Aproximación eficiente de una función afín definida sobre un politopo, por medio de redes neuronales

Llanas Juárez, Bernardo.(ma07@caminos.upm.es)
Sáinz Alonso, Francisco Javier.(fjavier.sainz@upm.es)
Departamento de Matemáticas e Informática aplicadas a la Ingeniería Civil
Universidad Politécnica de Madrid

RESUMEN

En este artículo se da una demostración del hecho de que una función afín, definida en un politopo de $\mathbb{R}^d$, puede aproximarse uniformemente por una red neuronal monocapa (SLFN) con dos neuronas en la capa oculta y función de activación logística. La demostración es constructiva y permite obtener los pesos y biases de la red como funciones de los coeficientes de la función afín y del error uniforme. La red resultante ofrece una mayor precisión y estabilidad que las obtenidas previamente.

***Palabras clave*:**
Redes neuronales, Teoría de la aproximación.

1. INTRODUCCIÓN

Attali and G. Pagés [1] probaron el siguiente:

Teorema: *Un polinomio de grado 1 y d variables, puede ser aproximado uniformemente sobre subconjuntos compactos de* $\mathbb{R}^d$ *por una SLFN con d+1 neuronas, suponiendo que la función de activación es diferenciable y que su valor y el de su primera derivada es diferente de cero en el origen.*

Aunque la demostración es semiconstructiva, Attali y Pagés no proporcionan una expresión explícita de los pesos de la red en función del error uniforme.

En [2], Malakooti y Zhou consideran SLFNs con función de activación logística y demuestran un teorema similar al anterior, pero reduciendo a 2 el número de neuronas necesarias y considerando el caso de dominios de la forma: $\{\mathbf{x} \in \mathbb{R}^d \ / \ |\mathbf{a}.\mathbf{x}+a_0| < M\}$, donde $f(x)= \mathbf{a}.\mathbf{x}+a_0$ es la función afín a aproximar y M es un número positivo arbitrario.

La demostración es constructiva y los autores presentan un algoritmo para obtener los pesos de la red, que proporciona valores muy pequeños para los pesos interiores y valores enormes para los pesos exteriores, incluso en el caso de errores uniformes relativamente grandes. De hecho, los pesos internos disminuyen como $1/ M^2$. Esto implica falta de precisión e inestabilidad en la red resultante cuando se desean obtener buenos aproximantes en regiones arbitrarias o cuando los coeficientes de la función afín se conocen con un cierto margen de error [3].

En este artículo extendemos el resultado de Malakooti y Zhou a regiones en forma de politopo arbitrarias. El algoritmo resultante permite obtener unos pesos internos que disminuyen como 1/m siendo m el valor mínimo de $|\mathbf{u}.\mathbf{x}|$ sobre los puntos del politopo y **u** el cociente de **a** por su modulo. Como en general m y M son de igual orden de magnitud, el algoritmo propuesto proporciona una red mas exacta y estable para valores de la precisión mayores que los considerados por el algoritmo de Malakooti y Zhou.

.

2. RESULTADOS

Se definen algunos conceptos previos necesarios para el resto de la explicación.

Definición 1. Sea σ una función sigmoidal acotada, esto es, $\lim_{t\to-\infty}\sigma(t)=0$ y $\lim_{t\to\infty}\sigma(t)=1$.Una red neuronal "feedforward" de una capa oculta (SLFN) es un función dada por

$$\sum_{k=1}^{n} c_k\sigma(\mathbf{w}_k.\mathbf{x}+b_k), \tag{1}$$

donde $\mathbf{x}\in\mathbb{R}^d$, $\mathbf{w}_k\in\mathbb{R}^d$, y $c_k, b_k\in\mathbb{R}$, $k=1,\ldots,n.$ Llamamos a $\mathbf{w}_k, c_k$ pesos internos y externos respectivamente. Los parámetros b_k se llaman sesgos. Cada sumando en (1) se llama neurona. Una red neuronal "feedforward" con varias capas (MLFN), se puede definir inductivamente usando (1).

