Przemysław Włodarczyk

Wizualizacja danych

Praca magisterska pod kierunkiem

prof. Włodzisława Ducha

Wydział Matematyki i Informatyki

Uniwersytet Mikołaja Kopernika, Toruń 2007

Spis treści

1. Wstęp

Eksploracja danych (ang. *data mining*) to jeden z etapów procesu odkrywania wiedzy z baz danych. Obszary jej zastosowania obejmują miejsca, w których stosuje się systemy informatyczne gromadzące pozyskane dane w postaci baz. Jak wiadomo bazy danych charakteryzują się dużą prostotą konstrukcji co powoduje, że znajdują zastosowanie prawie we wszystkich dziedzinach życia. Oczywiście kolejnym powodem ich tworzenia jest potrzeba składowania danych takich jak: dane klientów w firmach, dane pacjentów w szpitalach, wyniki badań , doświadczeń, dane towarów produkowanych bądź sprzedawanych lub jakiekolwiek inne informacje. Natomiast wszędzie tam, gdzie została już utworzona taka baza, pojawia się potrzeba analizy zgromadzonych w niej danych w celu odkrycia nieznanej dotąd wiedzy, takiej jak np. określenie grupy klientów do jakich trafiają produkty firmy, stworzenie długoterminowej prognozy pogody itp. Istnieje wiele technik eksploracji danych, i właśnie jedną z nich jest tytułowa wizualizacja.

Celem wizualizacji danych jest pokazanie posiadanych informacji w sposób pozwalający na ich dokładne i efektywne zrozumienie oraz analizę. Dzieje się tak dlatego ponieważ ludzie dobrze „skanują", rozpoznają i zapamiętują przedstawione im obrazy (kształt, długość, budowa itp.). Dzięki wizualizacji możemy łączyć wielkie zbiory danych i

pokazać wszystkie informacje jednocześnie, co znacznie ułatwia analizę. Możemy również stosować porównania wizualne, dzięki którym dużo łatwiej stwierdzić wiele faktów. Kolejną zaletą jest możliwość analizy danych na kilku poziomach szczegółowości.

Z wizualizacją mamy do czynienia na każdym kroku naszego życia. Reprezentacja graficzna jest używana w telewizji, w prasie i w każdym innym źródle informacji (wyłączając stacje radiowe), gdy tylko mamy do czynienia z danymi numerycznymi. Wizualizacja jest niezbędna gdy chcemy: pokazać kurs pewnej waluty na przełomie określonego czasu (wykres liniowy), wyniki wyborów (histogramy) lub chociażby prognozę pogody. Jednak nie są to jedyne przykłady reprezentacji graficznej danych. Może ona służyć nie tylko ułatwieniu dostrzeżenia pewnych własności, lecz wręcz ich odkryciu. Dotyczy to przede wszystkim wielkich zbiorów danych, które są kompletowane przez wiele lat na rzecz późniejszych badań. Właśnie w takim celu tworzone są narzędzia do wizualizacji danych w aplikacjach typu GhostMiner lub Yale.

2. Eksploracyjna Analiza Danych (EDA)

2.1. Wprowadzenie

Eksploracyjna Analiza Danych (EDA – Exploratory Data Analysis) wykorzystuje różnego rodzaju techniki (przede wszystkim graficzne) w celu: odkrycia istotnych zmiennych, szumu danych, struktury danych, przetestowania podstawowych założeń. Jednak nie chodzi tylko o zbiór technik. EDA przede wszystkim oznacza inne podejście do analizy danych. W porównaniu do analizy klasycznej:

problem => dane => model => analiza => wnioski,

oraz analizy Bayes'owskiej:

problem => dane => model => rozkład danych => analiza => wnioski,

kolejność postępowania w EDA jest następująca:

problem => dane => analiza => model => wnioski.

