

швидкого перетворення Фур'є fft та пошуку власних значень і векторів матриць EISPACK. Програма може успішно експлуатуватися на комп'ютерах під керуванням операційних систем Microsoft Windows 2000/XP/2003/Vista. Мінімальними апаратними вимогами є: процесор класу Intel Pentium III тактовою частотою 500 МГц; 96 МБ оперативної пам'яті; 200 МБ вільного місця на жорсткому диску; монітор SVGA і відеоадаптер з підтримкою розподільної здатності 1024x768.

Галузь застосування програми – підвищення якості зображень, отриманих за допомогою пристроїв ультразвукових досліджень або сейсмограм.

СПИСОК ЛІТЕРАТУРИ

1. *Newman, P.* The history of ultrasound / P. Newman, G. Rozycki // *Surgical Clinics of North America*. – 1998. – № 2 (78). – P. 179–195.
2. *Митьков, В. В.* Практическое руководство по ультразвуковой диагностике. Общая ультразвуковая диагностика / В. В. Митьков. – М.: Видар, 2003. – 720 с.
3. *Иванов, В. К.* Теория линейных некорректных задач и ее приложения / В. К. Иванов, В. В. Васин, В. П. Танана. – М.: Наука, 1978. – 200 с.
4. *Чайделрс, Д. Д.* Кепстр и его применение при обработке данных / Д. Д. Чайделрс, Д. П. Скиннер, Р. Ч. Кемерайт // *ТИИЭР*. – 1977. – № 10. – С. 5–23.
5. *Степаненко, А. А.* Повышение чувствительности ультразвуковой диагностики на основе метода параметрического спектрального анализа второго порядка / А. А. Степаненко, А. М. Ахметшин // *Клиническая информатика и телемедицина*. – 2005. – Т. 2, № 1. – С. 98–100.
6. *Степаненко, А. А.* Разложение суперпозиций неизвестных импульсных сигналов методом адаптивного спектрального анализа второго порядка / А. А. Степаненко, А. М. Ахметшин // *Искусственный интеллект*. – 2005. – № 3. – С. 610–618.

УДК 519.7:004.8

7. *Степаненко, А. А.* Анализ эхо-импульсных изображений слоистых структур: метод сигнального подпространства / А. А. Степаненко, А. М. Ахметшин // *Радиоэлектроника, информатика, управление*. – 2004. – № 1. – С. 5–9.
8. *O'Docherty, M.* Object-Oriented Analysis and Design / M. O'Docherty. – Wiley & Sons, 2005. – 560 p.
9. *Рыжиков, Ю.* Современный Фортран / Ю. Рыжиков. – Минск: Корона Принт, 2007. – 288 с.

Стаття надійшла до редакції 14.04.2011.

Степаненко А. А., Пиза Д. М.

ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ АНАЛИЗА И ОБРАБОТКИ ЭХО-ИМПУЛЬСНЫХ ИЗОБРАЖЕНИЙ

На основе анализа методов обработки эхо-импульсных изображений были выявлены недостатки существующих подходов и предложены два новых метода. Разработанные методы и базирующаяся на них программная система позволяют повысить эффективность визуального анализа медицинских и сейсмических ультразвуковых изображений.

Ключевые слова: математическая модель, метод линейного предсказания, суперпозиция импульсного сигнала, спектр, эхо-импульсное изображение, модуль математических вычислений.

Stepanenko O. O., Piza D. M.

PROGRAM COMPLEX FOR ANALYSIS AND TREATMENT ECHO-PULSE IMAGES

On the basis of analysis of methods of processing of echo-pulse images the lacks of existent approaches were deduced and two new methods are offered. Worked out methods and programmatic system which is based on them, allow to promote efficiency of visual analysis medical and seismic ultrasonic images.

Key words: mathematical model, method of linear prediction, superposition of impulsive signal, spectrum, echo-pulse images, module of mathematical calculations.

