PRACE NAUKOWE UNIWERSYTETU EKONOMICZNEGO WE WROCŁAWIU nr 107
Taksonomia 17 2010

Małgorzata Gliwa
Akademia Ekonomiczna w Katowicach

MAPY KOHONENA W KLASYFIKACJI OBIEKTÓW SYMBOLICZNYCH

Streszczenie: Celem artykułu było przedstawienie własności map Kohonena wykorzystywanych do klasyfikacji i wizualizacji obiektów symbolicznych. W artykule zaprezentowano podejście realizowane według algorytmu Stochastic Approximation, który jest uogólnieniem klasycznej sieci Kohonena. Przedstawiono również przykład zastosowania algorytmu metody do klasyfikacji obiektów symbolicznych z rzeczywistego zbioru danych. Obliczenia zostały wykonane za pomocą programu SODAS.

1. Wstęp

Metody klasyfikacji danych symbolicznych rozwijają się bardzo dynamicznie od końca lat 80. XX wieku (zob. [Bock, Diday 2000, s. 2]). Wśród nich wymienić można: hierarchiczne metody aglomeracyjne oparte na macierzy odległości między obiektami symbolicznymi, hierarchiczną metodę aglomeracyjną stosowaną bezpośrednio dla zbioru obiektów symbolicznych, metodę piramid, metodę *k*-średnich czy samoorganizującą się sieć neuronową Kohonena dla obiektów symbolicznych. Część z tych metod dedykowana jest wyłącznie obiektom symbolicznym, jednak większą grupę stanowią metody będące adaptacjami klasycznych metod klasyfikacji.

Celem artykułu jest przedstawienie własności samoorganizującej się sieci neuronowej Kohonena dla obiektów symbolicznych. W części empirycznej przedstawiono przykład zastosowania przedmiotowej metody w klasyfikacji, a także wizualizacji zbioru obiektów symbolicznych z przestrzeni wielowymiarowej na płaszczyźnie. Obliczenia wykonano za pomocą modułów SYKSOM (*Kohonen Self-Organising*), HIPYR (*Hierarchical and Pyramidal Clustering*) programu SODAS[1].

2. Charakterystyka samoorganizującej się sieci neuronowej Kohonena dla obiektów symbolicznych

Artykuł Hansa-Hermanna Bocka opublikowany w 1997 r. (zob. [Bock 1997]) można uznać za początek badań nad samoorganizującymi się sieciami neuronowymi Kohonena dla obiektów symbolicznych, zwanych sieciami lub mapami Ko-

[1] SODAS to darmowe oprogramowanie wykorzystywane w analizie danych symbolicznych.

honena dla obiektów symbolicznych. Jest to sieć uczona bez nauczyciela. Złożona jest z dwóch warstw neuronów: wejściowej i wyjściowej. Warstwę wejściową stanowią obiekty symboliczne charakteryzowane przez *p* zmiennych przedziałowych $o_k = \left(\left[a_{k1}, b_{k1} \right], \ ..., \ \left[a_{kp}, b_{kp} \right] \right), \ k = 1, ..., n$. Obiekty te w przestrzeni *p*-wymiarowej są hiperkostkmi Q_k, które rzutowane są na dwuwymiarową warstwę wyjściową zwaną mapą Kohonena. Każdy neuron mapy Kohonena reprezentuje pewne miniskupienie, które charakteryzowane jest przez tzw. prototypy.

Powstałe prototypy, podobnie jak obiekty symboliczne, również opisane są przez *p* zmiennych przedziałowych [Bock 2004, s. 6-10]. W przestrzeni *p*-wymiarowej są one hiperkostkami, które w dalszej części artykułu oznaczone zostały jako *G*.

Rodzaj prototypu miniskupienia definiowany jest następująco [Bock 2008, s. 220-221]:

a) hiperkostka *G*, która obejmuje wszystkie hiperkostki Q_k, $k = 1, \ ..., \ n$:

$$G = \left[u, v \right], \tag{1}$$

gdzie:

$$u = \left(u_1, ..., u_p \right), \ v = \left(v_1, ..., v_p \right) \text{ oraz } u_j = \min_{k=1,...,n} a_{kj}, \ v_j = \max_{k=1,...,n} b_{kj}, \ j = 1, \ ..., \ m. \tag{2}$$

Prototyp skonstruowany według powyższej formuły jest wrażliwy na obiekty oddalone. W celu przezwyciężenia tego problemu proponuje się wprowadzenie prototypu $\hat{G} = \left[\hat{u}, \hat{v} \right]$, którego współrzędne określane są następująco:

b) $$\hat{u} = \left(0{,}5 + \gamma \right) u + \left(0{,}5 - \gamma \right) v \text{ oraz } \hat{v} = \left(0{,}5 - \gamma \right) u + \left(0{,}5 + \gamma \right) v, \tag{3}$$

gdzie $0 < \gamma < 1/2$.