Jak widać każde z wyżej wymienionych podejść do analizy danych zaczyna się od jakiegoś naukowego bądź technicznego problemu, a kończy się na odpowiednich wnioskach. Jednak bardzo istotne są kroki pośrednie. W podejściu klasycznym oraz Bayes'owkim narzucany jest pewien model danych, a ich analiza oparta jest o ten model. EDA nie zakłada żadnego, lecz skupia się na strukturze danych i pozwala im „zasugerować" jaki model będzie dla nich najlepszy. Techniki składające się na podejście klasyczne są mocno

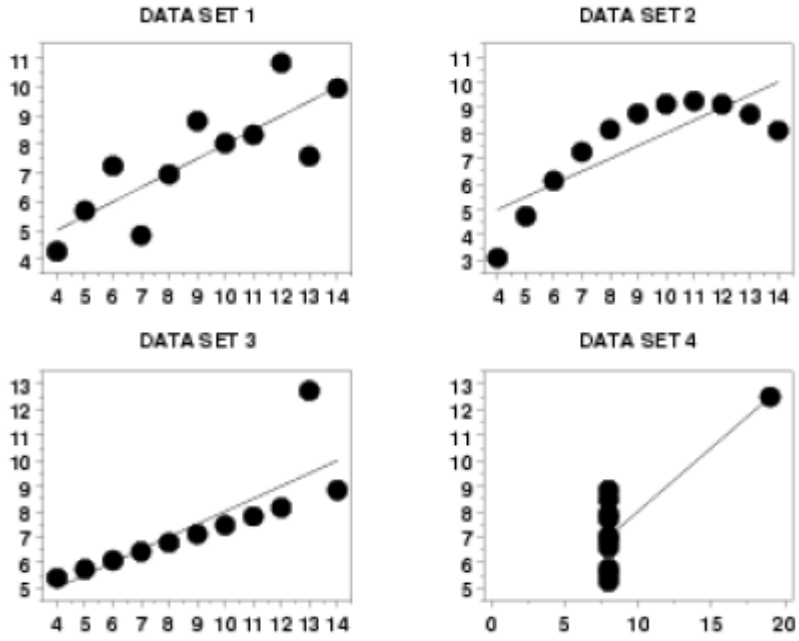
sformalizowane, podczas gdy EDA oferuje metody, których wyniki są zależne od interpretacji osoby (każdy analityk może wyciągnąć inne wnioski). W podejściu klasycznym dane zazwyczaj reprezentuje kilka liczb, które z jednej strony opisują ważne wartości takie jak wariancja , ale z drugiej strony koncentrując się tylko na nich (możemy pominąć inne, równie istotne). EDA oferuje techniki, które często wykorzystują wszystkie dostępne dane. Testy, które bazują na technikach klasycznych są bardzo wrażliwe na poczynione założenia, a prawdziwość wyników zależy od prawdziwości założeń. Często założenia te są nieznane lub nie zostały przetestowane, wtedy ciężko określić czy test daje dobre wyniki. Większość technik EDA nie czyni żadnych założeń, tylko przedstawia wszystkie dane w niezmienionej postaci, co pozwala uniknąć wielu błędów (założenie o modelu, przykład poniżej). EDA opiera się na metodach graficznych, ponieważ są one najkrótszą drogą do odkrycia modelu, danych, korelacji bądź szumu. Weźmy na przykład cztery zbiory:

Zbiór 1		Zbiór 2		Zbiór 3		Zbiór 4	
X_1	Y_1	X_2	Y_2	X_3	Y_3	X_4	Y_4
10.00	8.04	10.00	9.14	10.00	7.46	8.00	6.58
8.00	6.95	8.00	8.14	8.00	6.77	8.00	5.76
13.00	7.58	13.00	8.74	13.00	12.74	8.00	7.71
9.00	8.81	9.00	8.77	9.00	7.11	8.00	8.84
11.00	8.33	11.00	9.26	11.00	7.81	8.00	8.47
14.00	9.96	14.00	8.10	14.00	8.84	8.00	7.04
6.00	7.24	6.00	6.13	6.00	6.08	8.00	5.25
4.00	4.26	4.00	3.10	4.00	5.39	19.00	12.50
12.00	10.84	12.00	9.13	12.00	8.15	8.00	5.56
7.00	4.82	7.00	7.26	7.00	6.42	8.00	7.91
5.00	5.68	5.00	4.74	5.00	5.73	8.00	6.89

Następnie obliczamy dla nich ilość próbek, wartość średnią X, wartość średnią Y, korelację oraz dopasowujemy funkcje liniowe postaci Y = aX +b (zakładany model). Dla każdego zbioru mamy:

N = 11
Wartość średnia X = 9.0
Wartość średnia Y = 9.0
a = 0.5
b = 3
Korelacja = 0.816 (zbiór 4 0.817)

Używając zwykłych metod statystycznych możemy założyć, że zbiory te są bardzo do siebie podobne (a nawet takie same). Jednak gdy przeanalizujemy wszystkie dane i dla każdego z nich stworzymy wykres rozproszony (rys. 2. 1) wówczas możemy stwierdzić że:



2.1 Wykresy rozproszone przedstawiające cztery zbiory

Pierwszy zbiór jest liniowy z delikatnym rozproszeniem, drugi jest kwadratowy, trzeci ma jednego „wyrzutka”(*ang. outlier*), a czwarty jest słabo zaprojektowany, z jednym punktem mocno oddalonym od większości. Statystyka ilościowa w tym przypadku nie jest błędna, ale jest niekompletna. Wynika to z założenia liniowości modelu danych. Dopiero dzięki metodom EDA możemy dostrzec prawdziwe struktury zbiorów.

2.2. Metody graficzne

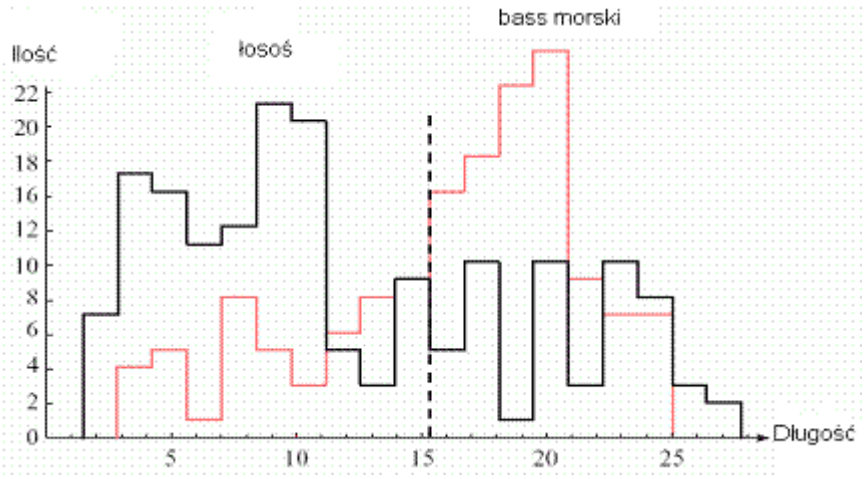
EDA charakteryzuje duży nacisk na metody graficzne, co pozwala na lepszy wgląd w dane. Technik wizualizacyjnych jest wiele i nie sposób wszystkie opisać, dlatego skupimy się na opisaniu najbardziej istotnych w kontekście dalszej pracy. Ze względu na dużą różnorodność metod graficznych można wprowadzić pewną ich klasyfikację. Zakładamy, że każdy z użytych zbiorów danych składa się z wektorów o takiej samej ilości cech, gdzie jedną z cech wektora może być numer, bądź nazwa klasy, do której on należy.

2.2.1. Wykresy jednej zmiennej

Są to metody, które pozwalają na wizualizację jednej cechy (dwóch licząc uwzględnienie klasy na wykresie np. przez kolor). Dzięki nim możemy obejrzeć rozkład cechy, wartości średnie, odchylenie standardowe itp.. Zaliczamy do nich m.in. wykresy pudełkowe oraz histogramy.

Histogramy są narzędziem, dzięki któremu możemy graficznie odtworzyć rozkład danej

cechy. Poza tym pozwalają one również dostrzec rozpiętość, skośność oraz szum danych. Często podczas tworzenia histogramów wprowadza się podział na klasy (rys. 2.2).