Шафроненко А. Ю.¹, Волкова В. В.², Бодянский Е. В.³

- ¹Стажер-исследователь Харьковского национального университета радиоэлектроники
²Старший преподаватель Харьковского национального университета радиоэлектроники
³Д-р техн. наук, профессор Харьковского национального университета радиоэлектроники

АДАПТИВНАЯ КЛАСТЕРИЗАЦИЯ ДАННЫХ С ПРОПУЩЕННЫМИ ЗНАЧЕНИЯМИ

Предложена адаптивная нейросетевая система, позволяющая решать задачу кластеризации данных с пропущенными значениями в on-line режиме с постоянной коррекцией восстанавливаемых элементов таблицы и центроидов кластеров. Введенная нейросистема характеризуется высоким быстродействием и простотой численной реализации.

Ключевые слова: адаптивная нейросетевая система, восстановление пропусков, кластеризация.

ВВЕДЕНИЕ

Задача кластеризации данных является важным элементом общей проблемы Data Mining, а для ее решения

сегодня существует множество подходов и алгоритмов: от сугубо интуитивных и эвристических до строго математических [1–9]. Вместе с тем, во многих задачах Data

Мінінг, включаючи, естественно, кластеризацію, вихідні таблиці даних «об'єкт-властивість» можуть містити порожні клітинки (пропуски), інформація в яких по тем або іншим причинам відсутня. Задачі відновлення таких пропущених спостережень приділяється достатнє увагу [6, 10–12], при чому найбільш ефективною в даній ситуації виявилися підходи, засновані на математичному апараті м'яких обчислень (чисельного інтелекту) [13, 14], і, зокрема, штучних нейронних мереж [12, 15–17]. Разом з тим, відомі підходи до відновлення пропусків і традиційні алгоритми кластеризації здатні лише в деяких випадках, коли вихідна таблиця задана априорно і кількість її рядків або стовпців не змінюється в процесі обробки. В той же час існує достатньо широка клас задач, коли дані надходять на обробку послідовно в on-line режимі, при цьому наперед невідомо, який з оброблюваних векторів-об'єктів може містити пропуски. При цьому процес відновлення даних і їх кластеризації повинні протікати одночасно в реальному часі.

ПОСТАНОВКА ЗАДАЧІ

Нехай задана вихідна $(N \times n)$ таблиця ($N \geq 1, n \geq 1$) «об'єкт-властивість».

Таблиця 1, що містить інформацію про N об'єктів, кожен з яких описується $(1 \times n)$ – вектором-рядком ознак $\vec{x}_i = (x_{i1}, \dots, x_{ip}, \dots, x_{ij}, \dots, x_{in})$, при цьому вважається, що N_G рядків можуть мати по одному пропуску, а $N_F = N - N_G$ заповнені повністю. В процесі обробки таблиці необхідно заповнити пропуски і сформувати m кластерів. При цьому вимагається, щоб відновлені елементи були б в певному сенсі «найбільш правдоподібні» або «близькі» до априорно невідомим закономірностям, що містяться в таблиці, які можуть невідомим чином змінюватися в процесі обробки.

АДАПТИВНИЙ АЛГОРИТМ ЗАПОЛНЕННЯ ПРОПУСКІВ

Для початку представимо таблицю 1 в вигляді $(N \times n)$ – матриці X , в якій відсутній один елемент x_{kj} або в більш загальному випадку відсутні N_G елементів. Вважається [10], що між стовпцями $\vec{x}_j = (x_{1j}, \dots, x_{ij}, \dots, x_{kj}, \dots, x_{Nj})^T$ існує лінійна кореляція, з урахуванням якої і про-

Таблиця 1.

	1	...	p	...	j	...	n
1	x_{11}	...	x_{1p}	...	x_{1j}	...	x_{1n}
...
i	x_{i1}	...	x_{ip}	...	x_{ij}	...	x_{in}
...
k	x_{k1}	...	x_{kp}	...	x_{kj}	...	x_{kn}
...
N	x_{N1}	...	x_{Np}	...	x_{Nj}	...	x_{Nn}
...

