DZIAŁ NAUKOWY

MATEMATYKA STOSOWANA 10, 2009

Jarosław Michalkiewicz (Wrocław)

Zmodyfikowane twierdzenie Kołmogorowa

Streszczenie. Praca dotyczy modyfikacji twierdzenia Kołmogorowa w celu przedstawienia aproksymacji ciągłej funkcji wielu zmiennych w postaci złożenia aproksymacji ciągłej funkcji jednej zmiennej zwanej wewnętrzną, z wieloma funkcjami, zwanymi zewnętrznymi, przy czym wszystkie funkcje są jednej zmiennej. Twierdzenie Kołmogorowa nie określa jak znaleźć funkcje wewnętrzną. Autor proponuje zastosowanie funkcji liniowej lub całkowitoliczbowej. Zmodyfikowana przez autora postać twierdzenia Kołmogorowa znajduje zastosowanie w teorii i praktyce sieci neuronowych oraz identyfikacji obiektów w automatyce.
Słowa kluczowe: aproksymacja, twierdzenie Kołmogorowa, nieliniowe systemy dynamiczne.

1. Wstęp. Prekursorem twierdzenia Kołmogorowa odnośnie funkcji wielu zmiennych był amerykański matematyk Hilbert. Podał on tzw. trzynasty problem dotyczący funkcji wielu zmiennych. Brzmi on: „Udowodnij, że są funkcje ciągłe trzech zmiennych, które nie mogą być reprezentowane przez funkcje ciągłe dwóch zmiennych". Kołmogorow podjął ten temat i udowodnił że, nie ma w matematyce funkcji ciągłej wielu zmiennych na zbiorze zwartym, których nie da się przedstawić w postaci sumy funkcji jednej zmiennej.

Kołmogorow podał i udowodnił twierdzenie dotyczące przedstawienia funkcji wielu zmiennych w postaci złożenia (superpozycji) funkcji jednej zmiennej. Twierdzenie to okazało się przydatne w teorii sieci neuronowych, ponieważ opierając się na tym twierdzeniu można udowodnić, że sieć neuronowa aproksymuje funkcje ciągłą.

Oryginalne twierdzenie Kołmogorowa nie nadaje się do konstrukcji algorytmu numerycznego mimo, że mówi o istnieniu funkcji $f_{p,q}$, nie podaje jak je znaleźć.

Autor zmodyfikował twierdzenie Kołmogorowa przez wprowadzenie funkcji liniowej lub całkowitoliczbowej jako wewnętrznej funkcji aktywacji.

Twierdzenie Kołmogorowa o reprezentacji ciągłej funkcji wielu zmiennych wyraża się wzorem (Kolmogorov, 1957)

$$h(x_1,\ldots,x_n) = \sum_{q=1}^{2n+1} g_q\Big(\sum_{p=1}^{n} f_{p,q}(x_p)\Big), \tag{1}$$

gdzie h jest dowolną funkcją n zmiennych określoną na jednostkowej kostce zaś g_q i $f_{p,q}$ są odpowiednio zewnętrzną i wewnętrznymi funkcjami aktywacji.

Aby zbudować sztuczną sieć neuronową (Kosiński, 2002), należy określić model pojedynczego neuronu. Jednym z najczęściej używanych modeli neuronu jest model McCullocha Pittsa. Stan takiego neuronu określony jest następującymi wzorami

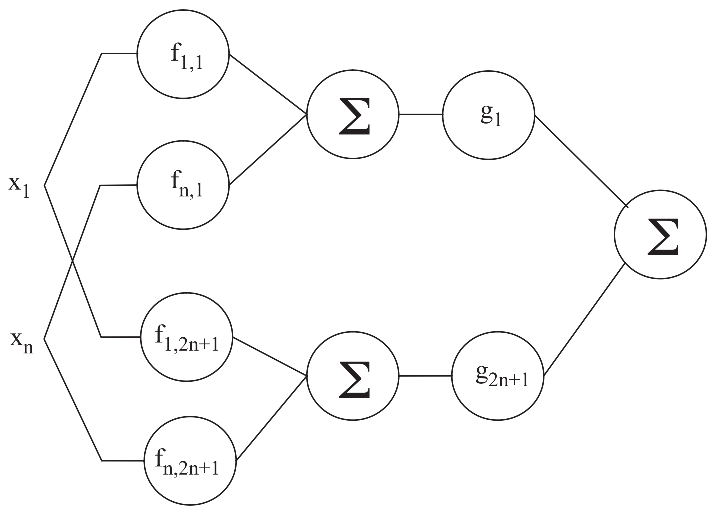
$$\sigma_i(t) = f[h_i(t) - T_i],$$

gdzie

$$h_i(t) = \sum_{j=1}^{n} J_{ij}\sigma_j(t-1),$$

$\sigma_i(t)$ jest stanem neuronu i-tego w czasie t, f funkcją aktywacji neuronu, $h_i(t)$ jest polem lokalnym działającym na i-ty neuron w chwili t , pochodzącym od wszystkich n neuronów połączonych z i-tym neuronem, T_i progiem zadziałania i-tego neuronu, J_{ij} siłą połączenia synaptycznego między i-tym neuronem a j-tym neuronem, a $\sigma_j(t-1)$ stanem j-tego neuronu w chwili wcześniejszej $t-1$.