2.2 Histogram przedstawiający rozkład długości gatunków ryb z zaznaczoną optymalną granicą podziału

Aby stworzyć histogram musimy najpierw określić ilość *n* oraz wielkość poszczególnych przedziałów (zazwyczaj wielkości są sobie równe). Zakładając $n = 20$ wielkość jednego przedziału wynosi:

$$\Delta=(x_{max}-x_{min})/n \tag{2.1}$$

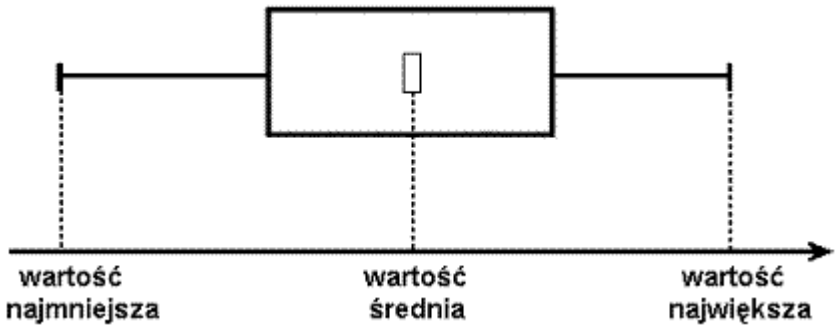
Następnie określamy kolejne przedziały:

$$r_i=[x_{min}+(i-1)\Delta,\ x_{min}+i\Delta],\ gdzie\ \ i=1...n \tag{2.2}$$

Oraz obliczamy ile wektorów do nich „wpada” (sprawdzamy do jakiego przedziału należy określona cecha poszczególnych wektorów). Reprezentacja graficzna powstaje poprzez zaznaczenie ilości przypadków na osi pionowej, a na osi poziomej odpowiadającej im wartości (przedziału).

Można też tworzyć histogramy, w których przedziały są równej, nie szerokości, a wysokości. Oznacza to, że każdym z nich musi znaleźć się równa ilość wektorów. W tym wypadku dzielimy ilość próbek na *n* części i odpowiednio dobieramy wielkości przedziałów. Metoda ta nie pozwala jednak ujawnić ważnych własności danych i w kontekście analizy danych ma niewielkie zastosowanie. Istnieją także histogramy dwuwymiarowe przedstawiane w trzech wymiarach (odnoszą się one do dwóch cech).

W celu pokazania statystyk takich jak wartość średnia, maksymalna, minimalna, bądź odchylenie standardowe warto zastosować wykresy pudełkowe (rys. 2.3).



2.3 Wykres pudełkowy

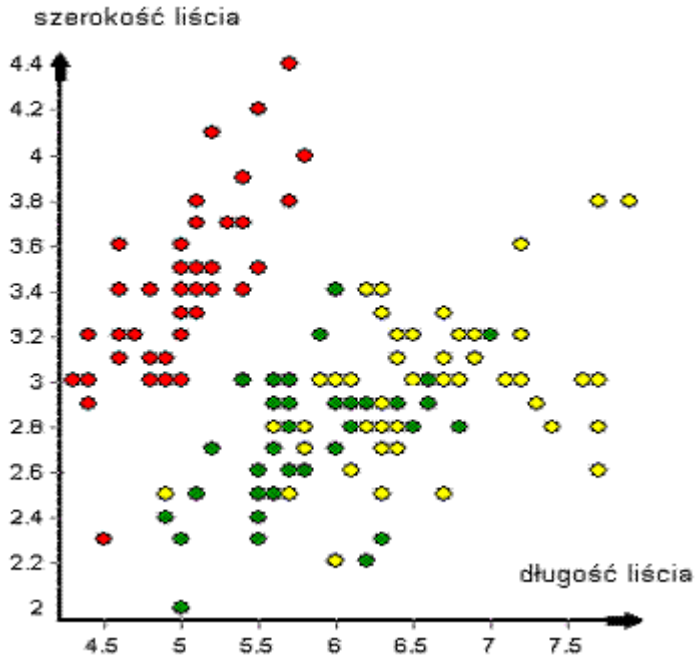
Według definicji lewy bok pudełka powinien wyznaczać wielkość pierwszego kwartyla (wielkość cechy, do której znajduje się 25% wszystkich obserwacji – w naszym przypadku wektorów), prawy wielkość trzeciego kwartyla (75% obserwacji) oraz dodatkowo powinna być zaznaczona mediana (50% obserwacji). Jednak ze względu na określone zastosowanie wykresy pudełkowe można stworzyć w następujący sposób: zaznaczamy wartość najmniejszą, największą oraz średnią, a wielkość pudełka jest określona przez odchylenie standardowe (przy czym środek pudełka to wartość średnia). Można oczywiście zestawić

kilka takich wykresów. Zestawienie takie może służyć do wstępnej analizy, która pozwoli nam stwierdzić zakres wartości poszczególnych cech, określić wielkość ich wariancji (kwadrat odchylenia standardowego) itd.

