XIV Konferencja PLOUG
Szczyrk
Październik 2008

Wybrane metody redukcji wymiarowości danych oraz ich wizualizacji

Jarosław Gramacki, Artur Gramacki
Uniwersytet Zielonogórski
Instytut Informatyki i Elektroniki

e-mail: j.gramacki@iie.uz.zgora.pl, a.gramacki@iie.uz.zgora.pl

Abstrakt. W artykule pokazano wybrane metody redukcji wymiarowości zbioru danych numerycznych. Celem redukcji jest albo odkrywanie niewidocznych związków zachodzących w tych danych, albo też użyteczna ich wizualizacja (lub jedno i drugie). Posłużono się kilkoma wybranymi metodami, w większości natury statystycznej, oraz metodą opartą na działaniu specjalnego rodzaju sieci neuronowych – samoorganizujących się map (Self-Organizing Maps, SOM). Omówione metody zademonstrowano na bardzo prostych przykładach. Zwrócono również uwagę na te elementy modułu Oracle Data Mining, które związane są z redukcją wymiarowości. Ponadto pokazano jak stosunkowo łatwo oprogramować w języku PL/SQL dowolne (w tym przed-stawione w artykule) metody posługując się mało znanym specjalizowanym pakietem UTL_NLA dostępnym automatycznie w każdej instalacji serwera Oracle począwszy od wersji 10g.

Informacja o autorze. Dr inż. Jarosław Gramacki jest pracownikiem naukowym w Instytucie Informatyki i Elektroniki Uniwersytetu Zielonogórskiego. Zajmuje się projektowaniem, wykonywaniem oraz wdrażaniem aplikacji bazodanowych usprawniających szeroko rozumiane zarządzanie Uczelnią. Od wielu lat prowadzi również zajęcia dydaktyczne dotyczące projektowania baz danych, działania i administrowania systemami zarządzania bazami danych oraz wykorzystania technologii Oracle w budowie aplikacji użytkowych.
Dr inż. Artur Gramacki pracuje w Instytucie Informatyki i Elektroniki Uniwersytetu Zielonogórskiego na stanowisku adiunkta. Jego zainteresowania koncentrują się wokół szeroko rozumianych zagadnień związanych z bazami danych, w szczególności firmy Oracle. Oprócz prowadzenia zajęć dydaktycznych stara się wykorzystywać swoją wiedzę uczestnicząc w różnych projektach informatycznych z tego zakresu. Brał udział w pięciu projektach, których celem było przygotowanie systemów wspomagających działalność Uniwersytetu Zielonogórskiego.

1. Wprowadzenie

Powszechny dostęp do wydajnych i względnie tanich systemów baz danych powoduje, iż ilość przechowywanych danych jest ogromna i ciągle rośnie. Dane te często charakteryzują się dużą wymiarowością (ang. high-dimensional data sets), która staje się istotnym problemem podczas ich analizowania. Mówiąc o dużej wymiarowości mamy na myśli to, że poszczególne porcje danych (np. rekordy w tabelach relacyjnych) posiadają dużą ilość atrybutów (zmiennych). W praktyce często okazuje się jednak, że wiele z tych atrybutów jest ze sobą dosyć mocno powiązanych (skorelowanych) i do otrzymania pełnego obrazu opisywanego zjawiska, czy zauważenia pewnych prawidłowości w danych, wystarczy uwzględnić jedynie niewielki ich podzbiór. Wychwycenie takich prawidłowości przy analizowaniu wszystkich (jak wspomniano wyżej często skorelowanych ze sobą) atrybutów jest zwykle albo niemożliwe, albo bardzo trudne do uzyskania. Spośród wielu zadań, które definiuje się w dziedzinie eksploracji danych (ang. data mining) duże znaczenie mają więc te, które pozwalają zredukować wymiarowość zbioru danych.

Przedstawione w artykule metody należą do grupy metod *nienadzorowanych*. Oznacza to, że wiedza (ang. knowledge), którą uzyskujemy analizując dane, powstaje niejako samorzutnie. Jest tak, gdyż nie dysponujemy żadnymi danymi wzorcowymi (treningowymi), które mogłyby posłużyć do wyciągnięcia wniosków dotyczących analizowanych danych. Klasycznym przykładem metody *nienadzorowanej* jest grupowanie danych. Poprzez analizę metodami statystycznymi zależności pomiędzy danymi, wyciągać można wnioski co do podobieństwa pewnych grup danych. Z kolei klasycznym przykładem metody *nadzorowanej* jest klasyfikacja. Dysponując wzorcowymi danymi, o których wiemy do jakiej grupy należą (czyli jak są zaklasyfikowane), można stosunkowo prosto i wiarygodnie wyciągnąć wnioski co do klasyfikacji pozostałych, niewzorcowych danych.

W pracy przedstawiono jedynie podstawowe metody związane z redukcją wymiarowości. Oprócz nich w literaturze można spotkać bardzo dużo innych metod, choć wiele z nich jest w gruncie rzeczy tylko modyfikacjami innych metod. Niektóre metody są oparte na różnych, często zaskakujących pomysłach. Jedną z nich (tzw. twarze Chernoff'a), która jest bardzo pomysłowa a zarazem zabawna, przedstawiono w jednym z kolejnych rozdziałów.

Artykuł ma dwa cele. Pierwszym jest pokazanie, jak na bazie metod wywodzących się ze statystyki, dokonywać redukcji wymiarowości danych. Przy czym redukcja ta nie powoduje zbyt wielkich strat informacji a czasami wręcz zwiększa wartość informacyjną danych. Drugim celem jest zwrócenie uwagi na istniejące w systemie Oracle stosunkowo proste w użyciu rozwiązanie, umożliwiające implementację tych metod w języku PL/SQL.

2. Program Oracle Data Miner

Program Oracle Data Miner jest w rzeczywistości graficzną nakładką ułatwiającą korzystanie z funkcjonalności systemu Oracle związanego z eksploracją danych i znanego pod nazwą Oracle Data Mining (ODM). ODM jest tzw. komponentem systemu Oracle, na który składają się głównie dwa specjalizowane pakiety w wersji PL/SQL (DBMS_DATA_MINING) oraz Java (Oracle Data Mining Java API oparte na standardzie JDM Java API [20]) plus zdefiniowane stosowne uprawnienia, role oraz schematy systemowe umożliwiające korzystanie z całości[1]. Wersja PL/SQL interfejsu ODM jest według firmy Oracle zalecanym interfejsem programistycznym. Należy jednak

[1] W wersjach Oracle wcześniejszych niż 11g Release 1 (11.1) istniał schemat DMSYS, w którym domyślnie przechowywane były wszystkie obiekty tworzone przez ODM. W wersji 11.1 schemat ten już nie istnieje i obiekty ODM przechowywane są w schemacie SYS.

zauważyć, że interfejs Java jest funkcjonalnie prawie w pełni kompatybilny z interfejsem PL/SQL-owym.

Dostępne interfejsy programistyczne w praktyce wykorzystywane są do osadzania wybranej funkcjonalności eksploracyjnej we własnych aplikacjach bazodanowych. Użytkownicy ukierunkowani jedynie na analityczne wykorzystanie posiadanych w bazie danych z pewnością sięgną po nakładkę ODM i często na jej możliwościach poprzestaną.

Tematyka ODM była już wielokrotnie omawiana w wielu publikacjach [21] lub przedstawiana w postaci tematycznych szkoleń [22]. Poniżej omówiona zostanie jedynie funkcjonalność systemu wiążąca się z tematyką artykułu, czyli redukcją wymiarowości danych poddawanych eksploracji.

2.1. Wartości puste oraz wartości oddalone

Obsługa wartości pustych (ang. missing values, null values) oraz oddalonych (ang. outliers) jest niezmiernie ważna w procesie redukcji wymiarowości[2]. Wiele z metod redukcji wymiarowości albo w ogóle nie zadziała w obecności pustych wartości albo otrzymane wyniki będą zniekształcone poprzez istnieje danych oddalonych [1][2]. Nieuwzględnienie z kolei w analizie eksploracyjnej części danych (odrzucenie wierszy ze „złymi" danymi) jest ze statystycznego punktu widzenia dyskusyjne.

W systemie ODM do obsługi wspomnianych przypadków stosowane są bardzo proste metody. Wartości oddalone poddawane są prostej procedurze statystycznej zwanej *winsorising* lub trywialnej metodzie zwanej *trimming* [23], polegającej na prostym odrzuceniu wartości oddalonych. Jeśli chodzi o obsługę wartości brakujących, to ogólną zasadą jest ich zastępowanie wartościami średnimi (dla danych numerycznych) lub modą (dla danych kategorycznych). O wiele jednak lepsze wyniki [2] uzyskać można stosując bardziej zaawansowane metody bazujące na analizie statystycznej tej części danych, w których nie występują wartości brakujące lub oddalone. Nie są one jednak zaimplementowane w systemie ODM. Jedną z takich metod jest metoda wielokrotnego wstawiania (ang. multiple imputation) [3].

2.2. Odkrywanie cech

Odkrywanie cech (ang. Feature Extraction, FE)[3] jest techniką określania nowego zbioru cech (ang. features) analizowanej rzeczywistości na potrzeby budowy modelu tejże rzeczywistości. Znalezione cechy są w pewnym sensie nowymi abstrakcyjnymi źródłami danych. Są one jednocześnie pewnymi kombinacjami (często liniowymi) oryginalnych atrybutów zbioru danych. Jest to więc z pewnością metoda redukująca wymiarowość pierwotnego zadania. Metoda FE określa również (w postaci wartości liczbowej) w jakim stopniu oryginalne atrybuty wpływają na wyliczoną cechę, czyli jaka jest ich wartość informacyjna.

Nowe cechy z reguły nie maja jasnej interpretacji fizycznej. Są jednak użyteczne przy wizualizacji zbioru danych na przykład w 2. lub 3. wymiarach. Często metoda FE używana jest również jako wstępna obróbka danych na potrzeby innych zadań eksploracyjnych. Z pewnością jednak największą zaletą metody FE jest możliwość interpretacji zadaniowej określonych cech. Możliwości interpretacyjne są już cechą systemu ODM a nie metody FE jako takiej. Przykładowo łatwo przypisać wyliczone cechy poszczególnym przypadkom (wierszom) analizowanego zbioru danych. Uzyskujemy więc pewien podział pierwotnych danych na grupy (klastry). Inny przykład to możli-

[2] Do wersji ODM 11.1 użytkownik miał możliwość wpływania na przygotowanie danych (ang. data preparation) poddawanych następnie eksploracji. Obsługa wartości pustych i oddalonych jest częścią tego przygotowania. Od wersji 11.1 czynności te wykonywane są już automatycznie i łatwo je „przeoczyć".