Sumę stojącą po prawej stronie równania (1) można zrealizować przy pomocy warstwowej sieci neuronowej z odpowiednio dobranymi funkcjami aktywacji, co przedstawiono na rysunku 1.

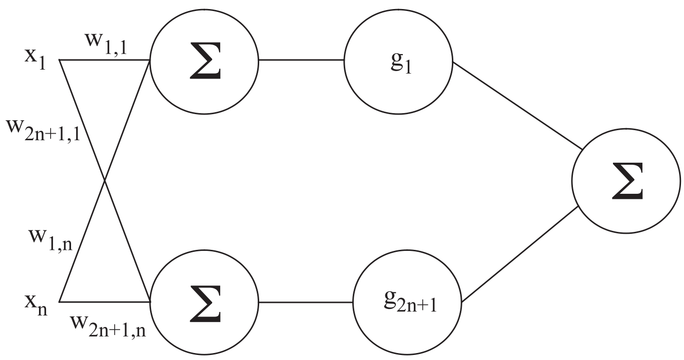


Rys. 1. Sieć neuronowa Kołmogorowa.

W tej pracy autor zajął się aproksymacją reprezentacji Kołmogorowa (1) i zaproponował następujący algorytm:

1) określanie funkcji g_q na podstawie danej funkcji h,
2) przyjęcie liniowej bądź całkowitoliczbowej $f_{p,q}$, aby zagwarantować zbieżność algorytmu.

Sieć neuronowa odpowiadająca liniowej funkcji aktywacji przedstawiono na rysunku 2.



Rys. 2. Klasyczna zmodyfikowana sieć neuronowa Kołmogorowa.

Bibliografia dotycząca aproksymacyjnych właściwości sieci jest bardzo obszerna, ale jej wyniki nie są konstruktywne, ponieważ nie podają wzoru na funkcję aproksymującą.

Następujący naukowcy zajmowali się tym problemem: Cybenko (Cybenko, 1989), Hornik, Stinchcombe, White (Hornik, 1989), Mhaskar, Micchelli (Mhaskar, 1995), Chui, Li (Chui, 1992), Leshno, Lin, Pinkus i Schoken (Leshno, 1993).

Konstruktywny algorytm przedstawiony przez Sprechera (Sprecher, 1996), (Sprecher, 1997) jest skomplikowany i wymaga modelowania trzech warstw.

Konstruktywny algorytm przedstawiony przez Igielnika (Igielnik, 2003) używa funkcji sklejanych do modelowania sieci neuronowej i wymaga użycia skomplikowanego aparatu matematycznego.

Cybenko opierając się na analizie funkcjonalnej udowodnił, że klasa funkcji

$$G(x) = \sum_{j=1}^{K} c_j f(w_j^T x + \Theta_j) \tag{2}$$

jest gęsta w $C(I_n)$, to jest w zbiorze funkcji ciągłych na kostce I_n, gdzie $w_j \in \mathbb{R}^n$, $c_j \in \mathbb{R}$, $\Theta_j \in \mathbb{R}$, $j = 1, \ldots, K$, K – dowolne (Cybenko, 1989).

Również Hornik, Stinchcombe i White, używając twierdzenia Stone'a udowodnili, że sieć neuronowa przybliża funkcje ciągłe (Hornik, 1989).

Leshno, Lin, Pinkus i Schocken udowodnili następującą tezę: jeśli sieć ma mieć właściwości aproksymacyjne, to zewnętrzna funkcja aktywacji nie może być wielomianem (Leshno, 1993).

Mhaskar i Micchelli (Mhaskar, 1995) rozważali zbiór funkcji:

$$A_\Phi = \left\{ \Phi(Ax - t) : A \in M, t \in \mathbb{R}^d \right\}, \tag{3}$$

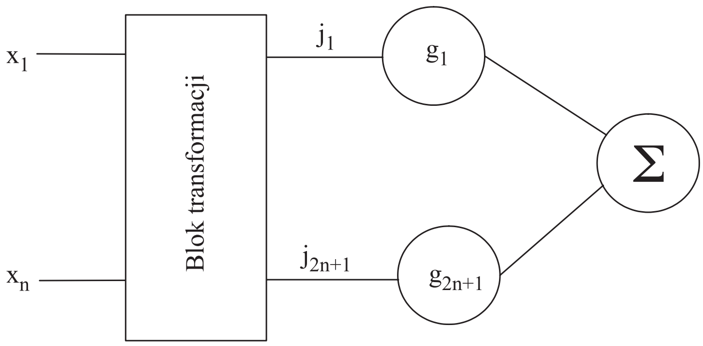
gdzie Φ jest funkcją aktywacji, A macierzą wag z przestrzeni M, x wejściem i t przesunięciem. Jeśli $d = 1$ zbiór funkcji A_Φ może być interpretowana jako wyjście sieci neuronowej. Warunkiem, aby zbiór funkcji A_Φ w zbiorze funkcji pewnej klasy był gęsty, jest prosta inkluzja przy okresowej funkcji Φ. Napisanie warunku inkluzji dla sieci dla autora wydaje się trudne.