2.2.2. Rzuty na dwie współrzędne

Do tej grupy zaliczamy metody, które pozwalają pokazać jednocześnie dwie współrzędne. Techniki te umożliwiają odkrycie związków między cechami (np. korelacja).

Wykresy rozproszone (*ang. scatterplot*) są podstawowym narzędziem, które rzutuje dane na dwie współrzędne. Ich analiza powinna odbywać się pod kątem odkrycia korelacji między poszczególnymi cechami oraz klasteryzacji danych. Wykresy rozproszone są tworzone poprzez zaznaczanie kolejnych punktów danych w przestrzeni dwuwymiarowej. Wartość współrzędnej X odnosi się do pierwszej cechy, a Y do drugiej. Często mamy do czynienia z danymi podzielonymi na klasy (rys. 2.4).

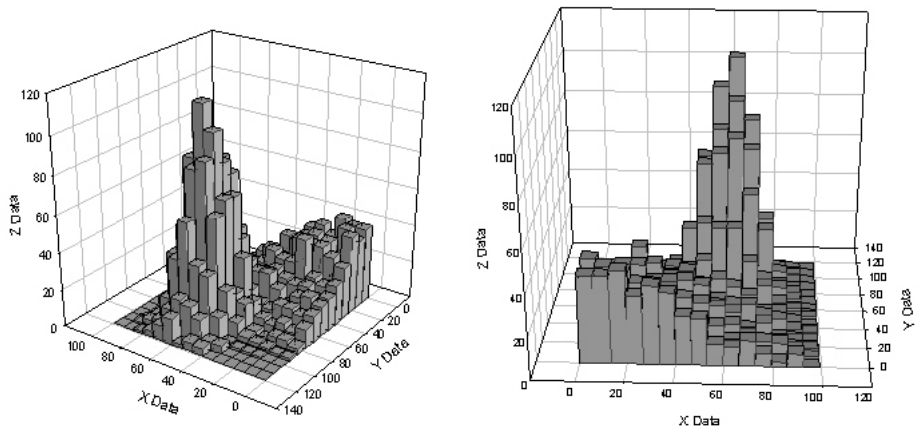


2.4 Wykres rozproszony na podstawie danych „Iris Plants Database”

W powyższym przypadku oś pionowa odpowiada szerokości liścia, a oś pozioma długości liścia. Łatwo zauważyć, iż wybór tych dwóch cech dobrze oddziela klasę Iris Setosa od dwóch pozostałych. Dzięki wykresom rozproszonym łatwo możemy określić, które pary cechy są redundantne. Umożliwiają one również znalezienie, jak w powyższym przypadku, pary cech, która pozwala na określenie klasy wektora. Główną wadą wykresów rozproszonych jest nakładanie się punktów, gdy mamy do czynienia z cechami dyskretnymi.

Gdy mamy do czynienia z dużymi zbiorami danych możemy stworzyć macierz wykresów rozproszonych. Dokładnie chodzi o zestawienie wszystkich bądź kilku wykresów rozproszonych możliwych do uzyskania w obrębie jednego zbioru danych. Pozwala ono na szybkie określenie, które z par cech mogą okazać się istotne, które są redundantne, oraz które nie są istotne dla określonego problemu (np. nie separują żadnej z klas od reszty).

Drugą metodą pozwalającą jednoczesne pokazanie dwóch cech są, wcześniej wymienione, histogramy dwuwymiarowe (rys. 2.5)

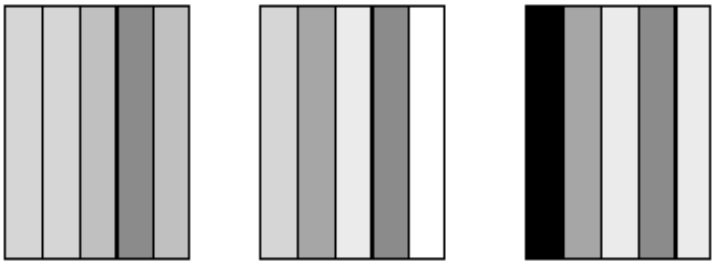


2.5 Przykłady histogramów dwuwymiarowych przedstawionych w trzech wymiarach.

Jednak w przypadku podziału danych na dwie lub więcej klas, ze względu na swoją konstrukcję, stają się one bardzo ciężkie do analizy. Powoduje to, że metoda ta może być przydatna tylko w określonych przypadkach, a w pozostałych nie jest wystarczająco efektywna, by była warta zastosowania.