Kolejną propozycją jest taki prototyp, który będzie rozwiązaniem następującego problemu optymalizacyjnego:

c) $$\sum_{k \in C} d\left(Q_k, \ G \right) \rightarrow \min_{\mathrm{G}}. \tag{4}$$

Zauważyć należy, że rozwiązanie powyższego problemu zależy od przyjętej miary odległości $d\left(\cdot \right)$ (zob. [Bock 2005]).

Sieć Kohonena dla obiektów symbolicznych tworzona jest według trzech algorytmów: Stochastic Approximation oraz Mac Queen 1 i Mac Queen 2 [Bock 2004, s. 9]. Algorytm Stochastic Approximation jest bezpośrednim uogólnieniem klasycznej sieci Kohonena, natomiast algorytmy Mac Queen 1 i Mac Queen 2 są uogólnieniami klasycznych algorytmów sekwencyjnych Mac Queena [Bock 2004, s. 9]. W dalszej część artykułu, w tym w części empirycznej, rozważone zostanie podejście realizowane według algorytmu Stochastic Approximation.

3. Algorytm sieci Kohonena dla obiektów symbolicznych

Algorytm uczenia sieci Kohonena dla obiektów symbolicznych $o_1, ..., o_n$ realizowany jest w poniższych krokach [Bock 2004, s. 18-20]:

Krok 1.

W etapie $t = 0$ następuje ustalenie wymiaru sieci (liczby neuronów) m oraz m pustych miniskupień $C_1^{(0)} = \phi, ..., C_m^{(0)} = \phi$ charakteryzowanych przez prototypy $p_1^{(0)}, \ldots, p_m^{(0)}$. Prototypy te stanowią albo m pierwszych, albo m losowo wybranych obiektów symbolicznych.

Krok 2.

Załóżmy, że miniskupienia $C_1^{(t)}, \ldots, C_m^{(t)}$ oznaczają podział obiektów symbolicznych $\{o_1, ..., o_t\}$ na koniec etapu t.

W etapach $t \to t+1$ prezentowane są kolejno obiekty symboliczne ze zbioru wejściowego. Dla każdego neuronu oblicza się odległość obiektu symbolicznego od prototypu. Neuron, który znajduje się najbliżej obiektu wejściowego, zostaje zwycięzcą:

$$d\left(o_{t+1}, p_{i^*}^{(t)}\right) = \min_{j=1,\ldots,m} d\left(o_{t+1}, p_j^{(t)}\right), \tag{5}$$

gdzie $i^* = 1, ..., m$, natomiast $d(\cdot, \cdot)$ jest miarą odległości pomiędzy obiektem symbolicznym o_{t+1} a prototypem $p_j^{(t)}$ (zob. [Bock 2008, s. 218-220]).

Krok 3.

– Obiekt o_{t+1} włączony zostaje do miniskupienia charakteryzowanego przez neuron zwycięski:

$$C_{i^*}^{(t+1)} = C_j^{(t)} \cup \{o_{t+1}\}, \; j = 1, ..., m. \tag{6}$$

– Struktury pozostałych klas nie zmieniają się: $C_i^{(t+1)} = C_i^{(t)}$ dla $i \neq i^*$, $i = 1, ..., m$. (7)

– Zwycięzca pociąga za sobą swoich sąsiadów z mapy, którzy dzięki bliskości ze zwycięzcą uzyskują przywilej zmiany swoich współrzędnych. Zatem dla $j \neq i^*$ aktualizowane są współrzędne prototypów $p_j^{(t+1)} = \left[u_j^{(t+1)}, v_j^{(t+1)}\right]$:

$$u_j^{(t+1)} = u_j^{(t)} + \alpha_{t+1} \cdot K_{i^* j} \cdot \left(a_{t+1} - u_j^t\right) \quad \text{oraz} \quad v_j^{(t+1)} = v_j^{(t)} + \alpha_{t+1} \cdot K_{i^* j} \cdot \left(b_{t+1} - v_j^t\right), \tag{8}$$

