

19 de Noviembre del 2018

Tercer avance del MLP

26 de Noviembre del 2018

Redes neuronales

Josue Ruiz Hernández

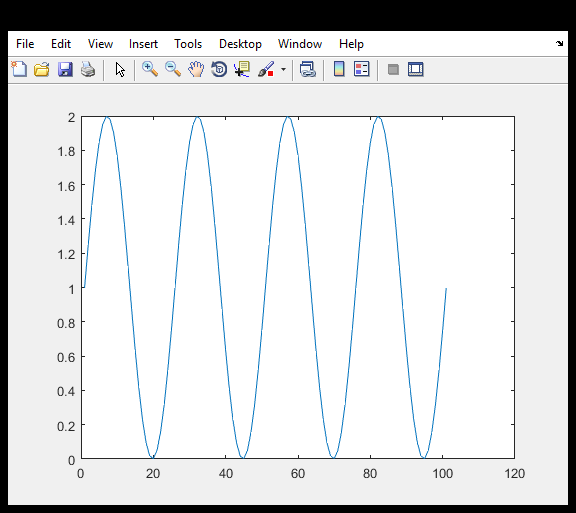
Juan Damian Osornio Gutierrez

ESCUELA SUPERIOR DE CÓMPUTO

Se aproximó la siguiente señal:

**01\_Polinomio\_Entrada**

**01\_Polinomio\_Target**



0.280000

0.320000

0.360000

0.400000

0.440000

0.480000

0.560000

0.600000

0.680000

0.720000

0.760000

0.800000

0.840000

0.880000

0.960000

1.000000

1.080000

1.120000

1.160000

1.200000

1.240000

1.280000

1.360000

1.400000

1.480000

1.520000

1.560000

1.600000

1.640000

1.680000

1.760000

1.800000

1.880000

1.920000

1.960000

2.000000

Conjunto de entrenamiento

-2.000000

-1.960000

-1.920000

-1.840000

-1.800000

-1.720000

-1.680000

-1.640000

-1.600000

-1.560000

-1.520000

-1.440000

-1.400000

-1.320000

-1.280000

-1.240000

-1.200000

-1.160000

-1.120000

-1.040000

-1.000000

-0.920000

-0.880000

-0.840000

-0.800000

-0.760000

-0.720000

-0.640000

-0.600000

-0.520000

-0.480000

-0.440000

-0.400000

-0.360000

-0.320000

-0.240000

-0.200000

-0.120000

-0.080000

-0.040000

0.000000

0.040000

0.080000

0.160000

0.200000

Conjunto de prueba:

-1.760000

-1.360000

-0.960000

-0.560000

-0.160000

0.240000

0.640000

1.040000

1.440000

1.840000

Conjunto de validación:

-1.880000

-1.480000

-1.080000

-0.680000

-0.280000

0.120000

0.520000

0.920000

1.320000

1.720000

Target de prueba:

1.998000

0.229500

1.248700

1.368100

0.155700

1.998000

0.229500

1.248700

1.368100

0.155700

Target de validación:

1.684500

0.874700

0.518200

1.904800

0.017700

1.684500

0.874700

0.518200

1.904800

0.017700

Target de entrenamiento:

1.770500

1.587800

1.368100

1.125300

0.631900

0.412200

0.095200

0.017700

0.002000

0.048900

0.155700

0.315500

0.751300

1.000000

1.481800

1.684500

1.844300

1.951100

1.998000

1.982300

1.770500

1.587800

1.125300

0.874700

0.631900

0.412200

0.229500

0.095200

0.002000

0.048900

0.315500

0.518200

0.751300

1.000000

1.000000

1.248700

1.481800

1.844300

1.951100

1.982300

1.904800

1.770500

1.587800

1.368100

1.125300

0.631900

0.412200

0.095200

0.017700

0.002000

0.048900

0.155700

0.315500

0.751300

1.000000

1.481800

1.684500

1.844300

1.951100

1.998000

1.982300

1.770500

1.587800

1.125300

0.874700

0.631900

0.412200

0.229500

0.095200

0.002000

0.048900

0.315500

0.518200

0.751300

1.000000

1.248700

1.481800

1.844300

1.951100

1.982300

1.904800

Con los siguientes datos el programa recurrió a early stopping.

