Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

INDICE

Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

OBJETIVO

El objetivo del presente Trabajo Práctico Especial es implementar una red neuronal multicapa, con aprendizaje supervisado, con la cual se resuelva el problema asignado. Para esto, se utilizaron como lenguajes de programación Octave y/o Matlab.

1. DESCRIPCIÓN del TRABAJO REALIZADO

Algoritmo backpropagation

Para poder cumplir con el objetivo propuesto en el informe, se implementó el algoritmo backpropagation, que se basa en la propagación del error delta de la capa de salida a las capas ocultas (de forma proporcional a los pesos considerados en cada capa).

La condición de corte utilizada en el algoritmo, es que el error cuadrático medio de aplicar el conjunto de entrenamiento a la red, sea menor a un epsilon dado, o que se alcance una cantidad de épocas (ambos valores se pasan como parámetros al algoritmo).

Cabe aclarar que se efectuó un aprendizaje supervisado, es decir, se hizo sobre la base de comparación directa de la salida de la red con respuestas correctas conocidas.

Dicho algoritmo se encuentra en la función multiLayerPerceptron.m

Aprendizaje de la función dada

Para poder tener mejor noción acerca del conjunto de datos asignado (almacenado en el archivo “samples2.txt”), se decidió generar un gráfico (**Figura 1 del Anexo**). De él, se puede inferir que la función es periódica, y que su máximo y su mínimo son 1.02... y -1.022 respectivamente. Con todo esto se pensó en tomar un conjunto de entrenamiento de 100 patrones (procurando que éstos representen el comportamiento de la función), y dejar el resto de los datos (que son 341) como conjunto de testeo (para evaluar en qué porcentaje la red infiere bien la función, y así determinar su capacidad de aprendizaje).

Con respecto a las funciones de activación g usadas en cada capa de la red, vale mencionar que en las capas ocultas se utilizó una función no lineal (que podía ser exponencial o tangente hiperbólica), y en la capa de salida una función lineal (para poder adecuar la salida al rango de la función del problema).

Para automatizar el proceso de obtención de resultados utilizando diferentes arquitecturas, se implementó la

Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

función main.m .

Mejoras implementadas al algoritmo

Dado que el algoritmo BackPropagation es muy lento a medida que aumenta la cantidad de patrones a entrenar, existen variantes que intentan hacer a dicho algoritmo más rápido. Entre todas ellas, se decidió implementar las siguientes:

➔ *Momentum*

Este concepto se basa en agregar un término que pese el descenso promedio, que matemáticamente hablando se expresa como:

$$\Delta wpq(t+1) = -\eta \frac{\partial E}{\partial wpq} + \alpha \Delta wpq(t) \ , \ 0 < \alpha < 1$$

Esta mejora se activa en el algoritmo backPropagation haciendo que la variable useMomentum tenga como valor “true” en la función multiLayerPerceptron.m .

➔ *Factor de aprendizaje (η) adaptativo*

Dado que a medida que el sistema va aprendiendo el problema, el error de los pesos debería ser cada vez menor, podría ser conveniente hacer que el factor de aprendizaje se vaya adaptando en base al comportamiento de la función de costo (ΔE) por cada k pasos: si ha decrementado, el η se incrementa una cantidad α; si ha incrementado, el η se decrementa un porcentaje β; y en el caso de que haya incrementado o decrementado alternadamente, no se variaría al η. Expresando esto matemáticamente sería:

$\Delta\eta = +\alpha$, si $\Delta E < 0$ consistentemente en k pasos,

$\Delta\eta = -\beta\eta$, si $\Delta E \geq 0$ consistentemente en k pasos, o sino

$\Delta\eta = 0$.

Esta mejora se activa en el algoritmo backPropagation, haciendo que la variable useAdaptative tenga como valor “true” en la función multiLayerPerceptron.m .

Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

2. ANÁLISIS de los RESULTADOS OBTENIDOS. CONCLUSIONES

Para realizar las mediciones, se tomaron en cuenta diferentes arquitecturas (de 1 a 10 capas).