2.2.3. Metody używające koloru oraz odcieni

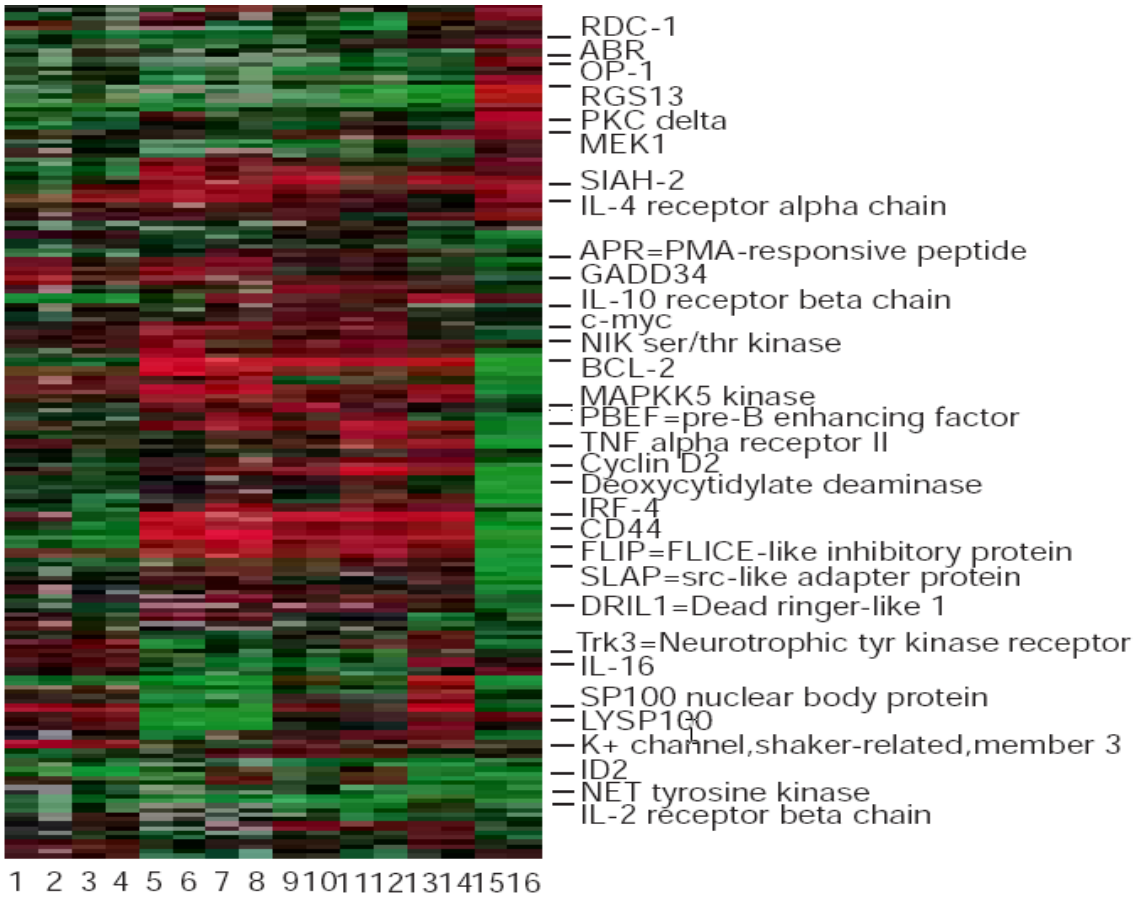
Jest to kolejny pomysł na wizualizację danych, wykorzystujący naturalne ludzkie zdolności rozróżniania kolorów (dotyczy ludzi nie cierpiących na choroby takie jak daltonizm). Do metod tych należą prostokąty Fortsona. Pozwalają one na wizualizację wielu cech jednocześnie. Wielkość zmiennych jest wyrażona odcieniem szarości kolejnych prostokątów (rys. 2.6).



