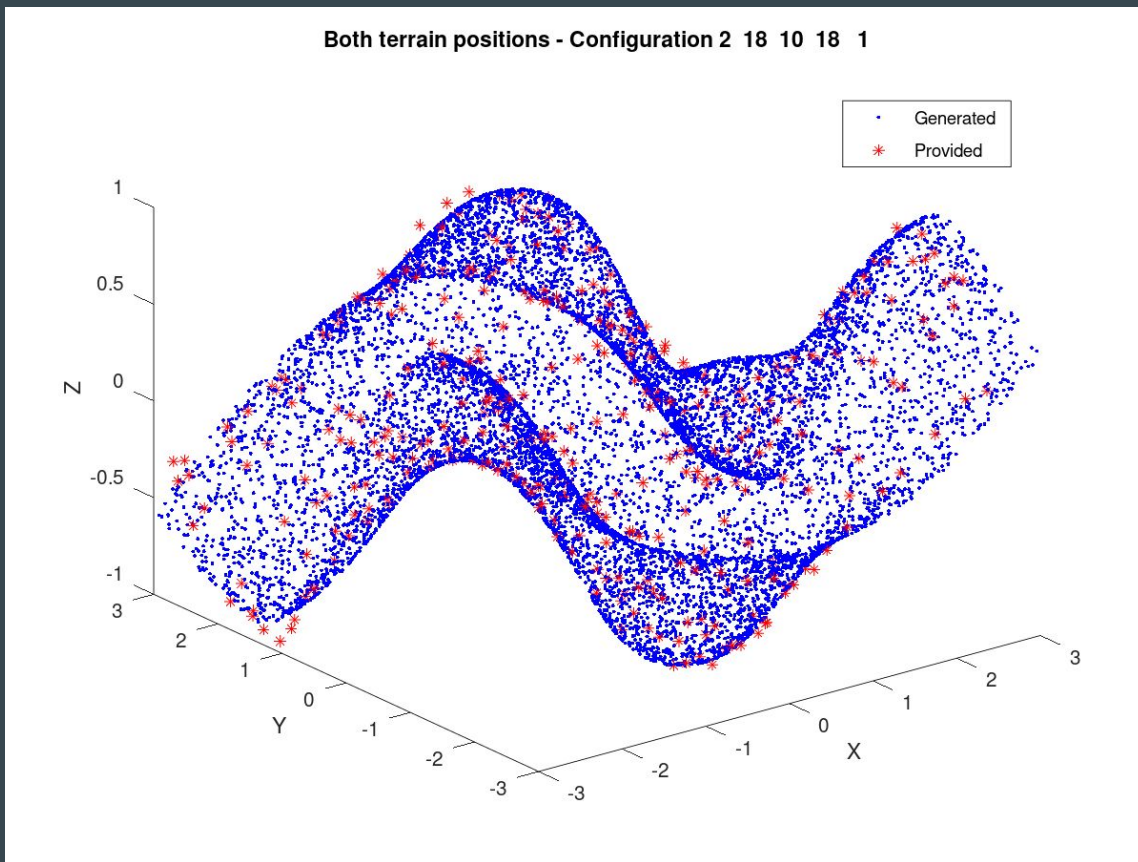
- Tres capas ocultas con 18, 10 y 18 unidades en cada una
- Eta inicial = 0.25
- Inicialización de pesos: aleatorio o fan-in
- Función de activación exponencial con  $\beta = 1$
- Alpha de momentum = 0.2
- Factor de decrecimiento del eta adaptativo = 0.125
- Factor de crecimiento del eta adaptativo = 0.1
- Porcentaje de entrenamiento = 88%



# Configuración Óptima



# Configuración Óptima



# Conclusiones

- Existen diferentes optimizaciones que permiten obtener un error menor o reducirlo más rápidamente (ej: momentum, eta adaptativo). Por ejemplo, en el caso de nuestra configuración óptima se llega a un error similar a la misma red sin optimizaciones en casi la mitad de épocas.
- Es posible encontrar rápidamente una configuración red que aprenda el problema, sin embargo llegado un punto es muy complicado seguir reduciendo el error, por lo que se vuelve más importante reducir la cantidad de épocas para obtener un error similar