[3] Inne używane nazwy to variable selection, feature reduction, attribute selection, variable subset selection. W różnych opracowaniach na ten temat zauważa się dużą dowolność w stosowanej terminologii.

wość filtracji atrybutów. Możemy wtedy analizować znalezione cechy jedynie poprzez wartości pozostałych po filtracji atrybutów.

2.3. Selekcja cech

Selekcja cech (ang. Feature Selection, FS)[4] jest techniką określania istotności atrybutów pierwotnego zbioru danych (tak zwane atrybuty predykcyjne; ang. predictive attributes) na potrzeby przewidywania wartości wybranego atrybutu celu (ang. target attribute). Chodzi więc o wybór z dużej liczby atrybutów tych z nich, które posiadają istotną dla danego zadania (tu: predykcji) wartość. Przykładowo interesuje nas, które z dostępnych w pierwotnym zbiorze danych atrybutów opisujących klienta firmy najbardziej wpływają na niską aktywność klienta w korzystaniu z nowych usług firmy.

W metodzie FS istotny jest fakt, że uzyskane wyniki silnie zależą od wyboru atrybutów predykcyjnych. W praktyce nie zawsze będziemy wybierali wszystkie dostępne atrybuty, pomijając z oczywistych względów atrybuty niemogące wpływać na atrybut celu (np. kolor oczu nie ma wpływu na inteligencję) lub świadomie rezygnując z uwzględniania któryś z nich (np. płeć). Ponadto dodanie do zbioru danych nowego atrybutu predykcyjnego (np. wcześniej nieuwzględnionego w analizie) może w jego obecności zupełnie zmienić wcześniej wyliczone ważności atrybutów.

Dydaktyczne przykłady odkrywania i selekcji cech dla przykładowych zbiorów danych znaleźć można w scenariuszach ćwiczeń laboratoryjnych zamieszczonych w [7].

3. Metody statystyczne

3.1. Analiza składowych głównych

Wielowymiarowe dane z reguły nie są równomiernie rozrzucone wzdłuż wszystkich kierunków układu współrzędnych, ale koncentrują się w pewnych podprzestrzeniach oryginalnej przestrzeni. Celem analizy składowych głównych (ang. Principal Component Analysis, PCA) jest znalezienie tych podprzestrzeni w postaci tzw. składników głównych (zwanych czasami kierunkami). Są to wektory, które pełnią rolę nowych współrzędnych analizowanych danych wielowymiarowych. Składników głównych jest zdefiniowanych tyle, ile wymiarów pierwotnych danych.

Analiza składowych głównych oparta jest na wykorzystaniu podstawowych w statystyce pojęć, jakimi są m.in. korelacja i wariancja [4]. Pojęcia te wraz z wybranymi elementami algebry liniowej tworzą matematyczną całość służącą do analizy danych wielowymiarowych [5]. Pojęcia powyższe nie będą dokładniej omawiane w pracy (praktycznie każdy podręcznik statystyki je wyjaśnia), niemniej wiele z nich pojawi się niejako samoistnie w przykładach. W literaturze statystycznej metoda PCA klasyfikowana jest jako *eksploracyjna analizy danych*.

Składnikami głównymi okazują się być (stosunkowo łatwe do wyliczenia) wektory własne tzw. macierzy kowariancji zbioru danych. Z wektorami własnymi związane są odpowiadające im wartości własne, które są (dla macierzy kowariancji) dodatnimi liczbami rzeczywistymi. Wybierając wektory odpowiadające kilku największym wartościom własnym, otrzymujemy poszukiwany zbiór nowych kierunków układu współrzędnych. Kierunki te (co jest istotą metody) są kierunkami maksymalizującymi zmienność danych w sensie wariancji. Kierunki te są ze sobą nieskorelowane (używa się też algebraicznego pojęcia ortogonalne). Przyjmuje się oczywiście, że wszystkie atrybuty pierwotnego zbioru danych opisują 100% zmienności tych danych. Załóżmy, że analizujemy dane 12. wymiarowe. Gdy na przykład pierwsze trzy składowe główne tych danych „wyjaśniają" 85% ich pierwotnej zmienności, założyć można, że przedstawienie danych w 3. wymiarach będzie

[4] Inna używana nazwa to Attribute Importance. Wydaje się, że ten termin lepiej oddaje istotę metody.

wystarczające do uchwycenia zależności pomiędzy nimi. Zależnościami takimi mogą być przykładowo wyraźnie widoczne grupy, na jakie można podzielić dane.

Analiza składowych głównych jest metodą liniową. W metodzie zakłada się możliwość znalezienia nowych osi (kierunków), które lepiej „opisują" nasze dane na przykład w sensie ich logicznego pogrupowania (ang. clustering). Osie te są oczywiście liniami prostymi. Zdarzyć się jednak może, że układ analizowanych danych jest taki, że nie będzie możliwa sensowna redukcja danych do ich rzutów na wyliczone kierunki. Redukcja wymiaru oryginalnych danych wymagać będzie wtedy ich zrzutowania na krzywą, a nie na prostą. W takim przypadku zastosować można uogólnienia klasycznej metody PCA jakimi są np. tzw. jądrowa metoda PCA (ang. kernel PCA, kPCA) lub nieliniowa analiza PCA (ang. Non-linear PCA, nPCA) [24]. Temat ten nie będzie jednak w tym miejscu rozwijany i poprzestajemy jedynie na wspomnieniu o tej możliwości.

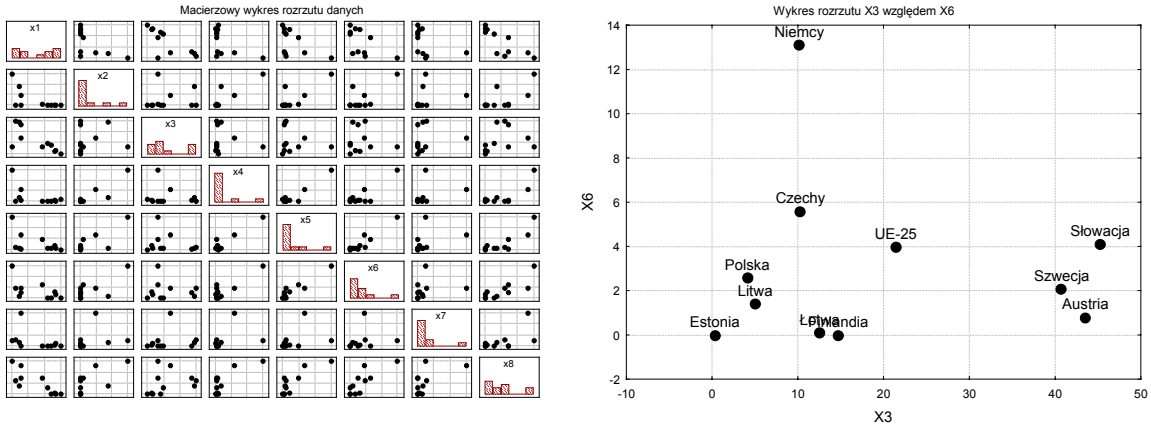
3.2. Przykład 1 – analiza składowych głównych (PCA)

Metoda PCA zilustrowana zostanie na przykładzie danych opisujących strukturę pozyskiwania energii ze źródeł odnawialnych w wybranych krajach UE w 2005 roku (tabela 1 oraz [8]). Chcemy sprawdzić, czy istnieją jakieś podobieństwa wśród uwzględnianych w zestawieniu krajów w zakresie produkcji energii ze źródeł odnawialnych. Analizując oryginalne dane opisane (aż) 8 atrybutami trudno, dostrzec zależności pomiędzy nimi. Wykonamy więc analizę PCA, która pokaże, że możliwa jest istotna redukcja ilości atrybutów, co z kolei umożliwi zauważenie niewidocznych na początku zależności pomiędzy danymi. W tabeli poszczególne źródła energii to: X1 – biomasa stała, X2 – energia promieniowania słonecznego, X3 – energia wody, X4 – energia wiatru, X5 – biogaz, X6 – biopaliwa, X7 – energia geotermalna, X8 – odpady komunalne. W ostatnim wierszu podano wartości wariancji dla każdego atrybutu. Duża zmienność ich wartości uzasadnia korzystanie z macierzy korelacji zamiast macierzy kowariancji. Innym powodem, kiedy uzasadnione jest użycie macierzy korelacji mogą być na przykład różne jednostki, w których podano analizowane dane.

Tab. 1. Struktura pozyskania energii w % wg. wybranych źródeł w wybranych krajach UE w 2005 roku

	X1	**X2**	**X3**	**X4**	**X5**	**X6**	**X7**	**X8**
UE-25	51,3	0,7	21,4	5,4	3,8	4,0	4,7	8,7
Austria	49,5	1,3	43,5	1,6	0,4	0,8	0,5	2,4
Czechy	76,4	0,1	10,2	0,1	2,8	5,6	0	4,8
Estonia	99,0	0	0,3	0,7	0	0	0	0
Finlandia	82,7	0	14,7	0,2	0,5	0	0	1,9
Litwa	92,9	0	5,0	0	0,3	1,4	0,4	0
Łotwa	86,9	0	12,5	0,2	0,3	0,1	0	0
Niemcy	41,3	2,2	10,1	14,0	8,6	13,1	0,8	9,9
Polska	91,2	0	4,1	0,3	1,2	2,6	0,2	0,4
Słowacja	45,1	0	45,2	0,1	0,6	4,1	0,9	4,0
Szwecja	51,7	0	40,7	0,5	0,2	2,1	0	4,8
Wariancja	485,0	0,5	274,0	18,0	6,7	14,5	1,9	12,2

Na początek przyjrzyjmy się paru wykresom bezpośrednio ilustrującym powiązania pomiędzy atrybutami. Dla niedużych wymiarów zadania (tu: 8) metoda taka może być całkiem użyteczna. Na rysunku 1 pokazano macierzowy wykres rozrzutu danych (ang. scatter plot) z dodatkowymi histogramami wartości poszczególnych atrybutów na przekątnej macierzy. Prezentuje on zależności pomiędzy poszczególnymi atrybutami (według zasady „każdy z każdym").