En base a ellas, se pueden sacar las siguientes conclusiones:

1) Como se puede ver tanto en la **Tabla 1** como en la **Tabla 2**, utilizar la función tangente hiperbólica como función de activación en las capas ocultas da mejores resultados que con la función exponencial. Esto puede deberse al hecho de que la función tangente hiperbólica tiene como rango de salida un intervalo bastante parecido al de la función del problema, a diferencia de la exponencial, cuyo rango de salida es (0,1), lo que puede generar que la transformación lineal aplicada en la capa de salida no sea muy efectiva (la normalización de los datos no ayudaría a representar bien las salidas deseadas).

2) Arquitecturas con mayor cantidad de capas ocultas (y de neuronas en cada capa) tienen mejor capacidad de generalización (de aprendizaje). Esto se debe a que el hecho de poner más capas ocultas (y con una cantidad considerable de neuronas) permite que la representación interna de cada patrón sea adecuada (es decir, que si dos entradas de la red tienen una misma representación interna en la misma capa, tendrán como salida el mismo valor), tarea laboriosa de efectuar cuando la salida tiene valores de hasta cuatro decimales.

3) Los datos de la **Tabla 3** evidencian lo sospechado: que aplicar momentum y un η adaptativo mejoran el aprendizaje de la red, aunque no de manera tan notable. Esto puede deberse a que estas mejoras impactan en la rapidez con la que se aprende, y que los parámetros α, k y β elegidos no son lo suficientemente idóneos para converger en la cantidad de épocas elegidas.

Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

ANEXO

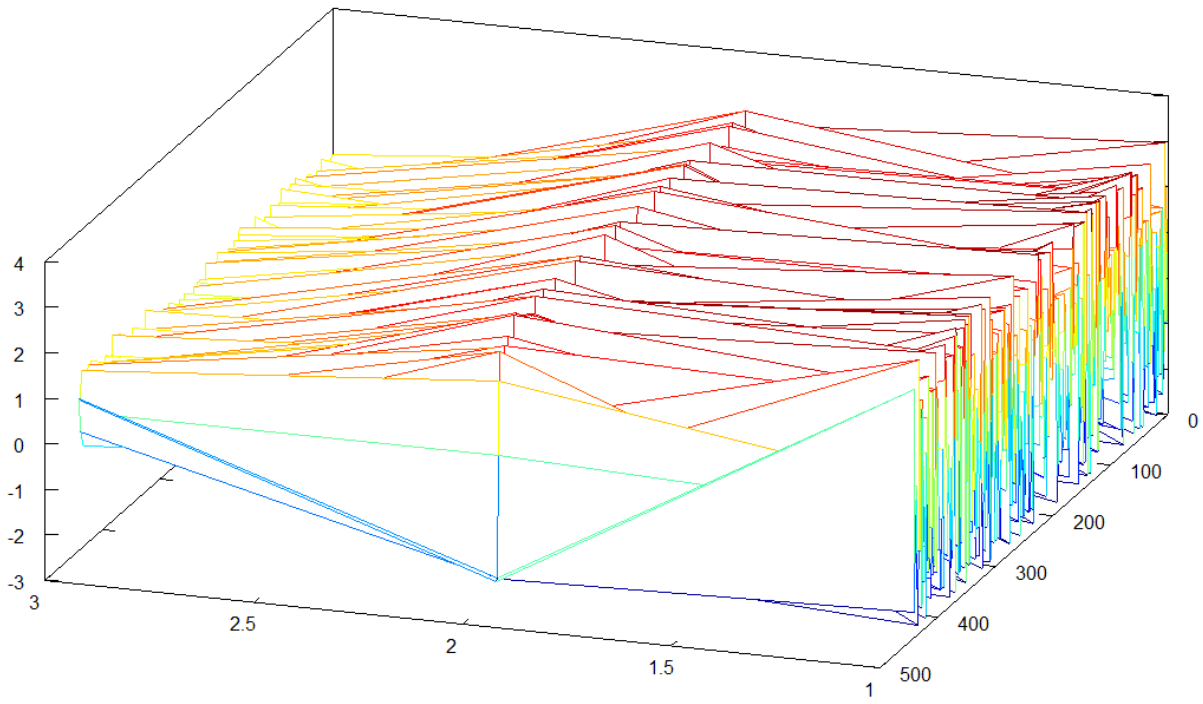

Figura 1

Gráfico aproximado de la función a inferir (en base al conjunto de datos dado)

Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

Arquitectura	Porcentaje aprendido	Error cuadrático medio	Épocas
[5]	0.53372	0.0048610	5000
[7]	0.53079	9.8994e-04	3628
[8]	0.47507	0.0040970	5000
[10]	0.53372	9.4644e-04	2310
[4 2]	0.52199	0.0043777	5000
[5 2]	0.52493	0.0013340	5000

Tabla 1
Resultados de probar redes de distintas arquitecturas con $g = exp,\ \eta = 0.2\ y\ epsilon = 0.001$

Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

Arquitectura	Porcentaje aprendido	Error cuadrático medio	Épocas
[5]	0.5161	0.0406	5000
[10]	0.48	0.0068	5000
[8]	0.48	0.0038	5000
[7]	0.53	0.00041	5000
[5 2]	0.47	0.010	5000
[4 2]	0.5279	0.0144	10000

Tabla 2 Resultados de probar redes de distintas arquitecturas con $g = tanh,\ \eta = 0.2\ y\ epsilon = 0.001$

Arquitectura	Porcentaje aprendido	Error cuadrático medio	Épocas
[4 2]	0.5279	0.0163	5000
[3 2]	0.5249	0.0340	10000
[3 3]	0.5279	0.0384	10000
[2 2 2]	0.5191	0.3711	10000

Tabla 3 Resultados de probar redes de distintas arquitecturas con las mejoras y con $g = tanh,\ \eta = 0.2\ y\ epsilon = 0.001$

Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

Perceptrones: 5 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0406 memorización: .5 aprendizaje: .5161 Modificado: no

Perceptrones: 4 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.1099 memorización: .48 aprendizaje: .4780 Modificado: Momentum

Perceptrones: 6 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0132 memorización: .51 aprendizaje: .5191 Modificado: Momentum

Perceptrones: 10 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0068 memorización: .48 aprendizaje: .475 Modificado: No

Perceptrones: 8 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0038 memorización: .49 aprendizaje: .48 Modificado: No

Perceptrones: 7 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.00041 memorización: .51 aprendizaje: .53 Modificado: No

Perceptrones: 5 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.010 memorización: .48 aprendizaje: .47 Modificado: No

Perceptrones: 4 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0163 memorización: .51 aprendizaje: .5279 Modificado: Momentum

Perceptrones: 4 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0317 memorización: .49 aprendizaje: .5073 Modificado: Momentum

Perceptrones: 3 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0175 memorización: .50 aprendizaje: .5044 Modificado: Momentum

Perceptrones: 3 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0339 memorización: .49 aprendizaje: .4809 Modificado: Momentum

Perceptrones: 3 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 10000 Em: 0.0340 memorización: .50

Grupo 1:
DOMINGUES, Matias (Leg. 50278)
FONTANELLA DE SANTIS, Teresa (Leg. 52455)
MARTINEZ CORREA, Facundo (Leg. 49139)

aprendizaje: .5249 Modificado: Momentum

Perceptrones: 3 3 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 10000 Em: 0.0384 memorización: .51 aprendizaje: .5279 Modificado: Momentum

Perceptrones: 2 2 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 10000 Em: 0.3711 memorización: .50 aprendizaje: .5191 Modificado: Momentum

Perceptrones: 4 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 10000 Em: 0.0144 memorización: .51 aprendizaje: .5279 Modificado: No

Perceptrones: 4 2 Patrones de prueba: 100 n: .2 epsilon: .001 épocas: 5000 Em: 0.0345 memorización: .51 aprendizaje: .5279 Modificado: Momentum

Perceptrones: 4 2 Patrones de prueba: 100 n: .1 epsilon: .001 épocas: 5000 Em: 0.0087 memorización: .49 aprendizaje: .5103 Modificado: Momentum