Definición 2. Sea $\{\mathbf{v_0},\mathbf{v_1},\ldots,\mathbf{v_d}\}$ un conjunto de $d+1$ vectores en $\mathbb{R}^d$.Este conjunto se llama independiente afín si los vectores $\{\mathbf{v_1}-\mathbf{v_0},\mathbf{v_2}-\mathbf{v_0},\ldots,\mathbf{v_d}-\mathbf{v_0}\}$ son linealmente independientes. Supongamos ahora que $\{\mathbf{v_0},\mathbf{v_1},\ldots,\mathbf{v_d}\}$ es un subconjunto afínmente independiente de $\mathbb{R}^d$. El *d*-simplex T generado por $\{\mathbf{v_0},\mathbf{v_1},\ldots,\mathbf{v_d}\}$ se denota por $<\mathbf{v_0},\mathbf{v_1},\ldots,\mathbf{v_d}>$, y es definido por el "convex hull" de los vectores $\mathbf{v_0},\mathbf{v_1},\ldots,\mathbf{v_d}$. Se denota por $|T|$ el diámetro de un *d*-simplex *T*.

A continuación demostramos que una función afín definida sobre un *d*-simplex en $\mathbb{R}^d$ puede ser aproximada uniformemente por una red neuronal de una capa teniendo solo dos neuronas y cuyos pesos se pueden expresar de forma analítica cerrada. Para simplificar la explicación, asumiremos que la función de activación es la función logística.

Si usamos la fórmula de Lagrange del resto en el desarrollo limitado de Taylor, en el método para aproximar polinomios por redes neuronales dado en [1], tenemos

Proposición 1 *Sea* $p(\mathrm{x})=A_0+A_1x$ *un polinomio de grado 1, en una variable y con coeficientes reales. Dado* $\varepsilon>0$, *existe una SLFN con dos neuronas*

$$N(x)=\sum_{j=0}^{1}\lambda_j s(\alpha_j x),$$

tal que

$$|N(x) - p(x)| < \varepsilon, \quad x \in [0,1],$$

donde $s(x) = 1/(1+e^{-x})$ *(función logística) y* $\lambda_j, \alpha_j, \quad j = 0,1,$ *son funciones de* A_0, A_1 *y* ε *que pueden ser expresadas en una forma analítica cerrada.*

Demostración. Sea $\alpha > 0.$ Aplicando el teorema de Taylor con el resto de Lagrange a la función $s(\alpha x),$ podemos escribir

$$s(\alpha x) = \frac{1}{2} + \frac{\alpha x}{4} + \frac{\alpha^2 x^2}{2} s''(z), \quad z \in (0, \alpha x).$$

Sean $\alpha_0, \alpha_1, \lambda_0, \lambda_1$ variables positivas. Si reemplazamos sucesivamente en la ecuación anterior, α por α_0 y α_1, multiplicamos la ecuación resultante por λ_0 y λ_1 respectivamente, y las sumamos, tenemos

$$r \equiv \left| \sum_{j=0}^{1} \lambda_j s(\alpha_j x) - \frac{1}{2} \sum_{j=0}^{1} \lambda_j - \frac{x}{4} \sum_{j=0}^{1} \lambda_j \alpha_j \right| = \frac{x^2}{2} \left| \sum_{j=0}^{1} \lambda_j \alpha_j^2 s''(z_j) \right|,$$

donde $z_j \in (0, \alpha_j x), \quad j = 0,1.$ Dado que $s'' = 2s^3 - 3s^2 + s$ y $s(z) \in (0,1)$ se sigue que $\left| s''(z) \right| < 1/10$ para todo $z \in \mathbb{R}$ .Por ello, puesto que $x \in [0,1],$ tenemos