відбудеться відновлення пропуску з допомогою рівняння лінійної регресії

$$\hat{x}_{kj} = w_{j0} + w_{j1}x_{k1} + w_{j2}x_{k2} + \dots + w_{j,j-1}x_{k,j-1} + \dots + w_{jn}x_{kn} = \underline{X}_{kj} w_j, \quad (1)$$

де $w_j = (w_{j0}, w_{j1}, \dots, w_{jn})^T$ – $(n+1)$ – вектор-стовпець параметрів, підлягає визначенню; $\underline{X}_{kj} = (1, x_{k1}, \dots, x_{k,j-1}, x_{k,j+1}, \dots, x_{kn})$ – $(1 \times n)$ вектор-рядок ознак k -го об'єкта без kj -го елемента і з одиницею на першій позиції.

Вектор невідомих параметрів w_j може бути знайдено за допомогою стандартного методу найменших квадратів, для чого з матриці X слід виключити k -ю рядок, j -й стовпець, додати зліва стовпець, утворений одиницями, і на основі отриманої $((N-1) \times n)$ – матриці X_j обчислити оцінки параметрів

$$w_j = (X_j^T X_j)^+ X_j^T \vec{X}_j, \quad (2)$$

де $\vec{X}_j = (x_{1j}, \dots, x_{ij}, \dots, x_{k-1,j}, x_{k+1,j}, \dots, x_{Nj})^T$.

Якщо пропуски зустрічаються в N_G рядках і в різних стовпцях, з матриці X виключаються всі ці рядки і на основі усеченої $(N_F \times n)$ матриці n раз обчислюються вектори параметрів w_j (2) для всіх $j = 1, 2, \dots, n$. Далі з допомогою рівняння (1) заповнюються всі пропуски отриманими оцінками \hat{x}_{kj} .

Якщо дані надходять на обробку послідовно рядком за рядком, замість оцінки (2) може бути використано рекурентний метод найменших квадратів, який, однак, пропонує незмінність во часі всіх зв'язей, що існують в матриці даних. Експоненціально ж важивший рекурентний метод найменших квадратів може бути нестійким при малих значеннях параметра згладжування.

В зв'язі з цим для on-line аналізу даних представляється доцільним використання адаптивних алгоритмів навчання, що мають як фільтруючі, так і слідячі (для нестационарних ситуацій) властивості [18]. Якщо в процесі обробки масиву даних було проаналізовано і відновлено N рядків, то з приходом $(N+1)$ -го спостереження оцінки w_j уточнюються з допомогою адаптивної рекурентної процедури

$$\begin{cases} w_j(N+1) = w_j(N) + r_j^{-1}(N+1)(x_{N+1,j} - \underline{X}_{N+1,j} w_j(N)) \underline{X}_{N+1,j}^T, \\ r_j(N+1) = \alpha r_j(N) + \|\underline{X}_{N+1,j}\|^2, \end{cases} \quad (3)$$

де $0 \leq \alpha \leq 1$ – параметр згладжування, що задає компроміс між процесами фільтрації спостережень і слідженням за змінюючими характеристиками даних.

Обработку информации в режиме последовательного поступления данных удобно организовать с помощью нейросетевой системы, основными элементами которой являются n -параллельно работающих адаптивных линейных ассоциаторов (ALA) [19], настраиваемых с помощью алгоритма (3). При этом, если в k -й момент времени на вход системы поступает вектор наблюдений $\vec{x}_k = (x_{k1}, x_{k2}, \dots, x_{kin})$ полный или с пропусками, на выходе появляется он же, если \vec{x}_k полный, или его оценка, если в \hat{x}_k содержались пробелы.

АДАПТИВНЫЙ АЛГОРИТМ КЛАСТЕРИЗАЦИИ

Для решения задачи кластеризации в on-line режиме целесообразно воспользоваться самоорганизующейся картой Т. Кохонена [20], имеющей простую архитектуру с прямой передачей информации и кроме нулевого рецепторного слоя содержащей единственный слой нейронов, чаще всего тех же адаптивных линейных ассоциаторов. Каждый нейрон связан с каждым рецептором нулевого слоя прямыми связями и со всеми остальными нейронами поперечными внутрислойными (латеральными) связями. Именно латеральные связи обеспечивают возбуждение одних нейронов и торможение других.