Chui i Li (Chui, 1992) rozważali właściwości aproksymacyjne funkcji wypukłych (wielomianowych) i określili, jakie kryteria muszą spełniać wagi, aby funkcje miały właściwości aproksymacyjne.

Wspomniane metody aproksymacji funkcji ciągłej różnią się zasadniczo od metod autora. Różnica polega na tym, że autor zakłada stałe wagi i dobiera zewnętrzną funkcje aktywacji, natomiast inni autorzy dobierają wagi przy ustalonej funkcji aktywacji.

Autor proponuje prosty algorytm numeryczny do uczenia sieci neuronowej oparty na:

1) zmodyfikowanym twierdzeniu Kołmogorowa, Rysunek 2,
2) zmodyfikowanym indeksowanym twierdzeniu Kołmogorowa, Rysunek 3.



Rys. 3. Sieć neuronowa zmodyfikowana.

Zmodyfikowana sieć neuronowa Kołmogorowa jest nieliniowa, niesigmoidalna, jednowarstwowa z dowolnie małym błędem aproksymacji. Algorytm jest dobrze zbieżny, prosty w obliczeniach i pozwala na ominięcie skomplikowanych metod gradientowych oraz wstecznej propagacji. Metoda prezentowana przez autora jest nieparametryczna, ponieważ nie zakłada się postaci funkcji g.

Wewnętrzna funkcja aktywacji jest określona w węzłach siatki przestrzennej i może być nieciągła i dlatego jest nieróżniczkowalna. Girosi i Poggio (Girosi, 1989) określili taką sieć jaką nieużyteczną do uczenia z dwóch powodów:

1) ponieważ wewnętrzna funkcja aktywacji jest nieróżniczkowalna,
2) ponieważ sieć jest nieparametryzowalna.

Z poglądem tym polemizowała Kurkowa (Kurkova, 1991). W swoim artykule zakwestionowała tezę Girosi i Poggio udowadniając, że dla problemu aproksymacji nie musi być spełniony warunek różniczkowalności i parametryzowalności. Poza tym zwróciła uwagę, że funkcje nieróżniczkowalne można przybliżyć ciągiem funkcji różniczkowalnych. Podobnie uważa autor, ponieważ nieciągłe funkcje można przybliżyć ciągiem funkcji ciągłych lub różniczkowalnych (oczywiście bez wymagania jednostajnej zbieżności). Poza tym autor formułuje problem w kategorii aproksymacji, stąd spełnione są warunki twierdzeń w artykule Kurkowej.

Próby modyfikacji twierdzenia Kołmogorowa podejmowali się także Lorentz i Sprecher, ale rozwiązania tych autorów są mało istotne lub bardzo skomplikowane w obliczeniach i stąd mało użyteczne.

Lorentz wykazał, że można zastąpić funkcje g_q przez jedną funkcje g, a Sprecher zmodyfikował funkcje $f_{p,q}$ przez $\lambda_p f_q$ i twierdzenie Kołmogorowa w tej wersji wyraża się wzorem

$$h(x_1,\ldots,x_n) = \sum_{q=1}^{2n+1} g\Big(\sum_{p=1}^{n} \lambda_p f_q(x_p)\Big) \tag{4}$$

Według Sprechera

$$h(x_1,\ldots,x_n) = \sum_{q=1}^{2n+1} g_q\Big(\sum_{p=1}^{n} \lambda_p f(x_p+(q-1)a)\Big) \tag{5}$$

Próbę konkretnego określenia funkcji $f_{p,q}$ podjął Sprecher lecz jej postać jest bardzo skomplikowana w obliczeniach.

Podane niżej zmodyfikowane twierdzenia Kołmogorowa można potraktować jako schematy numeryczne nadające się do użycia w sieciach neuronowych.

2. Zmodyfikowane twierdzenie Kołmogorowa. Niech dana będzie funkcja n zmiennych $h(x_1,\ldots,x_n)$.

Niech $D_k = \Big\{\frac{1}{(9n)^k}(i-1) : i = 2,\ldots (9n)^k+1\Big\}$.

Zbiór D_k wyznacza „ziarno” na odcinku $[0,1]$ i jak łatwo zauważyć, ze wzrostem k „ziarno” będzie coraz gęstsze. Tak przyjęty zbiór jest zbiorem lewych końców przedziałów z pracy Kołmogorowa.

Niech $D_k^n = D_k \times D_k \times \ldots \times D_k$, n razy.

Zbiór D_k^n jest „siatką przestrzenną” odpowiadającą zbiorowi D_k.

Niech $k = 1, 2, \ldots, 2 \leq q \leq m_k = (9n)^k + 1$.

Rozpatrzmy przedziały

$$A_{k,i}^q = \frac{1}{(9n)^k}\Big[i - 1 - \frac{q}{3n}, i - \frac{1}{3n} - \frac{q}{3n}\Big] \tag{6}$$

Niech

$$S_{k,i_1,\ldots,i_n}^q = \prod_{p=1}^n A_{k,i_p}^q \tag{7}$$

Lemat 1 (o pokryciu). *Zbiór hipersześcianów $S_{k,i_1,\ldots,i_n}^q$ przy ustalonym k i zmiennym q oraz $i_1, \ldots, i_n$ pokrywa zbiór D_k^n tak, że każdy punkt z D_k^n jest pokryty $2n + 1$ razy.*

Dowód. Prosty pomijamy. □

Zachodzi następujące twierdzenie.