Rys. 1. Macierzowy wykres rozrzutu danych zbioru z tabeli 1. Po prawej stronie pokazano powiększony fragment dotyczący atrybutów X3 oraz X6

Przy tej ilości atrybutów (w sumie niewielkiej) oraz ilości danych (również niewielkiej) stosunkowo łatwo dostrzec, które atrybuty i w jakim stopniu są ze sobą skorelowane. Przykładowo widać, że atrybuty X5 oraz X6 są ze sobą mocno i dodatnio skorelowane. Z kolej atrybuty X6 oraz X3 nie są skorelowane prawie wcale. Przy większej ilości atrybutów analiza wykresu rozrzutu jest trudna, mało czytelna i w praktyce niezbyt przydatna. Poniżej dane z tabeli 1 zostaną przeanalizowane z wykorzystaniem analitycznej metody PCA (a nie jak powyżej graficznej). Pozwoli ona bardziej elegancko uwidocznić występujące zależności miedzy danymi.

Przechodząc do rozwiązania czysto analitycznego metodą PCA, w pierwszym kroku obliczymy macierz korelacji opisującą stopień zależności poszczególnych atrybutów od siebie (tabela 2). Im większa bezwzględna wartość elementu, tym większa zależność (korelacja) pomiędzy poszczególnymi atrybutami. Macierz jest symetryczna, dlatego dla czytelności pominięto elementy nad główną przekątną. Przykładowo pomiędzy atrybutami X5 oraz X6 istnieje duża zależność. Pomiędzy atrybutami X3 oraz X6 zależności takiej prawie nie ma. Zauważmy również istnienie dodatnich i ujemnych korelacji pomiędzy zmiennymi.

Tab. 2. Macierz korelacji dla zbioru z tabeli 1

	X1	**X2**	**X3**	**X4**	**X5**	**X6**	**X7**	**X8**
X1	1,00							
X2	-0,63	1,00						
X3	-0,75	0,11	1,00					
X4	-0,55	0,89	-0,11	1,00				
X5	-0,49	0,80	-0,20	0,94	1,00			
X6	-0,57	0,72	**-0,06**	0,85	**0,94**	1,00		
X7	-0,43	0,28	0,14	0,37	0,37	0,22	1,00	
X8	-0,80	0,67	0,26	0,78	0,83	0,81	0,60	1,00

Następnie wyznaczmy wartości własne macierzy korelacji (tabela 3a). Są one miarą zmienności pierwotnych danych przedstawionych we współrzędnych składowych głównych. Wartości tych zmienności pokazano w drugim wierszu tabeli 3a. Wartości własne szeregujemy od największej do najmniejszej.

Tab. 3a. Wartości własne macierzy korelacji z tabeli 2 uszeregowane malejąco[5]

	PC1	**PC2**	**PC3**	**PC4**	**PC5**	**PC6**	**PC7**
Wartości własne	5,01	1,68	0,85	0,36	0,06	0,04	0,01
Zmienność (%)	62,6	20,95	10,58	4,51	0,75	0,51	0,08
Zmienność skumulowana (%)	**62,6**	**83,58**	94,16	98,67	99,42	99,92	100

[5] PC8 można było pominąć, gdyż już dla siedmiu składowych osiągnięto maksymalną zmienność 100%.

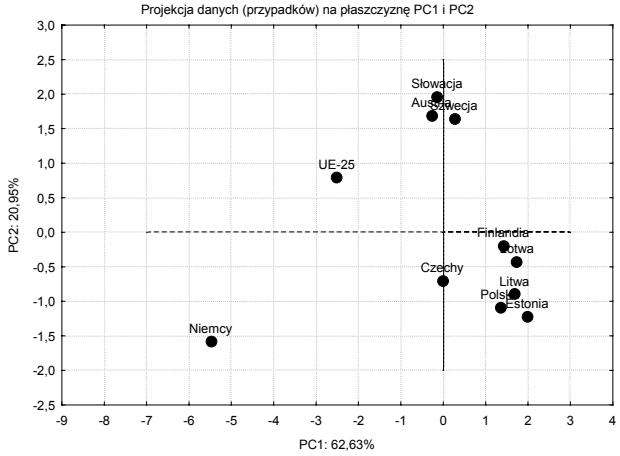
Widać, że dwie pierwsze wartości „przenoszą" ponad 80% zmienności pierwotnych danych. Możemy więc z dobrym przybliżeniem analizować pierwotny zbiór danych jedynie w dwóch wymiarach. Pozwala to prezentować dane w wygodnym do wzrokowej analizy układzie 2. wymiarowym.

Wektory własne (tabela 3b) pozwalają wyliczyć poszczególne składowe główne PC1-PC7. Przykładowe równanie pierwszej składowej głównej: PC1 = 0,34*X1 - 0,39*X2 - 0,07*X3 - 0,41*X4 - 0,41*X5 - 0,40*X6 - 0,23*X7 - 0,42*X8.

Tab. 3b. Wektory własne macierzy korelacji z tabeli 2

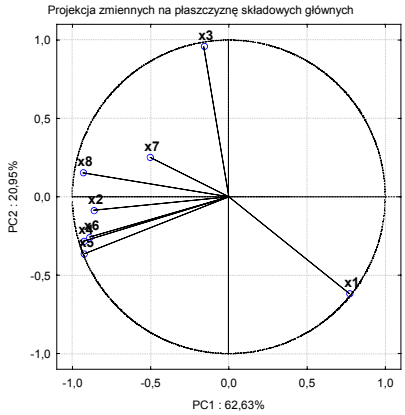
	PC1	PC2	PC3	PC4	PC5	PC6	PC7
X1	0,34	-0,48	-0,16	0,06	0,04	-0,14	0,09
X2	-0,39	-0,07	0,25	0,71	-0,13	-0,47	0,20
X3	-0,07	0,74	0,25	0,02	-0,09	0,09	-0,17
X4	-0,41	-0,22	0,05	0,27	0,32	0,76	0,07
X5	-0,41	-0,28	0,00	-0,14	-0,03	-0,18	-0,83
X6	-0,40	-0,20	0,20	-0,48	-0,61	0,08	0,37
X7	-0,23	0,20	-0,89	0,17	-0,30	0,04	0,03
X8	-0,42	0,12	-0,16	-0,38	0,64	-0,37	0,30

Po odpowiednim zrzutowaniu danych z tabeli 1 na nowe osie (tu: dwie pierwsze składowe główne), otrzymujemy wykres jak na rysunku 2. Wyraźnie widać, które kraje są do siebie podobne, które wyraźnie odróżniają się od siebie oraz jak mają się one do średnich wartości całej Unii Europejskiej (UE-25).



Rys. 2. Wykres obserwacji. Dane o produkcji energii ze źródeł odnawialnych z tabeli 1 zrzutowane na dwie pierwsze składowe główne

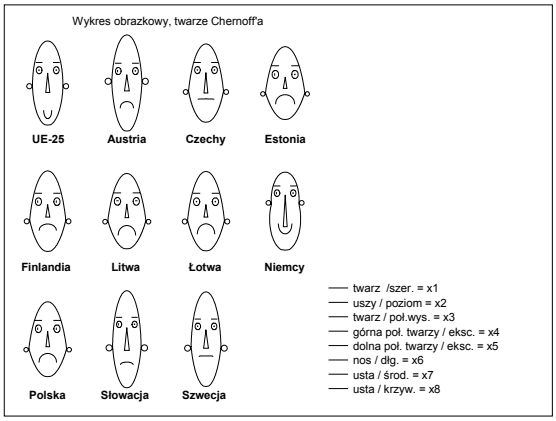
Możliwe jest również uwidocznienie wpływu zmiennych (atrybutów) X1-X8 na poszczególne składowe główne. Każda z tych zmiennych reprezentowana jest na rysunku 3 w postaci wektora. Kierunek i długość wektora określa, w jakim stopniu każda ze zmiennych wpływa na poszczególne składowe główne. Wektory takie nazywane są wektorami ładunków (ang. loadings). Gdy dwie zmienne leżą na wykresie blisko siebie oznacza to, że są one ze sobą silnie dodatnio skorelowane (np. zmienne X4 i X5). Gdy są do siebie prostopadłe, oznacza to brak skorelowania (np. X2 i X3). Gdy znajdują się po przeciwnych stronach, są silnie ujemnie skorelowane (np. X1 i X3).



Rys. 3. Wykres zmiennych. Położenie wektorów ładunków względem dwóch pierwszych składowych głównych

3.3. Twarze Chernoff'a

Rozwiązanie problemu wizualizacji danych wielowymiarowych doprowadzało czasami do bardzo ciekawych i zabawnych pomysłów. Przykładem są tzw. twarze Chernoff'a pokazane na rysunku 4. Podobieństwo lub niepodobieństwo danych wielowymiarowych odzwierciedlono tutaj w mimice rysunków twarzy. Legenda na rysunku 4 pokazuje, które elementy twarzy odpowiadają którym atrybutom zbioru danych z tabeli 1. Zachęcamy do porównania tego rysunku z wynikami rozwiązania zadania otrzymanymi za pomocą metody PCA omówionej w poprzednik podrozdziale. Z rysunku wynika, że Niemcy, które są niepodobne do innych krajów (w sensie danych z tabeli 1) reprezentowane są przez twarz o zupełnie innej mimice niż pozostałe. Twarz ta jest, jeżeli można tak powiedzieć, najbardziej optymistyczna i zadowolona. Podobny stopień zadowolenia widać jeszcze tylko na twarzy reprezentującej Unię Europejską ☺.



Rys. 4. Wykres obrazkowy w postaci tzw. twarzy Chernoff'a umożliwiający analizę danych wielowymiarowych

3.4. Analiza czynnikowa

Analiza czynnikowa (ang. Factor Analysis, FA)[6] sprowadza się do obliczenia wybranej z góry liczby tzw. ukrytych czynników (cech, zmiennych, nowych zmiennych), które „rządzą" danymi. Jest metodą badania struktury wewnętrznych zależności obserwacji wielowymiarowych. Celem

[6] Metoda FA często nazywana jest (przez analogie do nazwy PCA) analizą *czynników głównych* (ang. Principal Factors Analysis, PFA).

analizy czynnikowej jest znalezienie takiego zbioru czynników wspólnych oraz określenie ich relacji z pierwotnymi zmiennymi, które pozwalają na wyjaśnienie struktury powiązań pomiędzy tym zmiennymi. W literaturze statystycznej metoda FA klasyfikowana jest jako *modelowa analizy danych* (w tym sensie, że zakłada się budowę analitycznego *modelu* analizy czynnikowej).