Благодаря такой организации сети, каждый нейрон ALA получает всю информацию об анализируемом векторе-образе и генерирует на своем выходе соответствующий отклик, после чего между нейронами возникает конкуренция, в результате которой определяется единственный нейрон-победитель с максимальным выходным сигналом. Этот сигнал по латеральным связям обеспечивает возбуждение ближайших «соседей» победителя и подавление реакции далеко отстоящих нейронов. Таким образом в процессе конкурентного самообучения формируются группы нейронов, каждый из которых максимальным образом реагирует на образы «своих» кластеров-подобластей входного пространства сигналов.

Рассмотрим простейшую карту Кохонена, имеющую 1D топологию, n рецепторов и m (по числу возможных кластеров) нейронов, каждый из которых характеризуется собственным вектором синаптических весов $w_q^K, q = 1, 2, \dots, m$, определяющих центроид соответствующего кластера.

Каждый нейрон сети в k -й момент времени получает на входы n -мерный входной вектор

$$\tilde{x}_k = \frac{\hat{x}_k^T}{\|\hat{x}_k\|} \quad (4)$$

и генерирует на своем выходе сигнал

$$y_{kl} = w_l^{KT} (k-1) \tilde{x}_k, \quad (5)$$

зависящий от вектора синаптических весов $w_l^K (k-1)$, настроенных по данным, полученным до момента k , на определенную область (кластер) входного пространства. Близкие в смысле используемой метрики входные векторы \tilde{x}_k и \tilde{x}_i могут возбуждать либо один и тот же нейрон w_q^K , либо два нейрона соседа w_q^K и w_{q+1}^K или w_q^K и $w_{q-1}^K, q = 1, 2, \dots, m$.

В основе алгоритма самоорганизации карты Кохонена лежат принципы конкурентного самообучения [19], при этом как и любая другая процедура настройки нейронной сети, работа алгоритма начинается с инициализации синаптических весов сети, которые обычно выбираются с помощью генератора случайных чисел, при этом для каждого из нейронов должно выполняться условие

$$\|w_l^K(0)\| = 1, l = 1, 2, \dots, m. \quad (6)$$

Процедура самоорганизации реализуется в три основных этапа: конкуренции, кооперации и синаптической адаптации и начинается с анализа образа \tilde{x}_k , поступающего с рецепторного слоя на все нейроны сети. Для каждого из нейронов вычисляется расстояние

$$D(\tilde{x}_k, w_l^K (k-1)) = \|\tilde{x}_k - w_l^K (k-1)\|, \quad (7)$$

при этом, если входные векторы предварительно нормированы с помощью выражения (4) так, что

$$\|\tilde{x}_k\| = 1, \quad (8)$$

а в качестве расстояния (7) используется евклидова метрика, то мерой расстояния между \tilde{x}_k и $w_l^K (k-1)$ может служить

$$\begin{aligned} D(\tilde{x}_k, w_l^K (k-1)) &= 2(1 - \tilde{x}_k^T w_l^K (k-1)) = \\ &= 2(1 - \cos(\tilde{x}_k, w_l^K (k-1))), \end{aligned} \quad (9)$$

а мерой близости – скалярное произведение

$$-1 \leq \tilde{x}_k^T w_l^K (k-1) = \cos(\tilde{x}_k, w_l^K (k-1)) \leq 1. \quad (10)$$

Далее определяется нейрон-победитель, «ближайший» ко входному образу \hat{x}_k такой, что

$$D(\tilde{x}_k, w_l^K (k-1)) = \min_l D(\tilde{x}_k, w_l^K (k-1)), \quad (11)$$

после чего, временно опуская этап кооперации, можно подстроить синаптические веса сети с помощью модифицированного правила самообучения

$$w_q^K(k) = \begin{cases} \frac{w_q^K(k-1) + \eta(k)(\tilde{x}_k - w_q^K(k-1))}{\|w_q^K(k-1) + \eta(k)(\tilde{x}_k - w_q^K(k-1))\|}, & \text{если } q\text{-й нейрон победил,} \\ w_q^K(k-1) & \text{в противном случае.} \end{cases} \quad (12)$$

Несложно видеть, что процедура (12) реализует принцип «победитель получает все», при этом вектор синаптических весов $w_q^K(k-1)$ нейрона-победителя «подтягивается» ко входному вектору \tilde{x}_k на расстояние, определяемое шагом настройки $\eta(k)$.