$$r < \frac{1}{20} \left| \sum_{j=0}^{1} \lambda_j \alpha_j^2 \right|. \tag{2}$$

Si consideramos el sistema

$$\sum_{j=0}^{1} \lambda_j = 2A_0,$$

$$\sum_{j=0}^{1} \lambda_j \alpha_j = 4A_1,$$

y hacemos

$$\alpha_j = c_j h, \tag{3}$$

donde $c_j > 0, \quad j = 0,1,$ y $h > 0,$ tenemos

$$\lambda_0 = (2A_0 c_1 h - 4A_1)/(c_1 - c_0)h, \tag{4}$$

$$\lambda_1 = (4A_1 - 2A_0 c_0 h)/(c_1 - c_0)h, \tag{5}$$

De (2-5), se sigue

$$r < h(|2A_0 c_1 h - 4A_1| c_0^2 + |4A_1 - 2A_0 c_0 h| c_1^2 / (20 |c_1 - c_0|)). \tag{6}$$

De (6), la igualdad

$$\max_{|h|\le 1} |\gamma h+\delta| = max(|\gamma+\delta|,|\gamma-\delta|), \quad \gamma,\delta \in \mathbb{R}$$

y definiendo

$$M1 \equiv c_0^2 max(|2A_0c_1-4A_1|,|2A_0c_1+4A_1|),$$

$$M2 \equiv c_1^2 max(|4A_1-4A_0c_0|,|4A_1+2A_0c_0|),$$

$$M \equiv (M1+M2)/(20|c_1-c_0|),$$

se sigue que si $h \le 1$, entonces $r < Mh$. Por ello, dado $\varepsilon > 0$, es suficiente hacer $h=\min(1,\varepsilon/M)$. Los pesos de la red están dados por (3), (4) y (5).

$\square$

Comentario 1 La proposición 1 se puede establecer para cualquier función de activación ϕ satisfaciendo las siguientes condiciones:

(1) $\phi \in C^2(\mathbb{R},\mathbb{R})$,

(2) $\phi(0) \neq 0, \quad \phi'(0) \neq 0,$

(3) $|\phi''(z)|$ está acotada en $\mathbb{R}$.

Comentario 2. El sistema lineal que aparece en la demostración de la proposición 4

$$\begin{pmatrix} 1 & 1 \\ c_0h & c_1h \end{pmatrix}\begin{pmatrix} \lambda_0 \\ \lambda_1 \end{pmatrix} = \begin{pmatrix} 2A_0 \\ 4A_1 \end{pmatrix},$$

puede estar mal condicionado. Este hecho no implica la inestabilidad de la SLFN resultante, porque depende de los pesos en la siguiente forma

$$N(x) \approx (\lambda_0+\lambda_1)/2+(c_0h\lambda_0+c_1h\lambda_1)x/4.$$

Corolario 1 *Sea* $p^*(t)=a_0+a_1t$ *un polinomio de grado 1, en una variable y con coeficientes reales. Dado* $\varepsilon > 0$ *y dos números arbitrarios reales* t_l *y* t_u *tales que* $t_l < t_u$, *existe una SLFN con dos neuronas*

$$N^*(t)=\sum_{j=0}^{1}\lambda_j^* s(\alpha_j^* t+\beta_j^*),$$

tal que

$$|N^*(t)-p^*(t)|<\varepsilon. \; t\in[t_l,t_u],$$

donde $\lambda_j^*,\alpha_j^*,\beta_j^*,\; j=0,1,$ *son funciones de* a_0,a_1 *y* ε *que pueden ser expresadas de forma analítica cerrada.*

Demostración. Si hacemos el cambio de variable

$$t=(t_u-t_l)x+t_l,$$

y definimos

$$A_0 \equiv a_0 + a_1 t_l,$$

$$A_1 \equiv a_1(t_u - t_l),$$

el problema es equivalente a aproximar $p(\mathrm{x}) = A_0 + A_1 x$ sobre $[0,1]$. Sea $N(x)$ la red neuronal dada por la proposición 1, esto es , tal que

$$|N(x) - p(x)| < \varepsilon, \quad x \in [0,1],$$

Como $x = (t - t_l)/(t_u - t_l)$

$$|N(x) - p(x)| = |N((t - t_l)/(t_u - t_l)) - a_0 - a_1 t|, \quad t \in [t_l, t_u].$$

Como $N(x) = \sum_{j=0}^{1} \lambda_j s(\alpha_j x),$ se sigue que

$$N^*(t) = \sum_{j=0}^{1} \lambda_j \, s((\alpha_j t - \alpha_j t_l)/(t_u - t_l)).$$

$\square$

El principal resultado de esta sección es el siguiente

.