Twierdzenie 1 (zmodyfikowane twierdzenie Kołmogorowa). *Dowolną funkcje $h(x_1, \ldots, x_n)$ ciągłą i rzeczywistą na $E \times \ldots \times E$ (n-razy) w punktach zbioru D_k^n można aproksymować wzorem*

$$ha(x_1, \ldots, x_n) = \sum_{q=1}^{2n+1} g_q\Big(\sum_{p=1}^n w_{q,p} x_p\Big), \tag{8}$$

z dowolnie małym błędem, gdzie g_q są rzeczywiste i dane wzorem

$$g_q(y) = \frac{1}{2n+1} h(\varsigma_{k,i_1}^q, \ldots, \varsigma_{k,i_n}^q), \qquad y = \sum_{p=1}^n w_{q,p} x_p, \tag{9),(10}$$

zaś

$$\varsigma_{k,i_p}^q \in A_{k,i_p}^q \tag{11}$$

i są jego lewym końcem, a $w_{q,p}$ są współczynnikami spełniającymi warunek

$$w_{q,p} \geq w_{q,p-1}((9n)^k + 1). \tag{12}$$

Dowód. Niech

$$y = \sum_{p=1}^n w_{q,p} x_p. \tag{13}$$

Funkcja określona wzorem (10) przekształca zbiór $S_{k,i_1,\ldots,i_n}^q$ w pewien zbiór $\Delta_{k,i_1,\ldots,i_n}^q$. Ponieważ funkcja ta jest ciągła (jako suma funkcji ciągłych), zatem na mocy twierdzenia o przekształceniu ciągłym przestrzeni spójnej wnioskujemy, że obraz spójnego zbioru $S^q_{k,i_1,\ldots,i_n}$, tzn. $\Delta^q_{k,i_1,\ldots,i_n}$, jest zbiorem spójnym.

Następnie należy tak dobrać wagi synaptyczne, aby przedziały $\Delta^q_{k,i_1,\ldots,i_n}$ nie pokrywały się. Zauważmy, że rozpatrujemy wartości funkcji wyrażonej wzorem (10) na „siatce przestrzennej" D^n_k, wystarczy więc, aby obrazy węzłów „siatki przestrzennej" nie nakładały się. Jest wiele sposobów na dobór współczynników funkcji liniowych. Wystarczy, aby zachodził wzór (12).

Są jeszcze inne metody doboru współczynników.

Dla funkcji n wymiarowej $w_{q,j} = \log p_j$, gdzie $p_j, j = 1, 2, \ldots, n$, są liczbami pierwszymi.

Funkcja

$$\sum_{j=1}^{n} i_j \log p_j = \log \prod_{j=1}^{n} p_j^{i_j} \tag{14}$$

jest różnowartościowa dla różnych i_j, $j = 1, 2, \ldots, n$, bo rozkład liczby na iloczyn liczb pierwszych jest jednoznaczny, a to gwarantuje, aby obrazy węzłów „siatki przestrzennej" nie nakładały się.

Niech dana będzie funkcja

$$ha(x_1, \ldots, x_n) = \sum_{q=1}^{2n+1} g_q\Big(\sum_{p=1}^{n} w_{q,p} x_p\Big). \tag{15}$$

Rozpatrzmy

$$M = \sup_{E^n} |h| \,. \tag{16}$$

Niech

$$\varsigma^q_{k,i_p} \in A^q_{k,i_p} \tag{17}$$

są jego lewym końcem, wtedy

$$y \in \Delta^q_{k,i_1,\ldots,i_n} \tag{18}$$

są jego lewym końcem.

Wzór na g_q jest następujący

$$g_q(y) = \frac{1}{2n+1} h(\varsigma^q_{k,i_1}, \ldots, \varsigma^q_{k,i_n}).$$

Dalej

$$\begin{aligned} &h(x_1, \ldots, x_n) - ha(x_1, \ldots, x_n) \\ &\qquad = h(x_1, \ldots, x_n) - \sum_{q=1}^{2n+1} \left\{ g_q\Big(\sum_{p=1}^{n} w_{q,p} x_p\Big) \right\}. \end{aligned} \tag{19}$$

Z kolei dla sumy zachodzi

$$g_q(y) = \frac{1}{2n+1} h(\varsigma^q_{k,i_1}, \ldots, \varsigma^q_{k,i_n}) = \frac{1}{2n+1} h(x_1, \ldots, x_n) + \omega, \tag{20}$$

gdzie

$$|\omega| \leq \frac{1}{2n+1}\frac{1}{l}M,$$

bo tak można dobrać k, aby

$$\left| h(x_1, \ldots, x_n) - h(\varsigma^q_{k,i_1}, \ldots, \varsigma^q_{k,i_n}) \right| \leq \frac{1}{l}M \tag{21}$$

na dowolnym przedziale $A^q_{k,i}$ (co gwarantuje lemat 1).