Tomando en cuenta que [aprendiaje validacion pruebas]

1.[70% 15% 15%]

2.[80% 10% 10%]

Elija la opcion de distribución deseada: 2

Ingrese el vector de la arquitectura:

**[1 10 1]**

Tomando en cuenta que:

1. purelin(n)

2. logsig(n)

3. tansig(n)

Ingrese el vector de funciones de activación:

**[2 1]**

Dame el valor aceptable del error(error\_epoch\_train): .00000000000000001

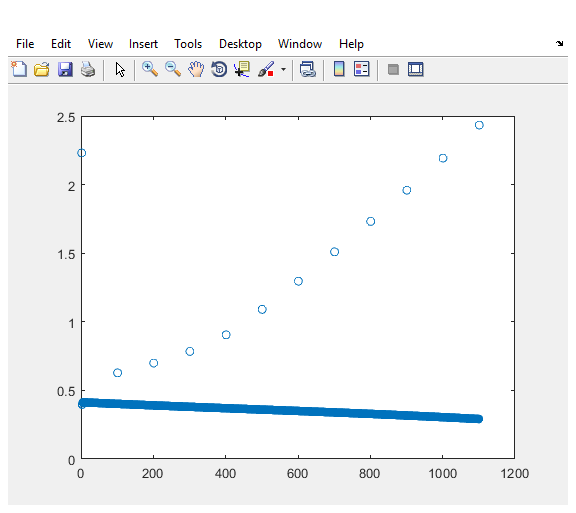
valor del factor de aprendizaje: .03

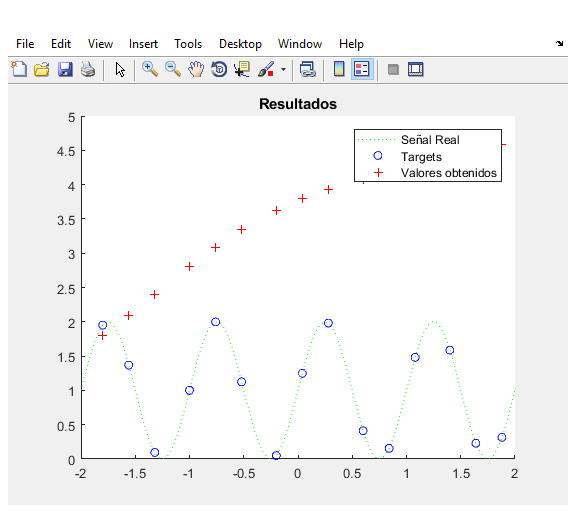
Número máximo de épocas: 7000

Épocas de validación: 100

Número máximo de incrementos consecutivos del error\_epoch\_validation: 10

Y se obtuvieron las siguientes graficas:





Al hacer varias pruebas notamos que necesitábamos agregar otra época.

Tomando en cuenta que [aprendiaje validacion pruebas]

1.[70% 15% 15%]

2.[80% 10% 10%]

Elija la opcion de distribución deseada: 1

Ingrese el vector de la arquitectura:

**[1 16 10 1]**

Tomando en cuenta que:

1. purelin(n)

2. logsig(n)

3. tansig(n)

Ingrese el vector de funciones de activación:

**[3 2 1]**

Dame el valor aceptable del error(error\_epoch\_train): .00000000000000001

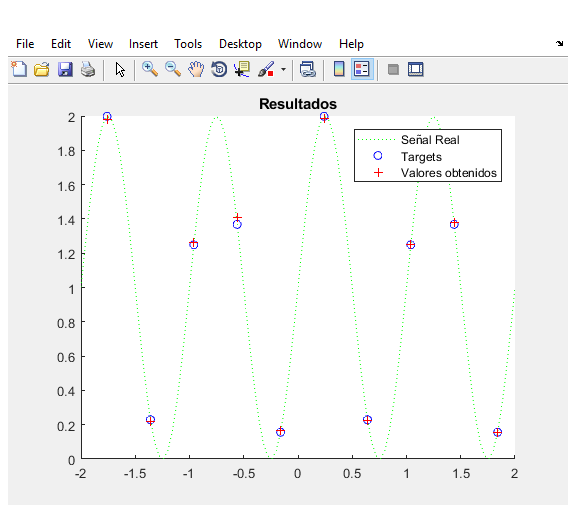
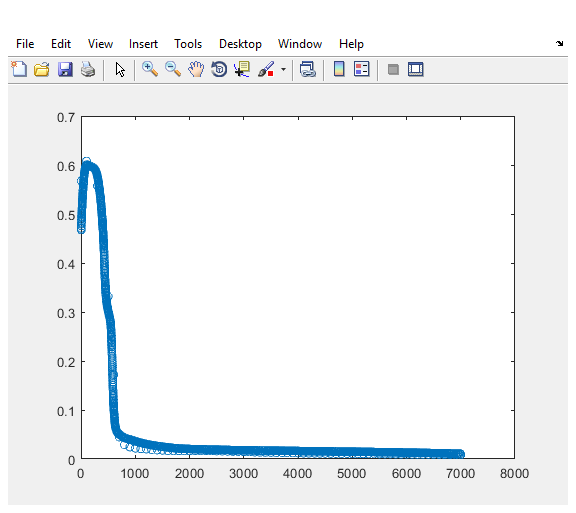
valor del factor de aprendizaje: .03

