



PRÁCTICA 2

NEUROCOMPUTACIÓN

José Antonio Guerrero Avilés
Nº Lista: 14
cany@correo.ugr.es

INDICE

1.- Descripción de los ficheros.	2
2.- Obtención de la topología (Logaritmo).	3
2.1- Prueba 1	3
2.2- Prueba 2	3
2.3- Prueba 3	4
2.4- Prueba 4	4
2.5- Prueba 5	5
2.6- Prueba 6	5
2.7- Prueba 7	6
2.8- Conclusión	6
3.- Obtención del mejor tipo de entrenamiento (Logaritmo)	7
3.1 – Trainlm	7
3.2 – Traingdx	7
3.3 – Traincgp	8
3.4 – Trainbfg	8
3.5 – Trainrp	8
3.6 – Trainscg	9
3.7 – Conclusiones	9
4. – Pruebas con la mejor red (Logaritmo)	9
5.- Obtención de la topología (Intervalos).	11
5.1- Prueba 1	11
5.2- Prueba 2	11
5.3- Prueba 3	12
5.4- Prueba 4	12
5.5- Prueba 5	13
5.6- Prueba 6	13
5.7- Prueba 7	14
5.8- Conclusión	14
6.- Obtención del mejor tipo de entrenamiento (Intervalos)	15
6.1 – Trainlm	15
6.2 – Traingdx	15
6.3 – Traincgp	16
6.4 – Trainbfg	16
6.5 – Trainrp	16
6.6 – Trainscg	17
6.7 – Conclusiones	17
7. – Pruebas con la mejor red (Intervalos)	17

1.- Descripción de los ficheros.

- **Funcion2Entrena.txt:** Entradas y salidas de entrenamiento sin modificar.
- **Funcion2Test:** Entradas y salidas de test sin modificar.
- **EntrenaModIn.txt:** Entrada de entrenamiento modificada eliminando cabeceras y demás.
- **EntrenaModOut.txt:** Salida de entrenamiento modificada sin transformar, eliminando cabeceras y demás.
- **EntrenaModOutInt.txt:** Salida de entrenamiento modificada transformada por intervalos.
- **EntrenaModOutLog.txt:** Salida de entrenamiento modificada transformada por logaritmo.
- **TestModIn.txt:** Entrada de test modificada eliminando las cabeceras y demás.
- **TestModOut.txt:** Salida de test modificada sin transformar, eliminando cabeceras y demás.
- **TestModOutInt.txt:** Salida de test modificada transformada por intervalos.
- **TestModOutLog.txt:** Salida de test modificada transformada por logaritmo.
- **ResultadoInt.txt:** Resultados aplicando la transformación por intervalos que se muestran por pantalla con formato: Entrada – Salida Esperada – Salida - Error
- **ResultadoLog.txt:** Resultados aplicando la transformación por logaritmo que se muestran por pantalla con formato: Entrada – Salida Esperada – Salida - Error
- **ProcesarArchivosIntervalos.m:** Función que crea todos los archivos que tienen que ver con los intervalos, excepto ResultadosInt.txt.
- **ProcesarArchivosLogaritmo.m:** Función que crea todos los archivos referentes al logaritmo excepto ResultadosLog.txt.
- **RedIntervalos.m:** Programa donde se crea y entrena la red para procesar los archivos de intervalos y obtener todo lo que se nos pide. Además, llama a la función ProcesarArchivosIntervalos.m para crear los archivos que necesita.
- **RedLogaritmica.m:** Programa donde se crea y entrena la red para procesar los archivos del logaritmo y obtener todo lo que se nos pide. Además, llama a la función ProcesarArchivosLogaritmo.m para crear los archivos que necesita.
- **MedicionesPruebas.xls:** Archivo con las mediciones para obtener la mejor topología y el mejor tipo de entrenamiento.

2.- Obtención de la topología (Logaritmo).

Para obtener la topología más adecuada, voy a realizar una serie de pruebas aumentando y/o disminuyendo tanto el número de capas de la red, como el número de neuronas de dichas capas dependiendo de los resultados que vayamos obteniendo. Cada prueba constará de 10 ensayos y como resultado de la prueba, nos quedaremos con la media de los resultados obtenidos en los ensayos realizados. RedLogaritmica.m ya realiza 10 ejecuciones y calcula la media de esas ejecuciones.