Stąd po przekształceniach

$$|h(x_1, \ldots, x_n) - ha(x_1, \ldots, x_n)| \leq \frac{1}{l}M, \tag{22}$$

czyli

$$ha \to h \quad \text{gdy} \quad l \to \infty \tag{23), (24}$$

co kończy dowód. □

3. Zmodyfikowane indeksowane twierdzenie Kołmogorowa. Niech i^q_p wyznacza się z warunku definicyjnego zbioru D_k, to znaczy

$$\frac{1}{(9n)^k}\left(i^q_p - 2 - \frac{q}{3n}\right) < x_p \leq \frac{1}{(9n)^k}\left(i^q_p - 1 - \frac{q}{3n}\right) \tag{26}$$

a

$$j_q = \sum_{p=1}^{n} i^q_p N^{p-1} \quad \text{i} \quad N = \operatorname*{card}_i D_k. \tag{26), (27}$$

Niech

$$\zeta^q_p = \frac{1}{(9n)^k}\left(i^q_p - 1 - \frac{q}{3n}\right). \tag{28}$$

Twierdzenie 2 (zmodyfikowane indeksowane twierdzenie Kołmogorowa). *Dowolną funkcję $h(x_1, \ldots, x_n)$ ciągłą i rzeczywistą na $E \times \ldots \times E(n-$ razy) można aproksymować rzeczywistą funkcją $ha(x_1, \ldots, x_n)$*

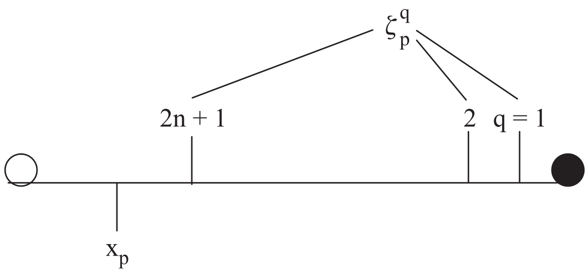
$$ha(x_1, \ldots, x_n) = \sum_{q=1}^{2n+1} g_q[j_q] \tag{29}$$

z dowolnie małym błędem gdzie $g_q[j_q]$ jest określone wzorem

$$g_q[j_q] = \frac{h(\zeta^q_1, \ldots, \zeta^q_n)}{2n+1} \quad \text{i} \quad 1 \leq p \leq n,\ 1 \leq q \leq l. \tag{30}$$

Oczywiste jest, że funkcja $ha(x_1, \ldots, x_n)$ jest stałą na węzłach „siatki przestrzennej".

Wzajemne położenie punktów ζ_p^q, x_p mieszczącego się w j-tym przedziale przedstawia rysunek 4.



Rys. 4. Fragment j-ty dziedziny.

Niech m będzie współczynnikiem określającym dokładność aproksymacji.

Dowód. Niech $ha(x_1, \ldots, x_n)$ będzie określone wzorem (29). Niech $g_q[j_q]$ będzie określone wzorem (30)

Jak łatwo zauważyć zachodzi

$$h(x_1, \ldots, x_n) - ha(x_1, \ldots, x_n) = h(x_1, \ldots, x_n) - \sum_q \{g_q[j_q]\}\,. \tag{31}$$

Z kolei dla składników sumy zachodzi

$$g_q[j_q] = \frac{h(x_1, \ldots, x_n)}{2n+1} + \varpi, \tag{32}$$

gdzie

$$|\varpi| \leq \frac{1}{2n+1}\frac{1}{m}\sup_{E^n}|h(x_1, \ldots, x_n)|\,, \tag{33}$$

bo dla dostatecznie dużego k zbiór D_k jest tak gęsty, że wahanie funkcji h na węzłach „siatki przestrzennej" jest tak małe, że

$$|h(x_1, \ldots, x_n) - h(\zeta_1^q, \ldots, \zeta_n^q)| \leq \frac{1}{m}\sup_{E^n}|h(x_1, \ldots, x_n)| = N.$$

Stąd po przekształceniach

$$\sup_{E^n}|h(x_1, \ldots, x_n) - ha(x_1, \ldots, x_n)| \leq \frac{1}{m}\sup_{E^n}|h(x_1, \ldots, x_n)|\,. \tag{34}$$

Czyli $ha \to h$ gdy $m \to \infty$ co kończy dowód. □

Błąd oszacowany wzorem (34) uwzględnia zarówno błąd uczenia jak i błąd uogólniania por. (Osowski, 1996). Wynikają z tego wnioski, że:

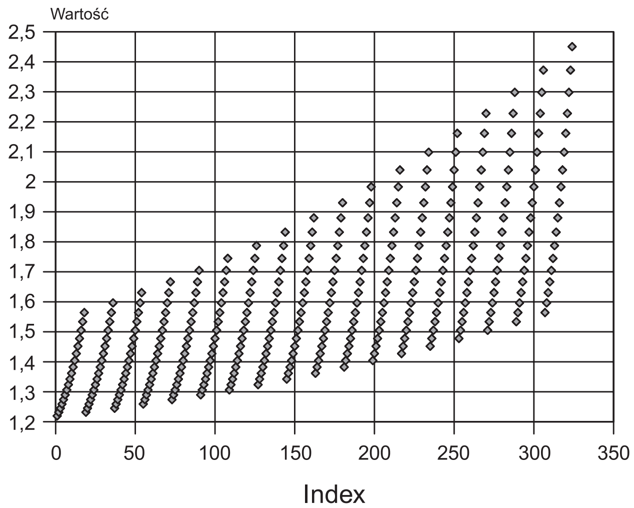
1) błąd oszacowany wzorem (34) można zmniejszyć zwiększając liczbę próbek uczących czyli zagęszczając siatkę przestrzenną,
2) nie ma optymalnej liczby próbek uczących jak w klasycznych sieciach sigmoidalnych, im więcej próbek tym mniejszy błąd, por. (Osowski, 1996),
3) błąd zależy od wahania funkcji aproksymowanej, im mniejsze wahanie tym mniejszy błąd.