gdzie $j = 1, ..., m$, $t = 0, 1, ...$,

$K_{i^* j}$ jest funkcją sąsiedztwa, która decyduje o liczbie i intensywności uczenia się neuronów z sąsiedztwa [Bock 2008, s. 224-225]:

a) $$K_{i^* j} = K\left(\delta\left(P_{i^*}, P_j\right)\right) = \begin{cases} 1, \text{ dla } \delta \le \varepsilon \\ 0, \text{ dla } \delta > \varepsilon \end{cases}, \tag{9}$$

gdzie $\varepsilon \in \{0, 1, 2, 3, 4\}$ to promień sąsiedztwa, czyli parametr wskazujący, ile neuronów wokół neuronu zwycięskiego ma się uczyć wraz z nim, natomiast δ określa, jak daleko na mapie znajdują się od siebie neurony wyrażone przez współrzędne P_{i^*} i P_j,

b) $$K_{i^* j} = K\left(\delta\left(P_{i^*}, P_j\right)\right) = e^{-\delta^2/2} \text{ dla } \delta \ge 0, \tag{10}$$

c) $$K_{i^* j} = K\left(\delta\left(P_{i^*}, P_j\right)\right) = e^{-\delta} \text{ dla } \delta \ge 0, \tag{11}$$

natomiast $\alpha_t > 0$ określa współczynnik uczenia dla etapu t definiowany następująco [Bock 2008, s. 223]:

a) $$\alpha_t = \frac{1}{t}, \text{ dla } t = 1, 2, ... \tag{12}$$

b) $$\alpha_t = \frac{1}{t \bmod n}, \text{ dla } t = 1, 2, ... \tag{13}$$

Krok 4. Procedura z kroków 2-3 powtarzana jest dla wszystkich obiektów ze zbioru.

Krok 5. Kroki 2-4 powtarzane są dla zadanej liczby iteracji uczących lub do momentu osiągnięcia kryterium stopu [Bock 2008, s.213]:

$$\Delta = \frac{\sum_{i=1}^{m} \left\| p_i^l - p_i^{l-1} \right\|^2}{\sum_{i=1}^{m} \left\| p_i^l \right\|^2} < \gamma, \tag{14}$$

gdzie l to numer pełnego cyklu, natomiast γ oznacza poziom precyzji ustalany przez użytkownika.

W module SYSOKM programu SODAS wszystkie parametry sieci Kohonena ustalane są arbitralnie przez użytkownika.

4. Przykład empiryczny

Prezentowany przykład stanowi ilustrację działania omawianej metody. Wykorzystano dane dotyczące modeli samochodów klasy C o silnikach benzynowych[2]. Na ich podstawie utworzono 34 obiekty symboliczne, scharakteryzowane przez

[2] Dane zgromadzono na podstawie informacji o cenie i parametrach technicznych poszczególnych modeli samochodów, które zamieszczone są na stronach internetowych odpowiednich producentów samochodów.

6 zmiennych przedziałowych: cenę samochodu[3] (zł), pojemność silnika (cm^3), moc silnika (KM), prędkość maksymalną (km/h), przyspieszenie $0-100\,\mathrm{km}/\mathrm{h}$ (s) oraz zużycie paliwa w cyklu miejskim (l).

W przykładzie analizowane były sieci o wymiarach od 2x2 do 7x7 oraz 5x7. Dla każdej sieci definiowano ten sam zestaw parametrów, a mianowicie: funkcję sąsiedztwa (10), odległość Hausdorffa jako miarę odległości pomiędzy obiektem symbolicznym a prototypem [Bock, Diday 2000, s. 302], współczynnik uczenia (12), rodzaj prototypu skupienia (1), 100 iteracji uczących oraz promień sąsiedztwa równy 2.