2.1- Prueba 1

Los datos de la tipología que voy a usar para esta prueba son:

Nº Capas	1
Neuronas Capa	10

Con esta topología, he obtenido los siguientes datos:

	MSE
MEDIA	0,7387

A partir de estos datos obtenidos (voy a usar la media de los ensayos realizados) iré cambiando el número de neuronas de la capa y el número de capas para ver si mejoran o empeoran los resultados. Si mejoran, me quedaré con la tipología que mejore estos resultados, si no, la desecharé.

2.2- Prueba 2

Para esta prueba, voy a probar, por ejemplo, a disminuir el número de neuronas de la capa y ver cuáles son los resultados.

Nº Capas	1
Neuronas Capa	5

Con esta topología he obtenido los datos que aparecen en la siguiente tabla:

	MSE
MEDIA	0,5251

Comparando las medias obtenidas en esta prueba (prueba 2) con las obtenidas en la prueba 1, se ve que teniendo una sola capa, si reduzco el número de neuronas de la capa, los resultados mejora, por tanto, de momento, me quedo con la topología de la prueba 2.

2.3- Prueba 3

Para esta prueba, voy a probar a disminuir el número de neuronas respecto de la prueba anterior, ya que como hemos visto, disminuir el número de neuronas de la capa, mejoraba los resultados. Los datos para esta prueba son:

Nº Capas	1
Neuronas Capa	2

Y los resultados que obtengo son los siguientes:

	MSE
MEDIA	0,8934

Comparando con los resultados obtenidos en la segunda prueba (que es la mejor hasta ahora), se observa que disminuyendo el número de neuronas de la capa, no se mejoran los resultados, por lo tanto, la topología con la que nos quedamos momentáneamente es la de la segunda prueba.

Como continuar disminuyendo el número de neuronas de la capa, no mejorará los resultados, lo que voy a probar ahora es a aumentar el número de neuronas de la capa.

2.4- Prueba 4

Los datos para esta prueba serán:

Nº Capas	1
Neuronas Capa	15

Y estos son los resultados que he obtenido:

	MSE
MEDIA	0,5635

Aumentar el número de neuronas tampoco mejora los resultados que hasta ahora tengo, por lo que ahora, voy a probar a añadir una capa más (ahora tendré 2 capas) e ir aumentando y/o disminuyendo el número de neuronas de las capas para ver si los resultados que voy obteniendo mejoran mi mejor topología hasta el momento.

2.5- Prueba 5

Para esta prueba voy a usar los siguientes datos:

Nº Capas	2
Neuronas Capa 1	5
Neuronas Capa 2	10

He decidido empezar con estos datos, porque 5 es el número de neuronas con el que mejor resultado me daba en la primera capa.

Los datos obtenidos en esta prueba son los siguientes:

	MSE
MEDIA	0,8203

Comparando con la topología de la prueba 2, que era la mejor hasta el momento, se observa que añadiendo 2 capas con 5 y 10 neuronas respectivamente, no se mejoran los resultados; por tanto, mantenemos la anterior topología.

Aunque añadiendo estas 2 capas no se mejoran los resultados obtenidos, voy a probar a aumentar y/o disminuir el número de neuronas de ambas capas para ver si alguna posible combinación puede mejorar los resultados de la primera prueba.

2.6- Prueba 6

Voy a aumentar el número de neuronas de la primera capa:

Nº Capas	2
Neuronas Capa 1	5
Neuronas Capa 2	15

Viendo los resultados de la tabla que pongo a continuación, se puede ver como tampoco se mejoran los resultados obtenidos por la mejor topología hasta el momento:

	MSE
MEDIA	0,951

No obstante, voy a probar a disminuir el número de neuronas de la segunda capa hasta 5 para ver qué sucede.

2.7- Prueba 7

Voy a mantener las neuronas de la capa 1 y a aumentar las de la capa 2:

Nº Capas	2
Neuronas Capa 1	5
Neuronas Capa 2	5

Los resultados que he obtenido son los siguientes:

	MSE
MEDIA	2,6782

Y como se puede ver, tampoco mejora los resultados de la topología que ya tenía.