Największy błąd aproksymacji (por. wzór (34)) nie przewyższa wahania funkcji aproksymowanej na elementach siatki przestrzennej N. Wahanie funkcji jest tu określone inaczej niż w klasycznych pracach matematycznych por. (Fichtenholz, 1980). Po pierwsze wahanie funkcji jest określone dla funkcji jednej zmiennej. W literaturze brak jest uogólnienia wahania na funkcje wielu zmiennych. Można tu zaproponować następującą metodę postępowania. Należy określić krzywe w przestrzeni dziedziny funkcji wielu zmiennych i dla każdej krzywej określić wahanie (jako funkcji jednej zmiennej). Następnie określić wahanie jako supremum wahań po wszystkich krzywych. Wygodniej jest jednak zastosować metodę określenia wahania jako różnicy między największą a najmniejszą wartością na elemencie siatki przestrzennej (dziedzinie).

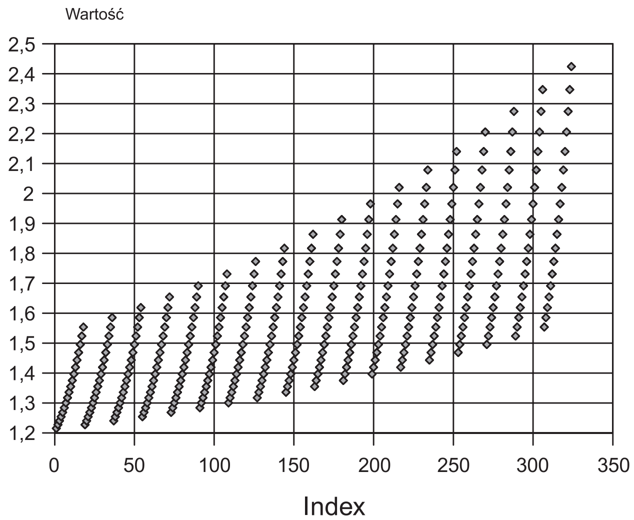
Błąd całkowity (będący błędem uczenia i uogólniania) należy określać dla konkretnej postaci funkcji wielu zmiennych. Biorąc pod uwagę błąd dopuszczalny należy dobrać wymiar kraty i co za tym idzie ilość próbek uczących. I tak np. dla funkcji $y = x_1 + x_2$ dla kostki $[0,1]^2$ podzielonej na 18 części mamy wahanie $\frac{1}{18}2$.

4. Algorytm uczenia sieci. Badanie eksperymentalne algorytmu uczenia sieci jest podane w (Michalkiewicz, 2004). Zakładamy, że mamy do wykonania następujące zadanie: dany jest ciąg uczący $\left\{(x^i, y^i)\right\}_{i=1}^{N}$ gdzie $x^i = [x_1^i, \ldots, x_n^i]$; $y^i = h(x_1^i, \ldots, x_n^i)$ i na tej podstawie należy określić funkcje aktywacji g_q, $q = 1, 2, \ldots, l$ w sieci neuronowej z rysunku 3. Zamiast x_p^i wprowadzono *indeks* i_p^q zgodnie z nierównością (25). Nowa koncepcja sieci neuronowej, podobnie jak zmodyfikowane twierdzenie Kołmogorowa, pozwala na budowę algorytmu, umożliwiającego wyznaczenie funkcji g_q na podstawie wzoru (30). Dla sieci klasycznej zmodyfikowanej Kołmogorowa funkcje g_q określamy na podstawie wzoru (9).

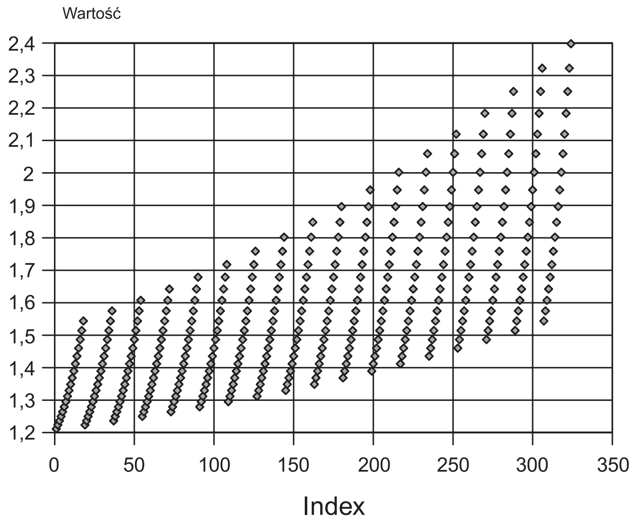
Działania symulacyjne algorytmu dla funkcji $h_1 = 5 + \exp(x_1 + x_2)$ dla $k = 1$ i przedstawiono na rysunkach 5–9. Na osi poziomej oznaczono indeks, zaś na osi pionowej odkłada się błąd aproksymacji liczony wg (34), różne punkty odpowiadają różnym punktom ciągu uczącego.