Número máximo de épocas: 7000

Épocas de validación: 100

Número máximo de incrementos consecutivos del error\_epoch\_validation: 10

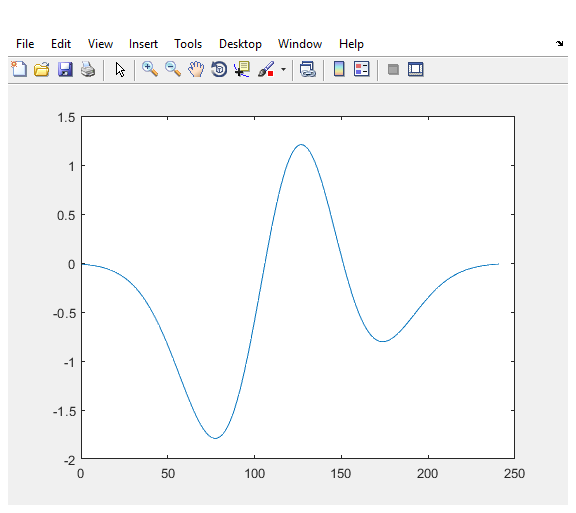
Se obtuvieron las siguientes graficas:

Error de época.

Se aproximó la siguiente señal:

**02\_Polinomio\_entrada**

**02\_Polinomio\_Target**



Para la siguiente señal se optó por ingresar arquitecturas similares a las de la señal 1.

Tomando en cuenta que [aprendiaje validacion pruebas]

1.[70% 15% 15%]

2.[80% 10% 10%]

Elija la opcion de distribución deseada: 1

Ingrese el vector de la arquitectura:

**[1 16 10 1]**

Tomando en cuenta que:

1. purelin(n)

2. logsig(n)

3. tansig(n)

Ingrese el vector de funciones de activación:

**[3 2 1]**

Dame el valor aceptable del error(error\_epoch\_train): .00000000000000001

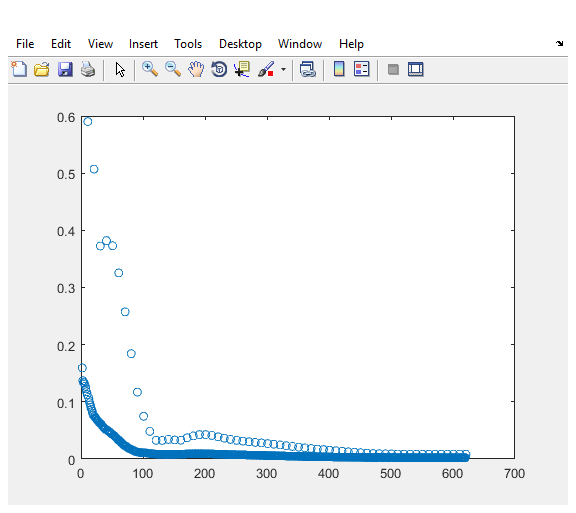
valor del factor de aprendizaje: .03

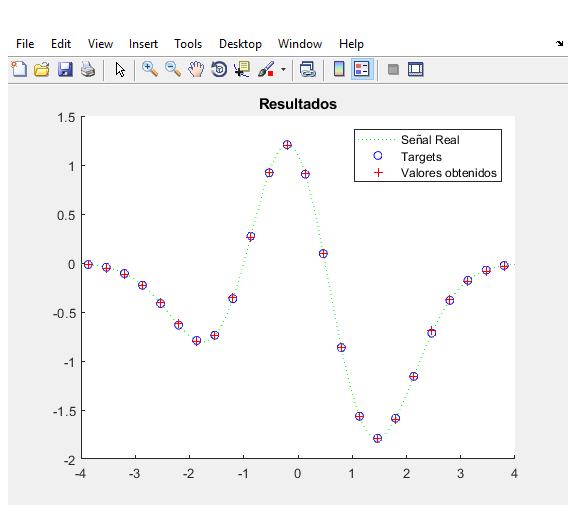
Número máximo de épocas: 7000

Épocas de validación: 10

Número máximo de incrementos consecutivos del error\_epoch\_validation: 7

Se obtuvieron las siguientes graficas:





En este caso el programa convergió ya que se alcanzó el numval máximo por early stopping, que es 7.