2.8- Conclusión

Viendo los resultados de las pruebas, se puede determinar que la mejor topología para la red es:

Nº Capas	1
Neuronas Capa	5

Obteniendo como resultado:

Nº Ensayo	MSE
MEDIA (10 ensayos)	0,5251

3.- Obtención del mejor tipo de entrenamiento (Logaritmo)

Con la mejor topología obtenida en las pruebas anteriores, voy a realizar ahora una batería de pruebas, cambiando el tipo de entrenamiento en cada una de las pruebas para ver cuál es el entrenamiento con el que la mejor topología obtiene los mejores resultados.

La topología será:

Nº Capas	1
Neuronas Capa	5

3.1 – Trainlm

Tipo de entrenamiento	trainlm
-----------------------	---------

Este entrenamiento es: Levenberg-Marquardt backpropagation

Los resultados que he obtenido son (los mismos que en la obtención de la topología ya que usé este entrenamiento como entrenamiento base para todas las pruebas anteriores):

	MSE
MEDIA	0,5251

3.2 – Traingdx

Tipo de entrenamiento	traingdx
-----------------------	----------

Este entrenamiento es: Gradient descent with momentum and adaptive learning rate backpropagation.

Los resultados con este tipo de entrenamiento son peores, como se puede observar en la siguiente tabla:

	MSE
MEDIA	5,7811

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

3.3 – *Traincgp*

Tipo de entrenamiento	traincgp
-----------------------	----------

Este tipo de entrenamiento es: Conjugate gradient backpropagation with Polak-Ribière updates (restarts).

Los resultados que obtengo, y que una vez más no mejoran a los obtenidos con trainlm son:

	MSE
MEDIA	0,1,5326

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

3.4 – *Trainbfg*

Tipo de entrenamiento	trainbfg
-----------------------	----------

Este tipo de entrenamiento es: BFGS quasi-Newton backpropagation.

Con él, tampoco se mejoran los resultados obtenidos hasta el momento con trainlm:

	MSE
MEDIA	0,9019

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

3.5 – *Trainrp*

Tipo de entrenamiento	trainrp
-----------------------	---------

Este tipo de entrenamiento es: Resilient backpropagation.

Los resultados obtenidos con este tipo de entrenamiento son:

	MSE
MEDIA	1,6541

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

3.6 – Trainscg

Tipo Entrenamiento	trainscg
--------------------	----------

Este tipo de entrenamiento es: Scaled conjugate gradient backpropagation

Los resultados que he obtenido usando este tipo de entrenamiento son:

	MSE
MEDIA	1,9166

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

3.7 – Conclusiones

Como se puede ver en los resultados que he obtenido, el tipo de entrenamiento con el que obtengo mejores resultados es el entrenamiento **Levenberg-Marquardt** (“trainlm”):

Tipo de entrenamiento	trainlm
MEDIA (MSE)	0,5251

4. – Pruebas con la mejor red (Logaritmo)

Después de realizar todas las pruebas anteriores, he obtenido la siguiente red como la mejor:

Nº Capas	1
Neuronas Capa	5
Tipo Entrenamiento	trainlm

Con esta red voy a realizar una batería de 10 pruebas: mi programa RedLogaritmica.m incluye un bucle de 10 ensayos, por lo que el resultado que te da tras ejecutarlo es la media de los 10 ensayos; pero además, voy a ejecutar RedLogaritmica.m 10 veces, por lo que realmente habré realizado 100 ejecuciones. De esta manera, obtendré unos datos más fiables ya que el número de ejecuciones será bastante alto.

Los resultados que he obtenido al realizar estas pruebas son los que aparecen en la siguiente tabla:

Nº Ejecución	MSE
1	0,5251
2	0,4269
3	0,4789
4	0,0117
5	0,7107
6	0,4397
7	0,4957
8	0,4613
9	0,5493
10	0,7841
MEDIA	0,48834

5.- Obtención de la topología (Intervalos).