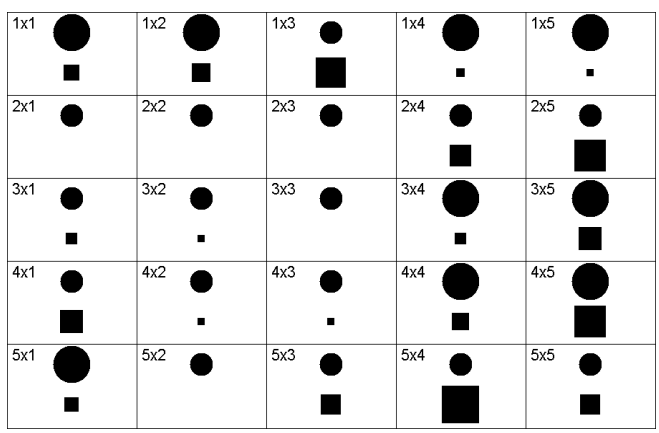
Kryterium wyboru rozmiaru sieci stanowił tzw. błąd kwantyzacji. Wyznaczany on był na podstawie odległości Hausdorffa odpowiedniego prototypu od obiektów wejściowych. Wartość błędu uczenia sieci to wartość średnia liczona dla całego zbioru uczącego. Dalszej analizie poddana została sieć o najmniejszym błędzie kwantyzacji, tj. sieć o wymiarach 5x5.

Tabela 1. Wartości błędu kwantyzacji dla poszczególnych sieci

Rozmiar sieci	2x2	3x3	4x4	5x5	5x7	6x6	7x7
Błąd kwantyzacji	9040,02	5409,91	2268,32	**876,90**	2899,85	3228,82	2625,68

Źródło: obliczenia własne.

W wyniku zastosowania modułu SYKSOM otrzymano następującą mapę Kohonena:

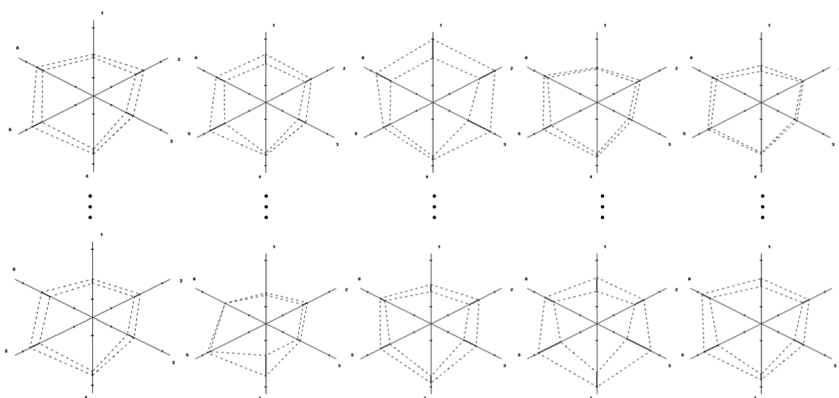


Rys. 1. Mapa Kohonena o wymiarach 5x5 (25 neuronów)

Źródło: opracowanie własne w programie SODAS v. 2.0.

[3] Ceny samochodów pochodzą z okresu od sierpnia do grudnia 2008 r.

Na rysunku 1 koła odzwierciedlają liczebność poszczególnych miniskupień, natomiast kwadraty objętość hiperkostek w przestrzeni 6-wymiarowej. Rysunek 1 to również wizualizacja zbioru obiektów symbolicznych z przestrzeni 6-wymiarowej na płaszczyźnie. Na takiej mapie trudno jest wyróżnić jednoznacznie klasy obiektów. Ponieważ sieć nie ma wewnętrznego kryterium pozwalającego ustalić ostateczną liczbę skupień, proponuje się utworzenie dla każdego prototypu wykresu dwuwymiarowego za pomocą techniki Zoom Star. Każdej zmiennej odpowiada oś wartości, która promieniście odchodzi od punktu centralnego. Na każdej osi zaznaczone są wartości poszczególnych zmiennych, które dla zmiennych przedziałowych odpowiadają dolnej i górnej granicy przedziału. Charakterystykę prototypu wyznaczają łamane łączące poszczególne wartości z osi zmiennych [Bock, Diday 2000, s. 125-138].



Rys. 2. Charakterystyki 10 z 25 prototypów otrzymane techniką Zoom Star

Źródło: opracowanie własne w programie SODAS v. 2.0.

Analizując otrzymane ilustracje graficzne prototypów, wskazać można miniskupienia, które mają te same lub podobne charakterystyki. Mogą one zostać połączone i w ten sposób będą odzwierciedlały podział zbioru obiektów symbolicznych na jednorodne skupienia.