Analiza czynnikowa, choć podobna, nie jest jednak tym samym, co omówiona wcześniej analiza składowych głównych PCA. Obie metody (PCA i FA) mają na celu w pewnym sensie redukcję wymiarowości zbioru danych[7]. Inaczej jednak rozwiązują to zadanie. W przypadku metody PCA nie można mówić o żadnym tworzonym przez nią modelu zbioru danych. Wyliczane są składowe główne a użytkownik podejmuje decyzję z ilu z nich skorzystać. W przypadku metody FA zmiana ilości czynników (zakłada się ich ilość przed analizą) skutkuje koniecznością powtórzenia wszystkich obliczeń, czyli znalezieniem zupełnie nowego rozwiązania. Rozwiązanie metodą PCA jest zawsze jednoznaczne, w przeciwieństwie do rozwiązania metodą FA.

W metodzie FA ważne jest również pojęcie tzw. *ładunków czynnikowych* (ang. factor loadings) oraz tzw. *rotacji* (ang. rotation). W największym uproszczeniu, ładunki czynnikowe to korelacje (czyli wartości liczbowe, dodatnie lub ujemne) pomiędzy zmiennymi oryginalnymi a wyodrębnionymi w wyniku analizy FA czynnikami. Podstawowym celem rotacji jest natomiast uzyskanie jak najprostszej interpretacji poszczególnych czynników (gdyż sama analiza czynnikowa prowadzi do niejednoznacznych rozwiązań. Jest to klasyczny „problem" omawiany w literaturze). Dodajmy jeszcze, że obroty mogą być ortogonalne (zachowują „prostopadłość osi czynnikowych") oraz ukośne (wspomnianej prostopadłości nie zachowują) oraz że istnieje wiele wariantów (strategii) takich obrotów. W literaturze pojawiają się one pod takimi nazwami jak *quartimax, varimax, equamax*.

Podkreślmy na koniec, że metody PCA oraz FA często dają bardzo podobne wyniki. W praktyce zwykle wykonuje się obie analizy dla tego samego zbioru danych i porównuje wyniki. Inne podejście zakłada, że analiza składowych głównych będzie preferowana jako metoda *redukcji danych*, podczas gdy analiza czynnikowa jest chętniej stosowana, gdy celem jest wykrycie *struktury danych* (np. zadanie klasyfikacji).

3.5. Przykład 2 – analiza czynnikowa (FA)

Rozważmy ponownie dane z tabeli 1. Zbadamy wewnętrzne zależności jakie zachodzą w przykładowym zbiorze danych. Zgodnie z istotą metody FA, zależności te są wywoływane przez pewne czynniki, które nie są bezpośrednio obserwowalne. Dla analizowanych danych znajdziemy te czynniki i skorzystamy z nich w celu wizualizacji zbioru danych.

Spośród wielu znanych metod znajdowania czynników wspólnych, najpopularniejsze to metoda oparta o składowe główne (porównaj poprzedni rozdział), metoda cetroidalna oraz metoda największej wiarygodności (ang. Maximum Likelihood Factors). W przykładzie użyta zostanie ta ostatnia. Zakładamy, że chcemy znaleźć dwa czynniki wspólne (w metodzie FA należy na początku określić tą ilość). Wyniki pokazane zostaną w wersji bez oraz z rotacją czynników. Rotacja czynników jest jedną z ważniejszych cech analizy czynnikowej. Spośród wielu znanych metod rotacji, użyta zostanie jedna przykładowa. Wyniki obliczeń, dla założonej liczby dwóch czynników (czyli dwóch), przedstawiono w tabelach poniżej.

[7] Pojęcie analiza czynnikowa używane jest również w szerszym sensie, jako rodzina metod i procedur statystycznych pozwalających na sprowadzenie dużej liczby badanych zmiennych do znacznie mniejszej liczby wzajemnie niezależnych (nieskorelowanych) czynników. Mówi się wtedy o czynnikowych technikach analitycznych analizy danych wielowymiarowych. Metody PCA i FA należą wtedy do tej rodziny.

Tabela 3. Wartości własne oraz procent wyjaśnianej wariancji

No.	Wartość własna	% ogółu	Skumulowana wartość własna	Skumulowana wariancja
1	3,22	40,30	3,22	40,30
2	3,36	41,95	6,58	82,25

Tabela 4. Ładunki czynnikowe (bez rotacji), oznaczone ładunki >0,7

	Czynnik 1	Czynnik 2
X1	**0,98**	0,11
X2	-0,56	-0,62
X3	**-0,81**	0,57
X4	-0,45	**-0,84**
X5	-0,39	**-0,91**
X6	-0,48	**-0,78**
X7	-0,39	-0,30
X8	**-0,73**	-0,60
Wartość wyjaśniana	3,22	3,36
Udział	0,40	0,42

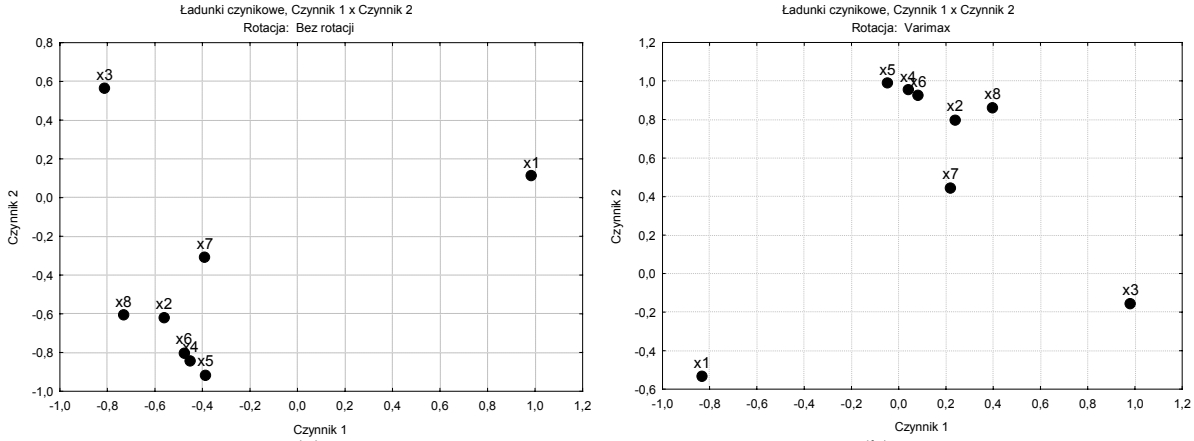
Wyznaczenie większej liczby czynników (nie pokazano tego w tabeli 4) nie jest uzasadnione. Dla czynników od trzeciego włącznie nie ma ładunków o wartościach większych niż 0,7.

Tabela 5. Ładunki czynnikowe (z rotacją typu Varimax), oznaczone ładunki >0,7

	Czynnik 1	Czynnik 2
X1	**-0,84**	-0,53
X2	0,23	**0,80**
X3	**0,98**	-0,15
X4	0,04	**0,96**
X5	-0,05	**0,99**
X6	0,08	**0,93**
X7	0,22	0,45
X8	0,40	**0,87**
Wartość wyjaśniana	1,937	4,65
Udział	0,24	0,58

Po rotacji widać, że obecnie tylko jedna (poprzednio dwie) zmienna nie jest „obsłużona" na założonym poziomie >0,7. Widać też, że obecnie bardzo zwiększył się udział czynnika 2 (poprzednio oba były mniej więcej równe).

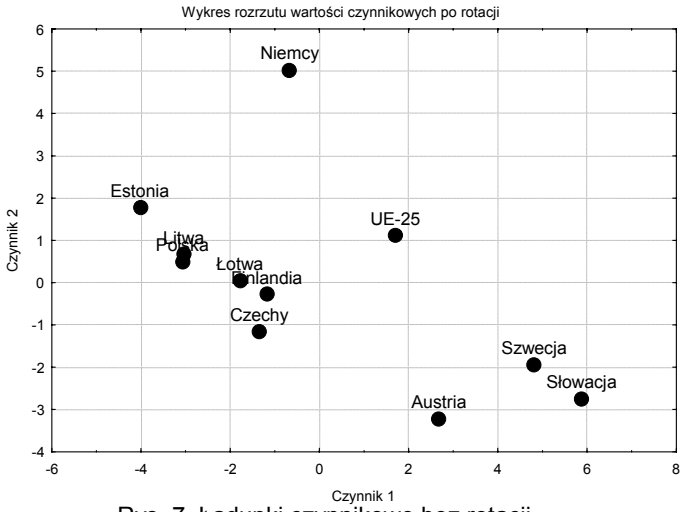
Na rysunkach 5a oraz 5b poszczególne atrybuty (X1-X8) przedstawione są jako punkty we współrzędnych wyznaczonych przez ładunki czynnikowe przed rotacją i po niej. W tabeli 6 pokazano wartości czynnikowe (ang. factor scores) po rotacji dla każdego wiersza danych oraz umieszczono te dane na wykresie rozrzutu na rysunku 7. Czytelnikowi pozostawiamy porównanie otrzymanych wyników z wynikami otrzymanymi metodą PCA.



Rys. 5. (a) – ładunki czynnikowe bez rotacji, (b) – ładunki czynnikowe z rotacją

Tabela 6. Wartości czynnikowe

	Czynnik 1	**Czynnik 2**
UE-25	1,705	1,127
Austria	2,673	-3,215
Czechy	-1,368	-1,148
Estonia	-4,024	1,773
Finlandia	-1,183	-0,268
Litwa	-3,038	0,675
Łotwa	-1,787	0,064
Niemcy	-0,677	5,039
Polska	-3,068	0,502
Słowacja	5,877	-2,749
Szwecja	4,804	-1,927



Rys. 7. Ładunki czynnikowe bez rotacji

3.6. Skalowanie wielowymiarowe

Skalowanie wielowymiarowe[8] (ang. Multidimensional Scaling, MDS) to metoda wizualizacji danych w niskim wymiarze oparta na wykorzystaniu tzw. macierzy bliskości (ang. proximity matrix) oraz jakiejś odpowiedniej do analizowanych danych metryki (miary odległości). Jest to podstawowa metoda[9], mająca jednak wiele modyfikacji (np. omawiane poniżej metody SM, RPM i wiele innych). Skalowanie wielowymiarowe dąży do rozmieszczenia obiektów jako punktów w przestrzeni niskowymiarowej w taki sposób, aby obiekty podobne do siebie (w oryginalnym zbiorze danych) znajdowały się blisko siebie. Redukujemy więc oryginalny rozmiar danych z zachowaniem zarówno ich własności topologicznych jak i metrycznych.