Para obtener la topología más adecuada, voy a realizar una serie de pruebas aumentando y/o disminuyendo tanto el número de capas de la red, como el número de neuronas de dichas capas dependiendo de los resultados que vayamos obteniendo. Cada prueba constará de 10 ensayos y como resultado de de la prueba, nos quedaremos con la media de los resultados obtenidos en los ensayaos realizados. Mi programa ya realiza 10 ejecuciones (incluye un blucle), así que el resultado del error tras ejecutar RedIntervalos.m es la media de esas 10 ejecuciones, y es lo que voy a usar para comparar los resultados.

5.1- Prueba 1

Los datos de la tipología que voy a usar para esta prueba son:

Nº Capas	1
Neuronas Capa	10

Con esta topología, he obtenido los siguientes datos:

	MSE
MEDIA	0,1809

A partir de estos datos obtenidos (voy a usar la media de los ensayos realizados) iré cambiando el número de neuronas de la capa y el número de capas para ver si mejoran o empeoran los resultados. Si mejoran, me quedaré con la tipología que mejore estos resultados, si no, la desecharé.

5.2- Prueba 2

Para esta prueba, voy a probar, por ejemplo, a disminuir el número de neuronas de la capa y ver cuáles son los resultados.

Nº Capas	1
Neuronas Capa	5

Con esta topología he obtenido los datos que aparecen en la siguiente tabla:

	MSE
MEDIA	0,1329

Los resultados mejoran a los anteriores, así que esta será mi nueva mejor topología.

5.3- Prueba 3

Para esta prueba, voy a continuar disminuyendo el número de neuronas de la capa, ya que he visto que al reducirlas, mejoraban los resultados obtenidos. Los datos para esta prueba son:

Nº Capas	1
Neuronas Capa	2

Y los resultados que obtengo son los siguientes:

	MSE
MEDIA	0,2886

Comparando con los resultados obtenidos en la segunda prueba (que es la mejor hasta ahora), se observa que seguir disminuyendo el número de neuronas de la capa, ya no mejora los resultados, por lo tanto, la topología con la que nos quedamos momentáneamente es la de la segunda prueba.

Como continuar disminuyendo el número de neuronas de la capa, no mejorará los resultados, lo que voy a probar ahora es a aumentar el número de neuronas de la capa a ver qué sucede.

5.4- Prueba 4

Los datos para esta prueba serán:

Nº Capas	1
Neuronas Capa	15

Y los datos obtenidos, que no mejoran a los que ya tenía son:

	MSE
MEDIA	0,1435

Tras realizar estas pruebas, voy a probar a añadir una capa más (ahora tendré 2 capas) e ir aumentando y/o disminuyendo el número de neuronas de las capas para ver si los resultados que voy obteniendo mejoran mi mejor topología hasta el momento.

5.5- Prueba 5

Para esta prueba voy a usar los siguientes datos:

Nº Capas	2
Neuronas Capa 1	5
Neuronas Capa 2	10

He decidido empezar con estos datos, porque 5 es el número de neuronas con el que obtenía mejor resultado con una capa.

Los datos obtenidos en esta prueba son los siguientes:

	MSE
MEDIA	0,4482

Comparando con la topología de la prueba 2, que era la mejor hasta el momento, se observa que añadiendo 2 capas con 5 y 10 neuronas respectivamente, no se mejoran los resultados, por tanto, mantenemos la anterior topología.

Aunque añadiendo estas 2 capas no se mejoran los resultados obtenidos, voy a probar a aumentar y/o disminuir el número de neuronas para ver si alguna posible combinación puede mejorar los resultados de la primera prueba.

5.6- Prueba 6

Voy a disminuir el número de neuronas de la primera capa:

Nº Capas	2
Neuronas Capa 1	5
Neuronas Capa 2	15

Viendo los resultados de la tabla que pongo a continuación, se puede ver como tampoco mejora los resultados que ya tenía:

	MSE
MEDIA	0,2381

No obstante, voy a probar a aumentar y disminuir el número de neuronas de la segunda capa y a mantener a 5 el número de neuronas de la primera para ver lo que sucede.

5.7- Prueba 7

Los datos de la prueba son:

Nº Capas	2
Neuronas Capa 1	5
Neuronas Capa 2	5

Y los resultados obtenidos:

	MSE
MEDIA	0,2257

Como se ve, tampoco mejoran los resultados de la topología de la prueba 2.