SODAS generuje również plik tekstowy, który przedstawia przyporządkowanie poszczególnych obiektów wejściowych do miniskupień oraz charakterystyki prototypów:

```
LIST OF SYMBOLIC OBJECTS IN EACH CLUSTER
Cluster 1 ( 1x1)   Size 2
List of objects:
( 30)  VWBeetle
( 34)  VolvoC30
```

⋮

CLUSTER DESCRIPTION: PROTOTYPE OF THE CLUSTERS

Prototype 1 (1x1) Size 2

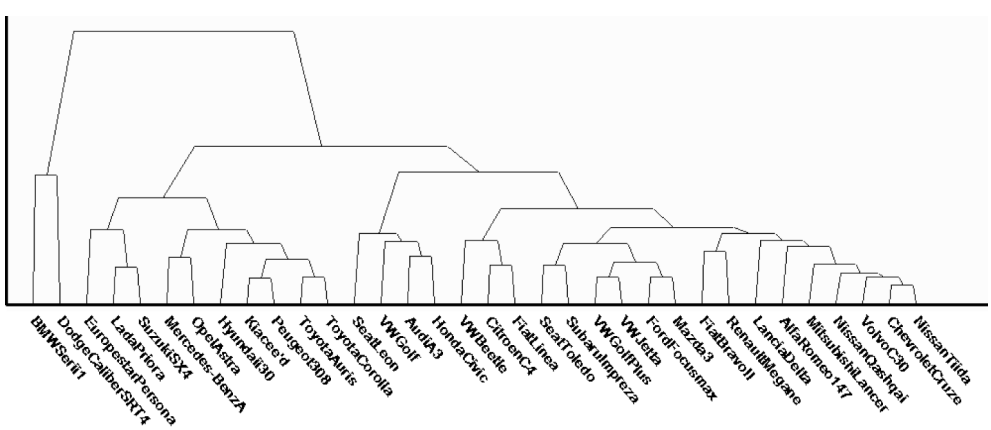
Variable	Minimum	Maximum	VMAP.Min	VMAP.max
cena	66870	78320	0.87	1.23
pojemnosc	1493	1991.5	0.42	1.60
moc	7.5	130	0.19	0.97
predmax	173	197.5	2.23	2.81
przyspiesz	10.15	13.2	1.74	3.16
spalanie_m	9.3	11.05	1.93	2.70

⋮

Migdał-Najman [2007] pokazała, że w przypadku analizy bardzo dużych zbiorów danych sieć Kohonena można zastosować jako preprocesor dla innych metod klasyfikacji. Duża liczba obiektów opisanych przez wiele zmiennych powoduje, że utrudniona jest analiza dendrogramów powstałych w wyniku zastosowania hierarchicznych metod aglomeracyjnych. Takie rozwiązanie pozwala również na analizę zbiorów z występującymi brakami danych. Sieć Kohonena identyfikuje także występujące w zbiorze danych wartości nietypowe i zazwyczaj przydziela je do osobnego miniskupienia. [Migdał-Najman 2007, s. 311-312]. Poniżej przedstawiono adaptację powyższego podejścia w odniesieniu do zbioru obiektów symbolicznych.

W etapie pierwszym dokonano klasyfikacji obiektów symbolicznych metodą najdalszego sąsiedztwa. Do wyznaczenia odległości pomiędzy obiektami symbolicznymi zastosowano znormalizowaną odległość Ichino-Yaguchiego [Ichino, Yaguchi 1994]. Jest to miara odległości stosowana dla obiektów symbolicznych.

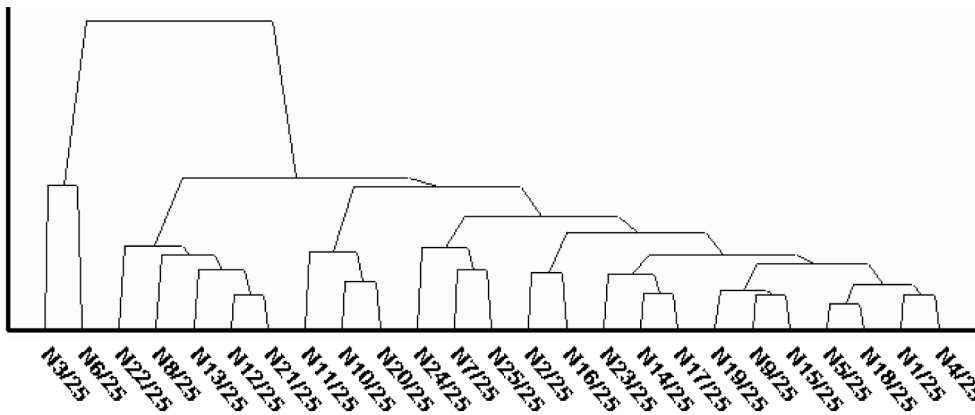
W wyniku zastosowania modułu HIPYR otrzymano następujący dendrogram:



Rys. 3. Dendrogram dla zbioru obiektów symbolicznych

Źródło: opracowanie własne w programie SODAS v. 2.0.