Регулирование величины шага обычно производится, исходя из эмпирических соображений, при этом весьма удобной является процедура [21]

$$\eta(k) = r^{-1}(k), r(k) = \alpha r(k-1) + \|\tilde{x}_k\|^2 = \alpha r(k-1) + 1. \quad (13)$$

Несложно заметить структурную схожесть алгоритмов обучения (3) и (12), (13).

Одной из особенностей карты Кохонена является наличие этапа кооперации в процессе самоорганизации, когда нейрон-победитель определяет локальную область топологического соседства, в которой возбуждается не только он сам, но и его ближайшее окружение, при этом более близкие к победителю нейроны возбуждаются сильнее чем удаленные. Эта топологическая область определяется функцией соседства Φ_{ql} , зависящей от расстояния $D(w_q^K(k-1), w_l^K(k-1))$ между победителем $w_q^K(k-1)$ и любым из нейронов сети $w_l^K(k-1)$. Как правило, Φ_{ql} – это ядерная функция симметричная относительно максимума в точке $D(w_q^K(k-1), w_q^K(k-1)) = 0$ и принимающая в ней единичное значение, монотонно убывающая с ростом расстояния и стремящаяся к нулю при $D(w_q^K(k-1), w_l^K(k-1)) = 2$. При этом наиболее часто в качестве функции соседства используются гауссиан, конус, функции Епанечникова, «мексиканская шляпа» и т. п.

Использование функции соседства приводит к модифицированному правилу обучения

$$\begin{cases} w_l^K(k) = \frac{w_l^K(k-1) + r^{-1}(k)\Phi_{ql}(\tilde{x}_k - w_l^K(k-1))}{\|w_l^K(k-1) + r^{-1}(k)\Phi_{ql}(\tilde{x}_k - w_l^K(k-1))\|} \forall l, \\ r(k) = \alpha r(k-1) + 1, \end{cases} \quad (14)$$

реализующему принцип «победитель получает больше». При $\Phi_{ql} = \delta_{ql} = 1$ при $q=l$ и 0 в остальных случаях, вновь приходим к алгоритму (12), обеспечивающему на каждом такте настройку единственного нейрона-победителя $w_q^K(k-1)$.

В принципе, для обучения самоорганизующейся карты можно вообще обойтись без нахождения победителя, а в качестве функции соседства использовать некоторую конструкцию, зависящую от выходных сигналов нейронов (5) так, что

$$\begin{cases} w_l^K(k) = \frac{w_l^K(k-1) + r^{-1}(k)\varphi(y_{kl})(\tilde{x}_k - w_l^K(k-1))}{\|w_l^K(k-1) + r^{-1}(k)\varphi(y_{kl})(\tilde{x}_k - w_l^K(k-1))\|} \forall l, \\ r(k) = \alpha r(k-1) + 1. \end{cases} \quad (15)$$

Поскольку

$$y_{kl} = w_l^{KT}(k-1)\tilde{x}_k = \cos(\tilde{x}_k, w_l^K(k-1)), \quad (16)$$

то наиболее простой такой конструкцией является

$$\varphi(y_{kl}) = \frac{1 + y_{kl}}{2}, \quad (17)$$

удовлетворяющая всем условиям, предъявляемым к ядерным функциям соседства. Тогда окончательно алгоритм обучения карты Кохонена для нахождения центроидов m кластеров имеет вид

$$\begin{cases} w_l^K(k) = \frac{w_l^K(k-1) + r^{-1}(k)\frac{1 + y_{kl}}{2}(\tilde{x}_k - w_l^K(k-1))}{\|w_l^K(k-1) + r^{-1}(k)\frac{1 + y_{kl}}{2}(\tilde{x}_k - w_l^K(k-1))\|} \forall l, \\ r(k) = \alpha r(k-1) + 1. \end{cases} \quad (18)$$

Нейросетевая архитектура, реализующая процесс кластеризации данных с пропусками, приведена на рис. 1.

Как видно, введенная нейронная сеть образована однопотипными нейронами – адаптивными линейными ассоциаторами, настраиваемыми с помощью достаточно простых и очевидных алгоритмов обучения.