5.8- Conclusión

Viendo los resultados de las pruebas, se puede determinar que la mejor topología para la red es:

Nº Capas	1
Neuronas Capa 1	5

Obteniendo como resultado:

Nº Ensayo	MSE
MEDIA (10 ensayos)	0,1329

6.- Obtención del mejor tipo de entrenamiento (Intervalos)

Con la mejor topología obtenida en las pruebas anteriores, voy a realizar ahora una batería de pruebas, cambiando el tipo de entrenamiento en cada una de las pruebas para ver cuál es el entrenamiento con el que la mejor topología obtiene los mejores resultados.

La topología será:

Nº Capas	1
Neuronas Capa	5

6.1 – Trainlm

Tipo de entrenamiento	trainlm
-----------------------	---------

Este entrenamiento es: Levenberg-Marquardt backpropagation

Los resultados que he obtenido son (los mismos que en la obtención de la topología ya que usé este entrenamiento como entrenamiento base para todas las pruebas anteriores):

	MSE
MEDIA	0,1329

6.2 – Traingdx

Tipo de entrenamiento	traingdx
-----------------------	----------

Este entrenamiento es: Gradient descent with momentum and adaptive learning rate backpropagation.

Los resultados con este tipo de entrenamiento son peores, como se puede observar en la siguiente tabla:

	MSE
MEDIA	1,3271

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

6.3 – *Traincgp*

Tipo de entrenamiento	traincgp
-----------------------	----------

Este tipo de entrenamiento es: Conjugate gradient backpropagation with Polak-Ribière updates (restarts).

Los resultados que obtengo, y que una vez más no mejoran a los obtenidos con trainlm son:

	MSE
MEDIA	0,5103

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

6.4 – *Trainbfg*

Tipo de entrenamiento	trainbfg
-----------------------	----------

Este tipo de entrenamiento es: BFGS quasi-Newton backpropagation.

Con él, tampoco se mejoran los resultados obtenidos hasta el momento con trainlm:

	MSE
MEDIA	0,3329

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

6.5 – *Trainrp*

Tipo de entrenamiento	trainrp
-----------------------	---------

Este tipo de entrenamiento es: Resilient backpropagation.

Los resultados obtenidos con este tipo de entrenamiento son:

	MSE
MEDIA	0,414

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

6.6 – Trainscg

Tipo Entrenamiento	trainscg
--------------------	----------

Este tipo de entrenamiento es: Scaled conjugate gradient backpropagation

Los resultados que he obtenido usando este tipo de entrenamiento son:

	MSE
MEDIA	0,4616

Como se puede ver, no mejoran a los resultados obtenidos con el tipo de entrenamiento “trainlm”.

6.7 – Conclusiones

Como se puede ver en los resultados que he obtenido, el tipo de entrenamiento con el que obtengo mejores resultados es el entrenamiento **Levenberg-Marquardt** (“trainlm”):

Tipo de entrenamiento	trainlm
MEDIA (MSE)	0,1329

7. – Pruebas con la mejor red (Intervalos)

Después de realizar todas las pruebas anteriores, he obtenido la siguiente red como la mejor:

Nº Capas	1
Neuronas Capa	5
Tipo Entrenamiento	trainlm

Con esta red voy a realizar una batería de 10 pruebas: mi programa RedIntervalos.m incluye un bucle de 10 ensayos, por lo que el resultado que te da tras ejecutarlo es la media de los 10 ensayos; pero además, voy a ejecutar RedIntervalos.m 10 veces, por lo que realmente habré realizado 100 ejecuciones, de esta manera, obtendré unos datos más fiables ya que el número de ejecuciones será bastante alto.

Los resultados que he obtenido al realizar estas pruebas son los que aparecen en la siguiente tabla:

Nº Ejecución	MSE
1	0,1329
2	0,1287
3	0,1615
4	0,1535
5	0,1329
6	0,1394
7	0,1613
8	0,1287
9	0,138
10	0,1366
MEDIA	0,14135

Aunque la media empeora un poco (de 0,1329 a 0,14135), tras haber realizado las 100 ejecuciones, sigue siendo obteniendo mejores resultados que el resto de topologías y entrenamientos usados en las pruebas.