Skalowanie wielowymiarowe wymaga posiadania informacji o „bliskościach" pomiędzy elementami zbioru danych. Stosowane do tego celu struktury danych to np. macierze podobieństwa, odmienności, odległości pomiędzy obiektami. W niektórych zastosowaniach informacja taka jest naturalna (np. porównanie przez respondentów jakości dwóch produktów). W innych macierz bliskości należy najpierw wyliczyć i nie zawsze będzie to proste zadanie. Zauważmy, że o np. odmienności możemy mówić również w kontekście danych nie tylko ilościowych ale i jakościowych. Wizualizacja metodami MDS może dotyczyć również danych mieszanych, które w postaci oryginalnej nie maja żadnej czytelnej interpretacji geometrycznej (jak na wykresie pokazać bliskość osób o różnych kolorach oczu?).

W sensie obliczeniowym skalowanie wielowymiarowe jest nie tyle ścisłą procedurą, ile raczej sposobem „zmiany rozmieszczenia" obiektów w sposób na tyle efektywny, aby otrzymać konfigurację, która jest najlepszym przybliżeniem oryginalnych (czasami mówi się obserwowanych) odległości. Metoda przemieszcza obiekty w przestrzeni zdefiniowanej przez pożądaną liczbę wymiarów i sprawdza, na ile ta nowa konfiguracja odtwarza odległości między obiektami. Matematycznie chodzi o minimalizację pewnej funkcji, która jest miarą jakości rozwiązania zadania.

Oznaczmy przez N_{ij} odległość pomiędzy i-tym i j-tym obiektem w oryginalnej przestrzeni wielowymiarowej. Analogicznie przez n_{ij} oznaczmy odległość pomiędzy i-tym i j-tym obiektem w przestrzeni o zredukowanej ilości wymiarów (w praktyce chodzi o płaszczyznę)[10]. MDS sprowadza się teraz do minimalizacji tzw. funkcji błędu (w literaturze anglojęzycznej zwanej stress function).

$$E = \sum_{i<j} (N_{ij} - n_{ij})^2$$

która jest miarą stosowaną do szacowania, na ile dobrze (lub źle) dana konfiguracja odtwarza macierz odległości.

3.7. Odwzorowanie Sammon'a

Odwzorowanie Sammon'a (ang. Sammon Mapping, SM) to popularna nieliniowa metoda wizualizacji danych wielowymiarowych na (z reguły) płaszczyźnie. Używając tzw. gradienowych[11] metod optymalizacji, minimalizuje pewną funkcję opisującą odległości pomiędzy danymi. Pomysł

[8] Tak jak miało to miejsce wcześniej, zwracamy uwagę, że termin MDS jest często używany do określenia rodziny metod służących wizualizacji danych na bazie (jakiejś) macierzy bliskości. Cele metod są takie same. Różnice występują w sposobach (algorytmach) osiągnięcia tegoż celu.

[9] Inna stosowana w literaturze nazwa może być trochę myląca: Analiza współrzędnych głównych (ang. Principal Coordinates Analysis).

[10] Zakładamy oczywiście, że odległości te da się określić w odpowiedniej dla analizowanych danych metryce.

[11] Jak pamiętamy, gradient to kierunek najszybszego wzrostu funkcji w danym punkcie. Oczywiście istotna jest tu minimalizacja funkcji jako taka a nie metoda „otrzymania" tej minimalizacji.

jest podobny do zastosowanego przy skalowaniu wielowymiarowym i wygląda następująco: Oznaczmy przez N_{ij} odległość pomiędzy *i*-tym i *j*-tym obiektem w oryginalnej przestrzeni wielowymiarowej. Analogicznie przez n_{ij} oznaczmy odległość pomiędzy *i*-tym i *j*-tym obiektem w przestrzeni o zredukowanej ilości wymiarów. SM sprowadza się teraz do minimalizacji funkcji błędu.

$$E = \sum_{i<j} \frac{(N_{ij} - n_{ij})^2}{N_{ij}}$$

W wyniku, dzięki maksymalnemu zachowaniu „struktury odległości" pomiędzy punktami w obu przestrzeniach, wiele ważnych informacji o danych zastaje zachowanych i uwidocznionych na płaszczyźnie. Odwzorowanie Sammona jest metodą iteracyjną, stąd łatwo kontrolować wartość funkcji błędu, czyli w pewnym sensie jakość redukcji wymiarowości rozumiana jako zachowanie maksimum informacji drzemiących w pierwotnych danych. Element w mianowniku powyższego wzoru ma za zadanie zredukowanie pewnej niekorzystnej cechy, która występuje w metodzie MDS. Chodzi o to, aby punkty w oryginalnej przestrzeni, które są odległe od siebie, nie zdominowały za bardzo funkcji błędu, a przez to nie zniekształcały wyników analizy.

3.8. Przykład 3 – odwzorowanie Sammon'a (SM)

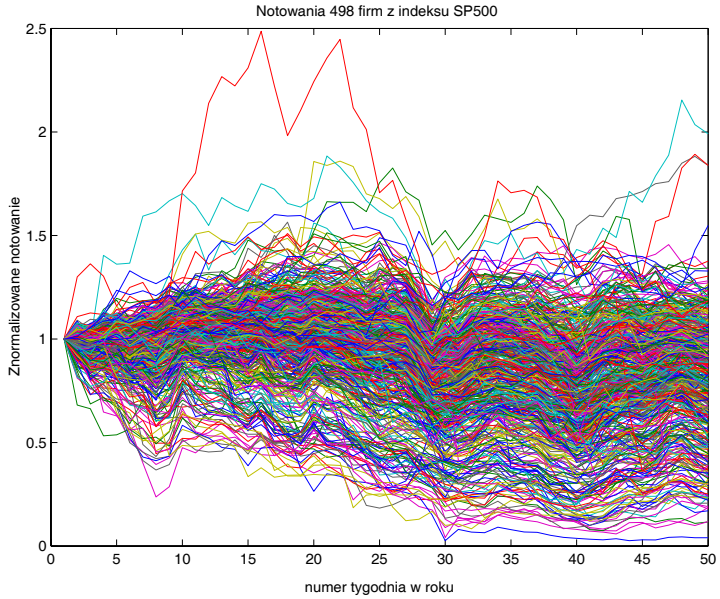
W obecnym przykładzie, w odróżnieniu od poprzednich bazujących na danych z tabeli 1, używany zbiór danych jest dużo większy. Zbiór ten (nazwijmy go SP500) zawiera notowania za okres jednego roku dla 498 firm tworzących indeks S&P 500 giełdy nowojorskiej. Notowania zostały znormalizowane w taki sposób, aby notowanie każdej firmy rozpoczynało się od wartości 1 w pierwszym tygodniu roku (atrybut A0). Każda firma opisana jest 52. atrybutami (od A0 do A51). Każdy atrybut to średnie notowanie danej firmy z jednego tygodnia. Mamy więc dane o wynikach firm z 52 tygodni, czyli z całego roku[12]. W tabeli 7 pokazano jedynie dane dla pierwszej firmy na liście (sortowanie alfabetyczne skrótów firm), ostatniej oraz ... naszej ulubionej ☺.

Tabela 7. Fragment notowań 498 firm z indeksu SP500 w okresie 52 tygodni

Id firmy	Skrót	A0	A1	A2	...	A50	A51
1	A	1	0.947	0.512	...	0.525	0.869
...	...	...	...	...	...	...	...
342	ORCL	1	1.053	1.067	...	0.73	0.689
...	...	...	...	...	...	...	...
498	ZMH	1	1.013	1.056	...	1.33	1.328

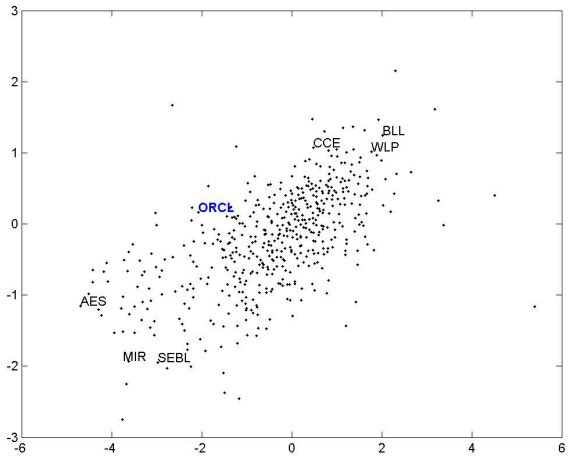
Zadaniem do wykonania jest sensowne pogrupowanie firm na przykład na te, których notowania w analizowanym okresie rosły, były w miarę równe oraz spadały. Oczywiście wzrokowa analiza przebiegów, z uwagi na dużą ilość danych, nie ma żadnego sensu, co widać na zbiorczym rysunku 9 przedstawiającym przebiegi notowań wszystkich firm.

[12] W rzeczywistości powinniśmy uwzględnić wszystkie dni notowań w roku (ok. 300) a nie tylko średnie tygodniowe, ale wtedy wymiar zadania wzrośnie do 300 !