ЗАКЛЮЧЕНИЕ

Рассмотрена задача кластеризации данных, содержащихся в таблицах «объект-свойство» и имеющих пропуски, в режиме последовательного поступления этих данных на обработку. Предложена адаптивная нейросетевая система, позволяющая решать эту задачу в on-line режиме с постоянной коррекцией восстанавливаемых элементов таблицы и центроидов кластеров. Введенная нейросистема образована набором адаптивных линейных ассоциаторов, характеризуется высоким быстродействием и простотой численной реализации.

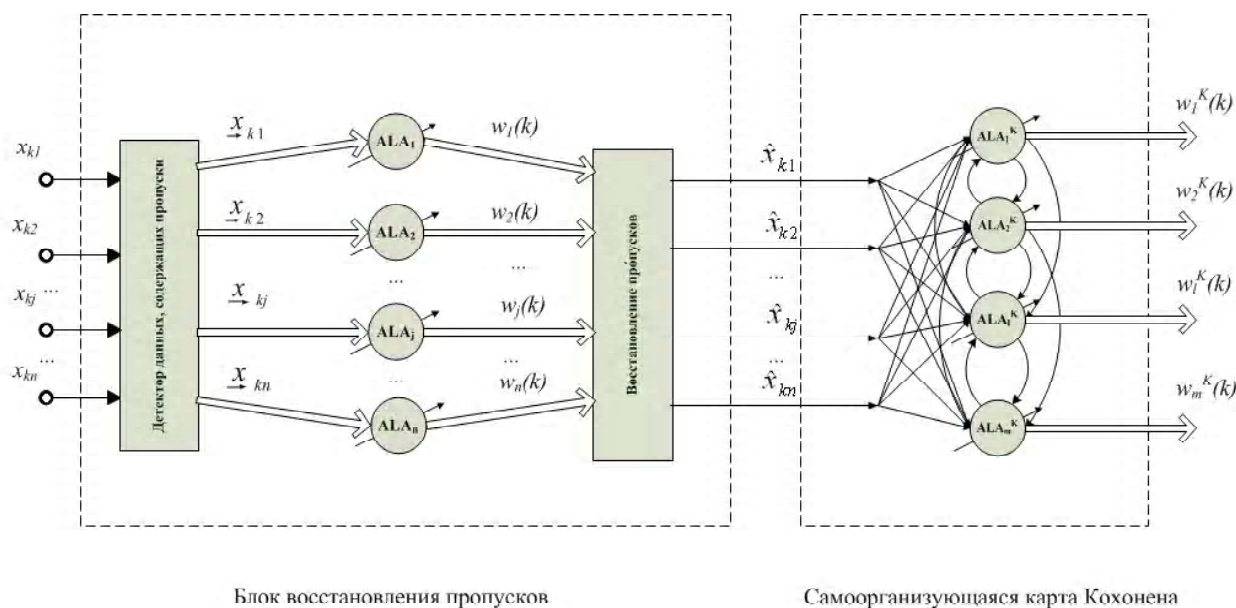


Рис. 1. Искусственная сеть для адаптивной кластеризации данных с пропущенными значениями