W etapie drugim 25 prototypów skupień wyodrębnionych za pomocą sieci Kohonena o wymiarach 5x5 wykorzystano jako zbiór danych wejściowych. Na jego podstawie dokonano klasyfikacji obiektów symbolicznych analogicznie do etapu pierwszego.



Rys. 4. Dendrogram dla prototypów skupień otrzymanych za pomocą sieci Kohonena

Źródło: opracowanie własne w programie SODAS v. 2.0.

Analizując przyporządkowanie obiektów symbolicznych ze zbioru wejściowego do miniskupień, przekonujemy się, że struktury klas uzyskane w etapie drugim nieznacznie różnią się od struktur uzyskanych w etapie pierwszym. Różnice te dotyczą 6 obiektów symbolicznych. Sieć Kohonena rozpoznała również dwa nietypowe obiekty symboliczne: BMW Serii 1 i Dodge Caliber SRT4, które przydzieliła do osobnych miniskupień N3/25 i N6/25 (por. rys. 4).

5. Podsumowanie

W artykule przedstawiono własności oraz przykład zastosowania sieci Kohonena dla obiektów symbolicznych. Dostępne oprogramowanie metod analizy danych symbolicznych pozwala na wykorzystanie mapy Kohonena dla obiektów symbolicznych opisanych tylko przez zmienne przedziałowe.

Pokazano, że sieć Kohonena może być stosowana nie tylko do klasyfikacji obiektów symbolicznych, ale także do wizualizacji wielowymiarowych obiektów symbolicznych na płaszczyźnie. Może być również wykorzystana w celu przygotowania danych do dalszej analizy. Wadą sieci Kohonena stosowanej dla obiektów symbolicznych jest jej wrażliwość na ustalane arbitralnie przez użytkownika parametry sieci.

Literatura

Bock H.H., *Visualizing Symbolic Data Tables by Kohonen maps: The SODAS module SYKSOM*, [w:] *User Manual for SODAS 2 Software [IST-200-25161*], http://www.info.fundp.ac.be/asso/sodaslink.htm, 2004.

Bock H.H., *Optimization in Symbolic Data Analysis: Dissimilarities, Class Centers, and Clustering*, [w:] *Data Analysis and Decision Support*, D. Baier, R. Decker, L. Schmidt-Thieme (red.), Springer-Verlag, Berlin 2005.

Bock H.H., *Simultaneous Visualization and Clustering Methods as an Alternative to Kohonen Maps*, [w:] *Learning, Networks and Statistics*, G. Della Riccia, H.J. Lenz, R. Kruse (red.), Springer-Verlag, New York 1997.

Bock H.H., *Visualizing Symbolic Data by Kohonen Maps*, [w:] *Symbolic Data Analysis and the SODAS Software*, E. Diday, M. Noirhomme-Fraiture (red.), Wiley, New York 2008.

Bock H.H., Diday E. (red.), *Analysis of Symbolic Data*, Springer Verlag, Berlin-Heidelberg 2000.

Ichino M., Yaguchi H., *Generalized Minkowski metrics for mixed feature type data analysis*, „IEEE Transactions on Systems, Man and Cybernetics" 1994 t. 24 (4).

Migdał-Najman K., *Propozycja hybrydowej metody grupowania dużych zbiorów danych wykorzystującej sieć Kohonena i taksonomiczne metody grupowania*, [w:] *Klasyfikacja i analiza danych – teoria i zastosowanie*, Taksonomia 14, K. Jajuga, M. Walesiak (red.), Prace Naukowe Akademii Ekonomicznej we Wrocławiu nr 1169, AE, Wrocław 2007.

KOHONEN MAPS IN CLASSIFICATION OF SYMBOLIC OBJECTS

Summary: The aim of this article is to present the properties of the Kohonen maps which are used to classify the symbolic objects. The properties of the Stochastic Approximation method are presented. This method is the direct generalization of Kohonen's classical self-organizing map. We also give an example of the application of the Kohonen maps to the classification of the symbolic objects. We use a real-world data set and SODAS software for our illustration.