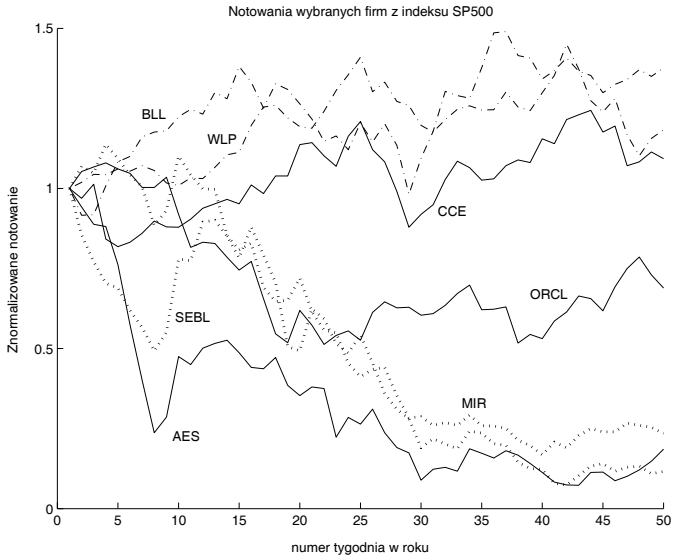


Rys. 9. Wykres notowań 498 firm z indeksu SP500 w okresie 52 tygodni

W przykładzie zastosowano odwzorowanie Sammon'a. SM jest procedurą iteracyjną. Iteracje zakończono przy wartości funkcji błędu odległości na poziomie 0,006 czyli bardzo małej. Otrzymane wynik pokazano na rysunku 10. Firmy (reprezentowane przez punkty) leżące na wykresie blisko siebie, mają podobne wykresy zmienności swoich notowań na przestrzeni roku. Zaznaczono na nim przykładowe dwie podobne firmy, których notowania rosną (BLL, WLP), dwie podobne firmy, których notowania maleją (MIR, SEBL), oraz dwie firmy zupełnie niepodobne do siebie (notowania jednej firmy rosną, drugiej maleją, AES, CCE). Zaznaczono również firmę Oracle. Na rysunku 11 wykreślono notowania firm wyróżnionych rysunku 10.



Rys. 10. Odwzorowanie Sammon'a 2D dla danych 52-wymiarowych z przykładu 3

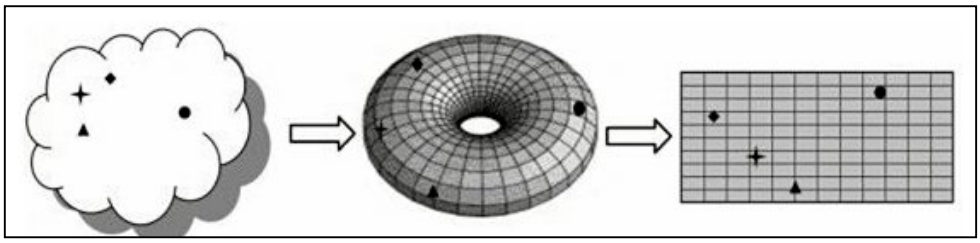


Rys. 11. Wykres notowań wybranych firm zaznaczonych na rysunku 10

3.9. Relacyjne mapy perspektyw

Relacyjne mapy perspektyw (ang, Relational Perspective Map, RPM) [6] są w pewnym sensie modyfikacją klasycznych metody mapowania danych wielowymiarowych, czyli na przykład omówionego wyżej odwzorowania Sammon'a. Najistotniejsza zmiana to fakt, że w wyniku otrzymujemy niskowymiarową reprezentację oryginalnych danych w postaci punktów, które są w miarę równomiernie (np. nie za gęsto jak w metodzie SM) rozmieszczone na płaszczyźnie. Unikamy w ten sposób ew. nakładania się punktów na siebie. W metodzie Sammon'a lub wcześniej opisanej metodzie MDS w ogólności nie musi tak być, co może prowadzić do gubienia pewnych informacji występujących w pierwotnym zbiorze danych. Zjawisko to będzie szczególnie widoczne, gdy w zbiorze danych znajdować będą się dane zarówno leżące daleko jak i blisko siebie.

W metodzie RPM punkty (dane) rzutowane są nie bezpośrednio na płaszczyznę, ale najpierw na torus (potocznie „dętka rowerowa") czyli *dwuwymiarową* powierzchnię leżącą w przestrzeni *trójwymiarowej*. W drugim kroku, dzięki cesze 2-wymiarowości jaką posiada torus, bryła jest poprzez poziome i pionowe cięcia[13] „rozwijana" do postaci prostokąta wraz z wcześniej zrzutowanymi na nią danymi.



Rys. 12. Fazy algorytmu RPM. Rzutowanie danych wielowymiarowych na torus a następnie rozwiniecie torusa na płaszczyznę

W metodzie RPM, funkcją analogiczną do pokazanych dla innych metod jest formuła zwana teraz funkcją energetyczną

[13] Zobacz dostępną w sieci animację: http://www.visumap.net/Resources/UnwrappingTorus.htm

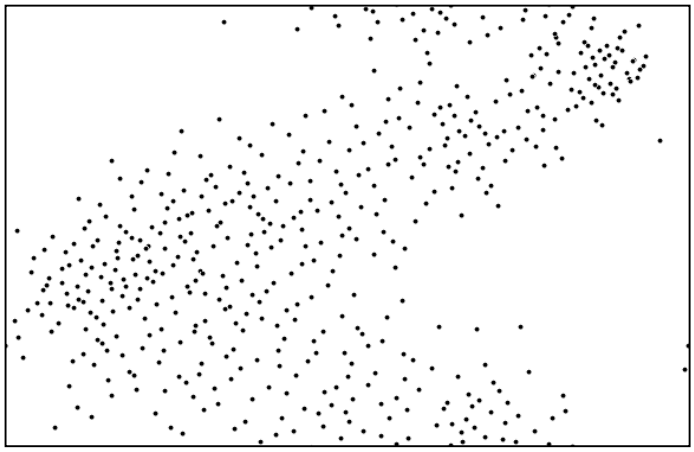
$$E(p) = \sum_{i<j} \frac{N_{ij}}{pn_{ij}^{p}}; E(0) = -\sum_{i<j} N_{ij} \ln(n_{ij})$$

(p – parametr algorytmu zwany sztywnością (ang. rigidity)). Odpowiednia metoda[14] znajduje takie rozmieszczenia punktów na torusie, które *minimalizują* tzw. całkowitą „energię układu". Matematycznie sprowadza się to do obliczenia pochodnej z powyższej funkcji (pochodna z energii to siła).

Pośredni etap (mapowanie na torus) jest ważny. Torus to bryła, po której punkty mogą swobodnie przesuwać się bez „spadnięcia" z niej. Algorytm może więc swobodnie nimi przesuwać w celu wspomnianej minimalizacji energetycznej. Z drugiej strony, jak również wspomniano, w końcowym kroku łatwo rozwinąć torus na regularnej (prostokąt) płaszczyźnie.

3.10. Przykład 4 – Relacyjne mapy perspektyw (RPM)

Poniżej tylko dla ilustracji i porównania z odwzorowaniem Sammon'a (rysunke 10) pokazano wynik uzyskany metodą RPM. Zwracamy uwagę na dużo bardzie równomierne rozmieszczenie obrazów punktów na mapie 2D. Jak już wspomniano przy opisie metody RPM nie ma tutaj niebezpieczeństwa, że punkty będą się pokrywały.



Rys. 13. Odwzorowanie RPM dla danych SP500

4. Mapy Kohonena

Kolejnym sposobem redukcji wymiarowości jest zastosowanie tzw. sieci Kohonena (czasami nazywane też mapami Kohonena) [15,16,17]. Sieci Kohonena nazwane zostały przez ich twórcę samoorganizującym odwzorowaniem (ang. Self-Organizing Maps – SOM) lub samoorganizującym odwzorowaniem cech (ang. Self-Organizing Feature Maps – SOFM). Ta pierwsza nazwa jest zdecydowanie częściej używana przez różnych autorów prac z tej dziedziny.

SOM są pewnym szczególnym rodzajem sieci neuronowych, gdzie uczenie odbywa się bez nauczyciela, tzn. użytkownik posiada jedynie dane wejściowe (uczące) [15], natomiast nie posiada żadnych wzorców wyjściowych. Zadaniem sieci w trakcie uczenia jest właśnie wytworzenie takich wzorców. Działanie sieci Kohonena opiera się na tzw. uczeniu konkurencyjnym (ang. competitive learning). Inaczej niż w klasycznych sieciach neuronowych, w trakcie uczenia adaptacji podlegają nie wszystkie neurony tworzące sieć, ale tylko jeden lub kilka wybranych neuronów. Po prezenta-

[14] W znanej implementacji [10] pierwotnie stosowana była klasyczna metoda optymalizacyjna Newtona. W najnowszej wersji stosowany jest algorytm genetyczny symulowanego wyżarzania.

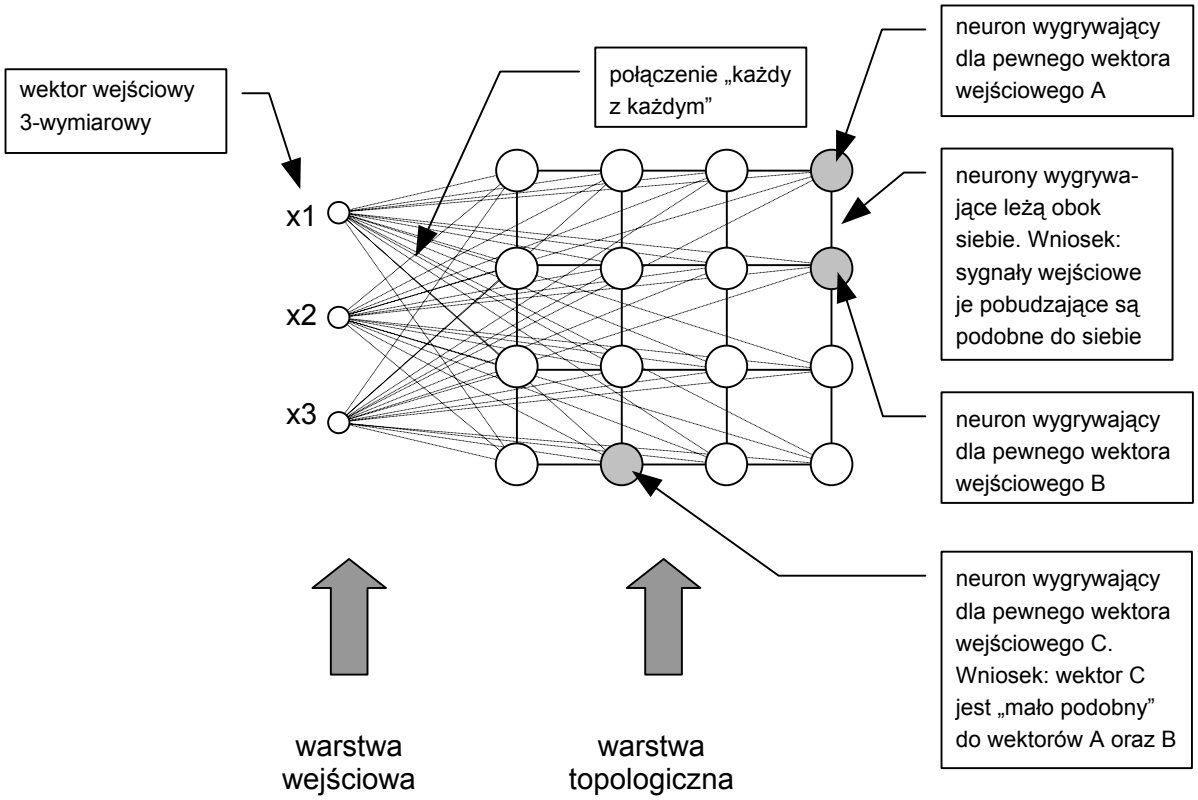
[15] W literaturze dotyczącej *stricte* sieci neuronowych zamiast pojęcia dane wejściowe częściej mówi się o sygnałach wejściowych. W gruncie rzeczy chodzi dokładnie o to samo, jest tylko różnica terminologiczna.