Teorema 1 *Sea* $a(\mathbf{x}) = \mathbf{a}.\mathbf{x} + a_0$ *una aplicación afín definida sobre el d-simplex* $T = < \mathbf{v}_0, \mathbf{v}_1 \ldots \mathbf{v}_d >$ *. Dado* $\varepsilon > 0$*, existe una SLFN con dos neuronas*

$$N(\mathbf{x}) = \sum_{j=0}^{1} c_j s(\mathbf{w}_j.\mathbf{x} + b_j),$$

tal que

$$|N(\mathbf{x}) - a(\mathbf{x})| < \varepsilon, \quad \mathbf{x} \in T, \tag{7}$$

donde $c_j, b_j \in \mathbb{R}$ *y* $\mathbf{w}_j \in \mathbb{R}^d$ *son funciones de* $\mathbf{a}$*,* a_0 *, y* ε *que pueden ser expresadas en una forma analítica cerrada.*

Demostración. Sea $\mathbf{u} = \frac{\mathbf{a}}{\|\mathbf{a}\|}$. Consideremos la línea recta $\mathbf{x} = t\mathbf{u}$. La función afín sobre esta línea recta es

$$h(t) = a(\mathbf{x}(t) = t\mathbf{a}.\mathbf{u} + a_0.$$

Definimos

$$t_u = \max_{i=0,\dots,d} \{\mathbf{u.v}_i\},$$

$$t_l = \max_{i=0,\dots,d} \{\mathbf{u.v}_i\},$$

$h(\mathrm{t})$ es un polinomio de grado 1 definido sobre $[t_l, t_u]$, por ello, por el Corolario 1, existe una red $N^*(t)$ satisfaciendo

$$|N^*(t) - h(t)| < \varepsilon \text{ para todo } t \in [t_l, t_u]. \tag{8}$$

Definimos

$$N(\mathbf{x}) \equiv N^*(\mathbf{u}.\mathbf{x}).$$

Para probar (7) es suficiente considerar que cualquier $\mathbf{x} \in T$ puede ser escrito como

$$\mathbf{x} = \mathbf{x}_0 + t\mathbf{u},$$

donde $\mathbf{x}_0$ es la proyección de $\mathbf{x}$ sobre el subespacio afín conteniendo el origen y que es ortogonal a $\mathbf{u}$, por ello $\mathbf{x}_0.\mathbf{u}=0$ y $t \in [t_l, t_u]$. Dado que

$$N(\mathbf{x}) = N^*(\mathbf{u}.(\mathbf{x}_0 + t\mathbf{u})) = N^*(t),$$

$$a(\mathbf{x}) = \mathbf{a}.(\mathbf{x}_0 + t\mathbf{u}) + a_0 = t\mathbf{au} + a_0 = h(t),$$

el resultado sigue de (8),

$\square$

3. REFERENCIAS BIBLIOGRÁFICAS

[1]. ATTALI, J.G. y PAGÉS, G. (1997). Approximations of functions by a multilayer perceptron: a new approach. Neural Networks, 10, pp.1069-1081.

[2]. MALAKOOTI y ZHOU YINGQING. (1998). Approximating Polynomial Functions by Feedforward Artificial Neural Networks: Capacity Analysis and Design. Applied Mathematics and Computation, 90, pp.27-52.

[3] LLANAS, B. y SÁINZ, F.J. (2008). Fast training of neural trees by adaptive splitting based on cubature. Neurocomputing en prensa (DOI:10.1016/j.neucom.2007.12.001).