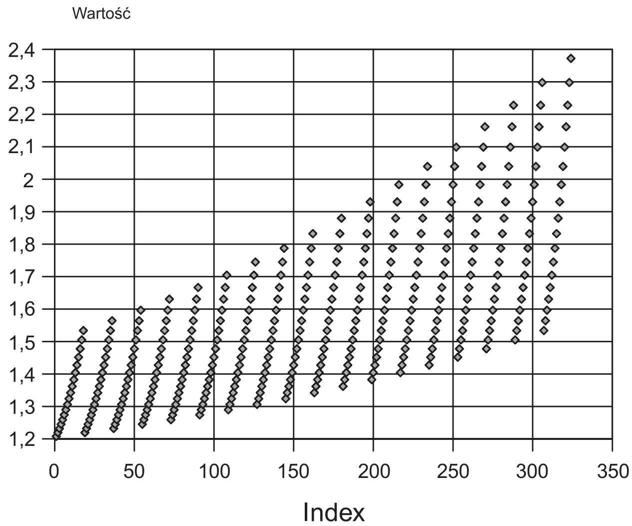
Rys. 5. Pierwsza funkcja aktywacji, tutaj i dalej index = indeks.



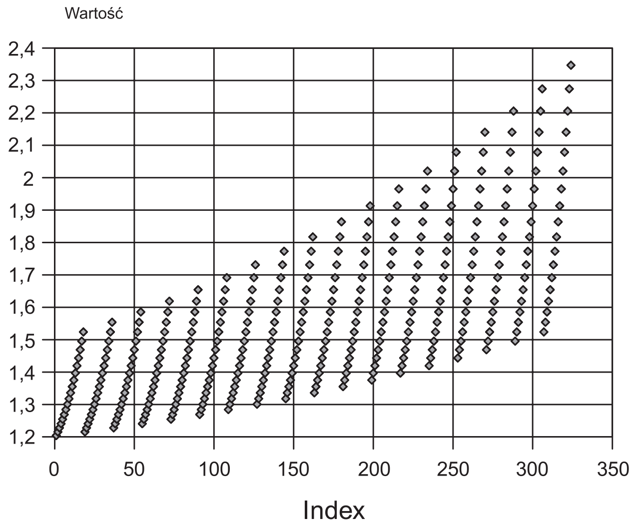
Rys. 6. Druga funkcja aktywacji.



Rys. 7. Trzecia funkcja aktywacji.



Rys. 8. Czwarta funkcja aktywacji.

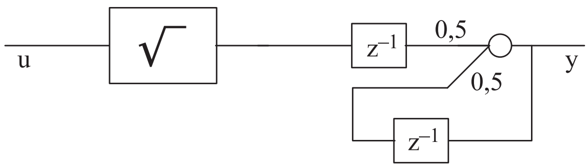


Rys. 9. Piąta funkcja aktywacji.

5. System dynamiczny Hammersteina. Badanie przeprowadzono na obiekcie jak na rysunku 10. Obiekt można opisać równaniem

$$y[i] = 0,5y[i-1] + 0,5\sqrt{u[i-1]} \tag{35}$$

Otrzymano pięć podobnych funkcji aktywacji obiektu. Pierwsza z nich jest na poniższym rysunku.



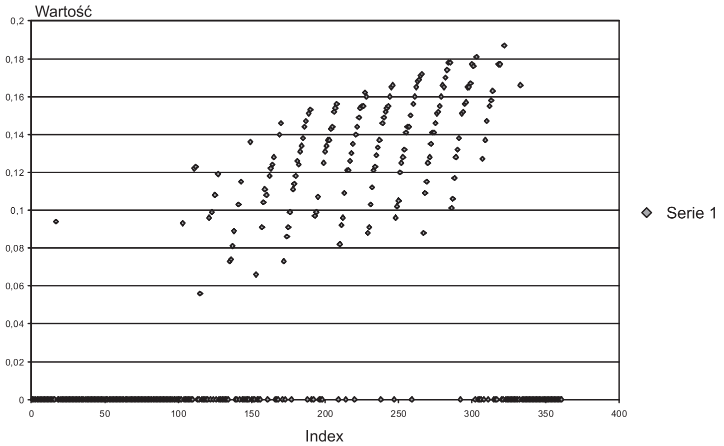
Rys. 10. System Hammersteina.

Podobne badania eksperymentalne są możliwe dla systemu Wienera (Michalkiewicz, 2007).

6. Zakończenie. Podane wyżej dwa twierdzenia autora umożliwiają konstrukcję prostego algorytmu numerycznego.

Autor proponuje następujący algorytm:

1) określanie funkcji g_q na podstawie danej funkcji h,
2) przyjęcie funkcji $f_{p,q}$ całkowitoliczbowych lub liniowych tak, aby zapewniona była zbieżność algorytmu.



Rys. 11. Pierwsza funkcja aktywacji systemu Hammersteina.