cji poszczególnych wzorców wejściowych, za pomocą odpowiedniego algorytmu, wybierany jest neuron zwycięzca i tylko on otrzymuje „przywilej" adaptacji swoich wag (w praktyce adaptacji podlegają wagi neuronu zwyciężającego oraz neuronów z jego sąsiedztwa). Zwycięski jest ten neuron, którego wektor wag jest najbliższy w stosunku do aktualnego wektora wejściowego. Algorytm wybory zwycięzcy należy do klasy zwanej popularnie „zwycięzca bierze wszystko" (ang. Winner Takes All – WTA).

Powyżej użyto sformułowania „sąsiedztwo neuronów". Pojęcie to jest kluczowe w sieciach SOM. Nie występuje ono w innych rodzajach sieci neuronowych. Okazuje się bowiem, że poszczególne neurony mają ściśle określonych „sąsiadów". Mamy więc do czynienia z rodzajem mapy neuronów, które tworzą swego rodzaju topologię. Obrazowo pokazano to na rysunku 14. Do warstwy wejściowej dopływają poszczególne dane wejściowe, które „pobudzają" tylko jeden neuron. Neuron ten zostaje zwycięzcą. Mówiąc o pobudzaniu mamy na myśli to, że na określone dane wejściowe najsilniej odpowiada jeden określony neuron. Prezentując wielokrotnie kolejne dane wejściowe, sieć SOM uczy się ich rozpoznawania. Każdorazowo wyłaniany jest neuron zwycięski. Najczęściej właśnie ten zwycięzca jest uczony (czyli modyfikowane są jego wagi). Po nauczeniu sieć Kohonena ma więc umiejętność rozpoznawania wzorców danych wejściowych oraz rozpoznawania wzorców tylko podobnych do danych wejściowych (ta ostatnia cecha, czyli umiejętność uogólniania zdobytej wiedzy, jest akurat wspólna dla praktycznie wszystkich rodzajów sieci neuronowych). Co więcej podobne sygnały rozpoznawane są przez neurony, które w sieci Kohonena znajdują się obok siebie! Warto również zwrócić uwagę na jeszcze jedną bardzo istotną właściwość sieci SOM. Jeżeli pewne dane wejściowe pojawiają się częściej niż inne, to będą częściej pobudzały wybrane neurony. Pewne neurony będą więc częściej zwyciężać niż inne. Z drugiej strony dane, które będą pojawiać się bardzo rzadko będą miały bardzo niewielkie szanse, na „wytrenowanie" swoich własnych neuronów. W ten sposób mapa Kohonena może odzwierciedlać nie tylko rozkład przestrzenny danych wejściowych, ale również częstotliwość ich występowania! Reasumując można więc powiedzieć, ze oprócz rozpoznawania podobnych sygnałów (co potrafią „klasyczne" sieci neuronowe) sieć Kohonena potrafi również te podobne sygnały ulokować na mapie obok siebie uwzględniając przy tym częstość ich pojawiania się.

Jak więc umiejscowić sieci Kohonena w kontekście redukcji wymiarowości? Należy zdać sobie sprawę, że wektor wejściowy pokazany na rysunku 14 może mieć praktycznie dowolny wymiar (na rysunku jest 3 wymiarowy). Mapa Kohonena jest natomiast obiektem dwuwymiarowym[16]. Mamy więc do czynienia z taką sytuacją, że wielowymiarowy wektor wejściowy zostaje „przekonwertowany" na postać 2. wymiarową, która jest już bardzo wygodna do wizualnej analizy. Sieć Kohonena dąży do tego, by stworzyć optymalną (a w praktyce prawie optymalną, gdyż trudno jest formalnie zdefiniować optymalność wynikowej mapy Kohonena) mapę obrazującą stosunki zachodzące w wielowymiarowej przestrzeni danych wejściowych. Ponadto sieci Kohonena mogą pełnić również funkcję klasyfikatora grupującego dane w tzw. klastry. Skoro podobne sygnały grupują się obok siebie na mapie, to gdy wektory wejściowe tworzą jakieś naturalne grupy, będą one widoczne na powstałej mapie jako łatwo zauważalne „wyspy".

[16] Czasami rozważa się mapy Kohonena 1. wymiarowe. Natomiast mapy więcej niż 2 wymiarowe, choć teoretycznie możliwe, nie są stosowane.



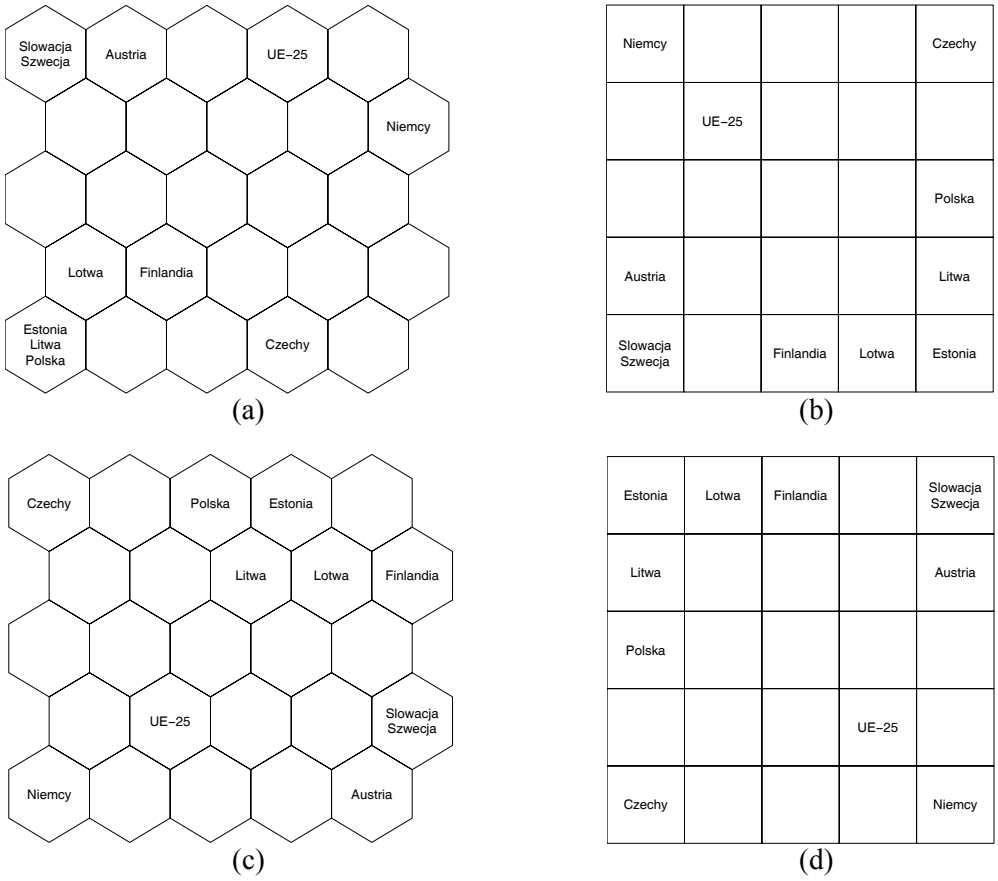
Rys. 14. Przykładowa mapa Kohonena. Wektor wejściowy jest 3 wymiarowy. Mapa ma postać siatki protokątnej o wymiarze 4 x 4.

Na rysunku 15 pokazano przykładowe wyniki uzyskany po zastosowaniu sieci Kohonena do danych z tabeli 1. Kwadraty i sześciokąty foremne[17] reprezentują poszczególne neurony tworzące mapę Kohonena. Widać, że sieć bardzo sprawnie pogrupowała poszczególne kraje. Powstałe grupy są w zasadzie takie same, jak te, które otrzymano wcześniej opisaną metodą PCA oraz FA.

Warto zwrócić uwagę na pewną niedogodność w stosowaniu sieci SOM. Na rysunku 15a oraz 15b widać generalnie podobnie pogrupowane kraje (np. Polska, Estonia, Litwa, Łotwa i Finlandia na obu rysunkach tworzą wyraźną grupę). Jednak na obu rysunkach poszczególne grupy znajdują się w innych obszarach mapy (lewy górny oraz prawy górny róg mapy). Jest to spowodowane tym, że oba rysunki powstały w wyniku dwóch oddzielnych trenowań sieci. Rozpoczynając uczenie sieci Kohonena musimy (najczęściej losowo) wybrać wagi początkowe neuronów. I praktycznie nie możemy nigdy przewidzieć, jak potoczy się proces uczenia. Nie mamy wpływu na to, które neurony, w których momentach uczenia będą stawały się neuronami wygrywającymi. Nie mamy również wpływu na to, które neurony będą pozostawały poza procesem wygrywania. W efekcie każdorazowe uruchomienie procedury uczącej da nieco inny graficzny wynik końcowy (w sensie wyglądu finalnej mapy). Widać to wyraźnie również na rysunkach 15c oraz 15d, gdzie kraje mają jeszcze inny układ, różny od układu z rysunków 15a oraz 15b. Oczywiście poprawnie nauczona sieć za każdym razem pokaże poprawne wyniki grupowania, jednak wizualnie mogą one często wyglądać inaczej. Tak jednak dzieje się z wszystkimi innymi sieciami neuronowymi, które naśladując ludzi mózg, tworzą szczegółowe wyniki za każdym razem nieco inne, choć co do pryncypiów takie same (dwaj malarze rysując ten sam przedmiot narysują go każdy nieco inaczej). Jest to w sumie bardziej zaleta niż wada sieci neuronowych, gdyż często interesuje nas otrzymanie wyni-

[17] Prezentowanie map Kohonena jako przylegające do siebie kwadraty lub sześciokąty foremne jest najczęściej spotykane w praktyce.

ku satysfakcjonującego praktycznie a niekoniecznie idealnego czy optymalnego z matematycznego punktu widzenia.