СПИСОК ЛИТЕРАТУРЫ

- Jain, A. Data clustering: A review / A. Jain, M. Murty, P. Flynn // ACM Computing Surveys. – 1999. – №3(31). – P. 264–323.
- Дюк, В. Data Mining / В. Дюк, А. Самойленко. – С. Пб. : Питер, 2001. – С. 368.
- Friedman, J. The Elements of Statistical Learning. Data Mining, Inference and Prediction / J. Friedman, T. Hastie, R. Tibshirani. – Berlin : Springer, 2003. – P. 552.
- Xu, R. Survey of clustering algorithms / R. Xu, D. Wunsch // IEEE Trans on Neural Networks. – 2005. – №3 (16). – P. 645–678.
- Han, J. Data Mining: Concepts and Techniques / J. Han, M. Kamber. – Amsterdam : Morgan Kaufman Publ, 2006. – P. 754.
- Gan, G. Data Clustering: Theory, Algorithms and Applications / G. Gan, Ch. Ma, J. Wu. – Philadelphia : SIAM, 2007. – P. 466.
- Abonyi, J. Cluster Analysis for Data Mining and System Identification / J. Abonyi, B. Feil. – Basel : Birkhaeuser, 2007. – P. 303.
- Olson, D. Z. Advanced Data Mining Techniques / D. Z. Olson, D. Dursun. – Berlin : Springer, 2008. – P. 180.
- Xu, R. II. Clustering / R. Xu, D. C. Wunsch. – Hoboken, N. J. : John Wiley & Sons, Inc., 2009. – P. 341.
- Загоруйко, Н. Г. Эмпирические предсказания. – Новосибирск : Наука, 1979. – С. 120.
- Gorban, A. Principal Manifolds for Data Visualization and Dimension Reduction. Lecture Notes in Computational Science and Engineering, Vol. 58 / A. Gorban, B. Kegl, B. Wunsch, A. (Eds.) Zinovyev. – Berlin-Heidelberg – New York : Springer, 2007. – P. 330.
- Marwala, T. Computational Intelligence for Missing Data Imputation, Estimation, and Management / Marwala T. // Knowledge Optimization Techniques. – Hershey – New York : Information Science Reference, 2009. – P. 303.
- Maimon, O. Soft computing for Knowledge Discovery and Data Mining / O. Maimon, L. Rokach. – New York : Springer-Verlag, 2007. – P. 448.
- Rutkowski, L. Computational Intelligence Methods and Techniques / L. Rutkowski. – Berlin-Heidelberg : Springer-Verlag, 2008. – P. 514.
- Bishop, C. M. Neural Networks for Pattern Recognition. – Oxford : Clarendon Press, 1995. – P. 482.
- Gorban, A. N. Neural network modeling of data with gaps / A. N. Gorban, A. A. Rossiev, D. C. Wunsch II //Радіоелектроніка, інформатика, управління. – 2000. – №1 (3). – С. 47–55.
- Tkacz, M. Artificial neural networks in incomplete data sets processing / In: Eds. M. A. Klopotek, S. T. Wierzchon, K. Trojanowski // Intelligent Information Processing and Web Mining. – Berlin-Heidelberg : Springer–Verlag, 2005. – P. 577–583.
- Bodyanskiy, Ye. An adaptive learning algorithm for a neuro-fuzzy network / Ye. Bodyanskiy, V. Kolodyazhnyi, A. Stephan // Ed. B. Reusch «Computational Intelligence. Theory and Applications». – Berlin-Heidelberg – New York : Springer, 2001. – P. 68–75.
- Haykin, S. Neural Networks. A Comprehensive Foundation / Haykin, S. – Upper Saddle River, N. J.: Prentice Hall, 1999 – P. 842.
- Kohonen, T. Self-Organizing Maps / T. Kohonen. – Berlin: Springer-Verlag, 1995. – P. 362.
- Бодянский, Е. В. Искусственные нейронные сети: архитектуры, обучение, применения / Е. В. Бодянский, О. Г. Руденко. – X. : ТЕЛЕТЕХ, 2004. – С. 372.

Стаття надійшла до редакції 06.05.2011.

Шафроненко А. Ю., Волкова В. В., Бодянский С. В.
АДАПТИВНА КЛАСТЕРИЗАЦІЯ ДАНИХ З ПРОПУЩЕНИМИ ЗНАЧЕННЯМИ.

Запропонована адаптивна нейромережева система, що дозволяє вирішувати задачу кластеризації даних з пропущеними значеннями в реальному часі з постійним коригуванням відновлених елементів таблиці і центрів кластерів. Впроваджена нейронна система характеризується високою швидкістю і простою чисельної реалізації.

Ключові слова: адаптивна нейромережева система, відновлення пропусків, кластеризація.

Shafronenko A.Yu. Volkova V. V., Bodyanskiy Yev. V.
ADAPTIVE CLUSTERING WITH MISSING VALUES

In this paper the adaptive neural network system that solves the clustering problem of data with gaps is proposed. This system allows to process the data in the on-line mode with a constant correction of recoverable table’s elements and centers of clusters. A proposed neural system has high speed and simple numerical realization.

Key words: adaptive neural network system, the restoration of gaps, clustering.