Modyfikacja twierdzenia Kołmogorowa polega na przejściu do problemu aproksymacji i wprowadzeniu funkcji o wartościach całkowitych lub liniowych jako wewnętrznej funkcji $f_{p,q}$. Umożliwia to budowę algorytmu numerycznego na wyznaczanie funkcji g_q zbieżnego do dowolnej funkcji h z zadaną dowolną dokładnością.

Twierdzenie 1 mówi o aproksymacji funkcji ciągłej. Wyprowadzone twierdzenie jest prawdziwe dla węzłów siatki przestrzennej, ale można je uogólnić dla dowolnego punktu.

Zawarta w twierdzeniu nr 2 teza odnośnie aproksymowanej funkcji wielu zmiennych jest nową koncepcją sieci neuronowej typu Kołmogorowa. Algorytm uczenia sieci wynikający z podanego twierdzenia jest prosty i łatwy w obliczeniach. Indeks konieczny dla dobrania zewnętrznej funkcji aktywacji jest określony na podstawie nierówności (25) dla żądanej wartości dziedziny. Dysponując indeksem korzystamy z odpowiedniego wzoru (29), aby wyznaczyć wartości zewnętrznej funkcji i aproksymującej funkcji.

Podane twierdzenia mogą mieć zastosowania do identyfikacji obiektów dynamicznych i prostego sposobu uczenia sieci neuronowych.

Bibliografia

Chui C. K., Li X.: *Approximation by ridge functions and neural network with one hidden layer*, Journal of Approximation Theory, **70**, pp. 131–141, 1992.

Cybenko G.: *Approximation by superposition of a sigmoidal function*, Mathematics of Control, Signals, and Systems, **2**, pp. 303–314, 1989.

Engekling R., Sieklucki K.: *Wprowadzenie do topologii*, PWN Warszawa 1986.

Fichtenholz G. M.: *Rachunek różniczkowy i całkowy.* Tom I,II,III. PWN Warszawa 1980.

Girosi F., Poggio T.: *Representation properties of networks: Kolmogorov's theorem is irrelevant*, Neural Computation, **1**, pp. 465–469, 1989.

Hornik K. , Stinchcombe M. , White H.: *Multilayer feedforward networks are universal approximators*, Neural Networks, **2**, pp. 359–366, 1989.

Igielnik B.: *Kolmogorovs Spline Network*, IEEE Transactions on Neural Networks, **14**, No 4, July 2003.

Kołmogorow A. N.: *On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition* (in Russian), Dokl. Akad. Nauk ZSRR, **114**, pp. 953–956, 1957.

Kosiński R. A.: *Sztuczne sieci neuronowe, dynamika nieliniowa i chaos*, WNT Warszawa 2002.

Kurkova V.: *Kolmogorov's theorem is relevant*, Neural Computation, **3**, pp. 617–622, 1991.

Leshno M., Lin V. Y., Pinkus A., Schocken S.: *Multilayer feedforward networks with a nonpolynomial activation function can approximate any function*, Neural Networks, **6**, pp. 861–867, 1993.

Lorentz G. G.: *The 13th problem of Hilbert*, Proceeding of Symposia in Pure Mathematics, **28**, 1976.

Mhaskar H. N., Micchelli C. A.: *Degree of approximation by neural and translation network with a single hidden layer*, Advances in Applied Mathematics, **16**, pp. 151–183, 1995.

Michalkiewicz J. , Hasiewicz Z.: *An algorithm for training a neural network based on Kolmogorovs modified theorem*, pp. 1339–1342, MMAR Międzyzdroje 2004.

Michalkiewicz J.: *Polynomial Kolmogorov's neural network in the identification of Wiener systems*, pp. 691–694, MMAR Szczecin 2007.

Michalkiewicz J.: *New concept of Kolmogorov's neural network*, pp. 77–82, Artificial Intelligence and Soft Computing, Springer, Warsaw 2006.

Michalkiewicz J.: *Kolmogorov's neural network in the identification of Hammerstein system*, pp. 915–918, MMAR Międzyzdroje 2005.

Osowski S.: *Sieci neuronowe w ujęciu algorytmicznym*, WNT Warszawa 1996.

Sprecher D. A.: *A numerical implementation of Kolmogorov's superpositions*, Neural Networks, **9**, pp. 765–772, 1996.

Sprecher D. A.: *A numerical implementation of Kolmogorov's superpositions II*, Neural Networks, **10**, pp. 447–457, 1997.

Jarosław Michalkiewicz
ul. Kwiska 12/5
54-210 Wrocław
E-mail: jarekmichal@poczta.onet.pl

Modified Kolmogorov's theorem

Abstract. The article takes up the modification of the Kolmogorov's representation theorem of a multivariant continuous function as a superposition of one continuous function, called internal, with many continuous functions, called external, all of one variable.

The Kolmogorov's theorem does not determine how to find the internal function. Author suggests an application of functions of particular forms. This requires a modification of the theorem. The form of the Kolmogorov's theorem modified by the Author finds its application in the theory and practise of neural networks and in the identification of

objects in the automatics. The modified Kolmogorov's theorem enables the author to construct a simple computer algorithm.
Keywords: approximation, Kolmogorov's theorem, non-linear dynamical systems.

(*wpłynęło 18 lutego 2008 r.*)