Rys. 15. Otrzymane mapy Kohonena dla przykładu z tabeli 1. Pokazano 2 typowo spotykane warianty mapy. (a) mapa, gdzie każdy neuron ma 6. sąsiadów (sześciokąt foremny), (b) mapa, gdzie każdy neuron ma 8. sąsiadów (kwadratowa), (c), (d) mapy uzyskane w wyniku powtórnego wykonania eksperymentu. Widać, że kraje zostały pogrupowane praktycznie identycznie ale układ graficzny jest nieco inny

Wspomnijmy jeszcze w tym miejscu o pewnej metodzie, która jest niejako rozwinięciem idei map Kohonena. Chodzi o tzw. generowane mapy topograficzne (ang. Generative Topographic Mapping, GTM) [24,25], które rozwiązują pewne problemy występujące w mapach Kohonena. Są to jednak bardzo szczegółowe (i dość trudne) zagadnienia, dlatego nie będą w tym miejscu omawiane. Zainteresowane osoby mogą łatwo sięgnąć do dostępnej literatury.

5. Implementacja metod w środowisku Oracle – pakiet UTL_NLA

Na koniec wspominamy o stosunkowo mało znanym pakiecie PL/SQL, domyślnie instalowanym w bazie Oracle począwszy od wersji 10g, mającym duże znaczenie z punktu widzenia ewentualnej implementacji omówionych w pracy metod redukcji wymiarowości. Definiowanie macierzy numerycznych w środowisku Oracle zawsze sprawiało użytkownikom problemy. Nie należy oczywiście utożsamiać tabeli relacyjnej i macierzy numerycznej. Oba obiekty są co prawda „tabelopodobne", jednak na tym podobieństwa się kończą. Macierze numeryczne wymagają implementacji efektywnych metod ich obsługi, jak np. mnożenie macierzy, obliczanie odwrotności macierzy, rozwiązywanie układów równań liniowych i bardzo wiele innych. Do wersji 10g serwera Oracle użytkownicy byli skazani na samodzielne implementowanie tych metod, co było zadaniem i trudnym, i bardzo pracochłonnym.

Wraz z wersją 10g release 2 pojawił się pakiet UTL_NLA [11], który udostępnia wszystkie potrzebne do implementacji algorytmów macierzowych mechanizmy definiowania struktur macierzowych w bazie danych wraz z wydajną implementacją podstawowych operacji macierzowo-wektorowych. Twórcy pakietu przenieśli do języka PL/SQL podstawową funkcjonalność doskonale znanych w środowisku numerycznym pakietów BLAS (ang. Basic Linear Algebra Subprograms) [12] oraz LAPACK (ang. Linear Algebra PACKage) [13]. Poniżej pokazano przykład użycia pakietu UTL_NLA do wykonania podstawowego zadania algebry liniowej jakim jest rozwiązanie układu równań liniowych AX=B. Przykład dotyczy rozwiązania równania

$$x_1 + 2x_2 = 3$$
$$4x_1 + 5x_2 = 6$$

czyli w zapisie macierzowym

$$A = \begin{bmatrix} 1 & 2 \\ 4 & 5 \end{bmatrix}, X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, B = \begin{bmatrix} 3 \\ 6 \end{bmatrix}.$$

Zauważmy, że zastosowano indeksowanie macierzy kolumnowo (nie wierszowo, jak na przykład w języku C). Jest to ukłon w stronę języka Fortran, który indeksuje macierze kolumnowo, a jest do dziś podstawowym językiem szybkich i wydajnych obliczeń numerycznych. Biblioteki BLAS oraz LAPAC w nim zresztą powstały. Użytkownik, jeżeli zechce, może jednak w pakiecie UTL_NLA korzystać również z indeksowania wierszowego. Kod PL/SQL rozwiązujący powyższe równanie jest następujący:

```
DECLARE
  A     utl_nla_array_dbl := utl_nla_array_dbl(1, 4, 2, 5);
  B     utl_nla_array_dbl := utl_nla_array_dbl(3, 6);
  ipiv  utl_nla_array_int := utl_nla_array_int(0,0);
  info  integer;

BEGIN
  UTL_NLA.LAPACK_GESV (
    n      => 2,    -- A number of rows and columns
    nrhs   => 1,    -- B number of columns
    a      => A,    -- matrix A
    lda    => 2,    -- max(1, n)
    ipiv   => ipiv, -- pivot indices (set to zeros)
    b      => B,    -- matrix B
    ldb    => 2,    -- ldb >= max(1,n)
    info   => info, -- operation status (0=sucess)
    pack   => 'C'   -- how the matrices are stored
                    -- (C=column-wise)
);

  dbms_output.put_line('LINEAR EQUATION SOLVED');
  dbms_output.put_line('-------------');
  dbms_output.put_line('info = '||info);
  IF info = 0 THEN
    FOR i IN 1..B.count LOOP
      dbms_output.put_line('x' || i ||' = ' || TO_CHAR(B(i),'99.99'));
    END LOOP;
  END IF;
END;
/
```

Po wykonaniu powyższego bloku anonimowego otrzymujemy wynik:

```
LINEAR EQUATION SOLVED
----------
info = 0
x1 =  -1.00
x2 =   2.00
```

7. Wnioski

W pracy przedstawiono najważniejsze i najpopularniejsze zdaniem autorów metody służce do redukcji wymiarowości danych oraz do ich wizualizacji. W ogólności metod tych jest o wiele więcej i zainteresowane osoby z pewnością będą mogły dotrzeć do właściwych tekstów źródłowych.

Redukcja wymiarowości jest bardzo ważnym elementem statystycznej analizy danych oraz analiz wykonywanych w ramach szeroko pojętej eksploracji danych. Dzięki redukcji wymiarowości osiągamy nie tylko to, że mamy do przeanalizowania po prostu mniej danych. Najważniejsze zdaniem autorów jest mimo wszystko to, że dane po zredukowaniu pokazują nam niejednokrotnie więcej informacji niż przed redukcją! Wiele zależności staje się bardziej czytelnych i zmniejsza się prawdopodobieństwo ich przeoczenia. Bardzo często bowiem jest tak, że ilość posiadanych danych przerasta nas i bardzo łatwo wówczas pogubić się w ich analizie. Oczywiście nie wszystkie dane należy automatycznie i bez zastanowienia redukować. Warto jednak zawsze rozważyć taką możliwość i poeksperymentować.

Bibliografia

[1] Little R.J.A., Rubin D.B.: Statistical Analysis with Missing Data, J. Wiley & Sons, New York, 1987

[2] Schafer J.L.: Analysis of Incomplete Multivariate Data, Chapman & Hall, Londyn, 1997

[3] Schafer J.L.: Multiple imputation: a primer, Statistical Methods in Medical Research, 8, 3-15, 1999

[4] Koronacki J., Mielniczuk J.: Statystyka, WTN, 2004

[5] Koronacki: J., Ćwik J.: Statystyczne systemy uczące się, WTN, 2005

[6] Li J. X.: Visualization of High Dimensional Data with Relational Perspective Map. Information Visualization 2004, Vol. 3, No. 1, 49-59

[7] http://wazniak.mimuw.edu.pl/index.php?title=Eksploracja_danych

[8] GUS, http://www.stat.gov.pl/gus/prod_bud_inw_PLK_HTML.htm

[9] James Xinzhi Li: Visualization of High Dimensional Data with Relational Perspective Map. *Information Visualization*, Vol. 3, No. 1, 49-59, 2004

[10] http://www.visumap.net/index.aspx

[11] Oracle® Database PL/SQL Packages and Types Reference, 11g Release, Part Number B28419-03

[12] http://en.wikipedia.org/wiki/Basic_Linear_Algebra_Subprograms

[13] http://en.wikipedia.org/wiki/LAPACK

[14] http://www.jcp.org/en/jsr/detail?id=247

[15] http://www.ploug.org.pl

[15] Kohonen T.: Self-Organizing Maps, Springer Series in Information Sciences, Vol. 30, Springer, Berlin, Heidelberg, New York, 2001, Third Extended Edition, ISBN 3-540-67921-9

[16] Duch W., Korbacz J., Rutkowski L., Tadeusiewicz R. (red.): TOM 6 SIECI NEURONOWE, Seria: Biocybernetyka i inżynieria biomedyczna 2000, Akademicka Oficyna Wydawnicza EXIT, Warszawa 2000, SBN: 83-87674-18-4

[17] Osowski S.: Sieci neuronowe Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 1994

[18] http://research.microsoft.com/~cmbishop/downloads/Bishop-GTM-Ncomp-98.pdf

[19] Svensen J.F.M.: GTM: The Generative Topographic Mapping, PhD dissertation, Aston Iniversity, Birmingham, Report NCRG/98/024, April 1998

[20] Java Data Mining API 2.0, JSR 247; http://www.jcp.org/en/jsr/detail?id=247

[21] Morzy M.: Oracle Data Mining - odkrywanie wiedzy w dużych wolumenach danych, XI Krajowa Konferencja PLOUG'2005, „Systemy informatyczne. Projektowanie, implementowanie, eksploatowanie", Zakopane, 2005

[22] Implementacja metod eksploracji danych - Oracle Data Mining. Materiały IV Szkoły PLOUG; http://www.ploug.org.pl/showhtml.php?file=szkola/szkola_4/materialy

[23] http://en.wikipedia.org/wiki/Winsorising

[24] Schölkopf B., Smola A. J., Müller, K.R.: Nonlinear component analysis as a kernel eigenvalue problem; Neural Computation, 10, 1299-1319, 1998

[25] http://www.ncrg.aston.ac.uk/GTM/