2.6 Prostokąty Fortsona

Nie jest to jednak metoda pozwalająca na wnikliwą analizę danych. Można oczywiście zestawić kilka wektorów (jak na rysunku powyżej), ale wyciągnięcie jakichkolwiek przydatnych informacji z takiego zestawienia jest bardzo trudne. Zamieszczenie ich w tym zestawieniu ma bardziej na celu pokazanie różnorodność metod, ponieważ praktyczne zastosowanie prostokątów Fortsona jest znikome.

Istnieją też specjalne histogramy używane w bioinformatyce. W odróżnieniu od wcześniej opisanych, wysokości poszczególnych słupków zastąpione są odpowiednim kolorem.

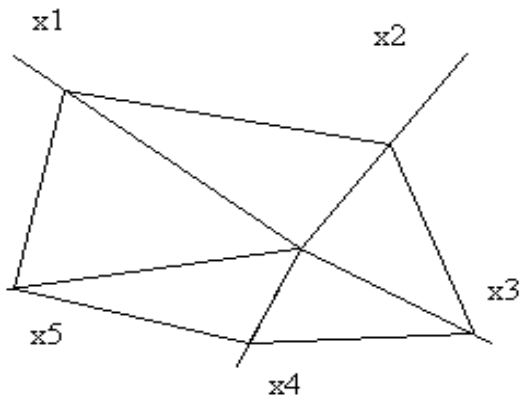


2.7 Histogramy w bioinformatyce

W powyższym przypadku mamy do czynienia z dwiema cechami dyskretnymi (16 genów i próbki) oraz z jedną ciągłą (aktywność poszczególnych genów), która została znormalizowana do przedziału [-1, +1]. Używamy kolorów: jasny zielony – słaba aktywność genu (czyli -1), czarny – normalna aktywność (0), jasny czerwony – wysoka aktywność (+1). Wartości pośrednie reprezentują kolejne odcienie danych kolorów. Jak widać na powyższym przykładzie nawet metody, na pierwszy „rzut oka" niezbyt wyraźne i łatwe do przeanalizowania znajdują obszerne zastosowanie w pewnych dziedzinach życia.

2.2.4. Metody korzystające z osi gwiazdowych (radarowych)

Ta grupa składa się tylko z jednej metody czyli wykresów gwiazdowych (ang. star plot, radar plot). Technika pozwala na zaprezentowanie danych wielowymiarowych z dowolną ilością zmiennych. Każdy przypadek jest reprezentowany przez wykres, przypominający gwiazdę, w którym każdy promień przedstawia jedną zmienną (rys. 2.8).



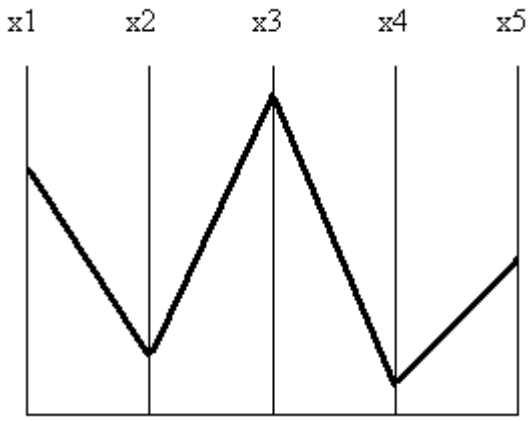
2.8 Wykres gwiazdowy (radarowy) przedstawiający wektor składający się z pięciu zmiennych

Analizowanie pojedynczych „gwiazd" może okazać się mało efektywne, dlatego właśnie należy zestawić kilka wykresów. Łatwiej jest zauważyć schemat w danych, kiedy wektory są przedstawione w nie arbitralnym porządku, a cechy są przyporządkowane do promieni w logicznej kolejności.

Metoda ta jest szczególnie przydatna, gdy wszystkie zmienne mają taki sam wymiar. Niestety w przypadku bardzo dużych zbiorów danych staje się bezużyteczna (analiza wykresu składającego się np. z 700 „gwiazd").

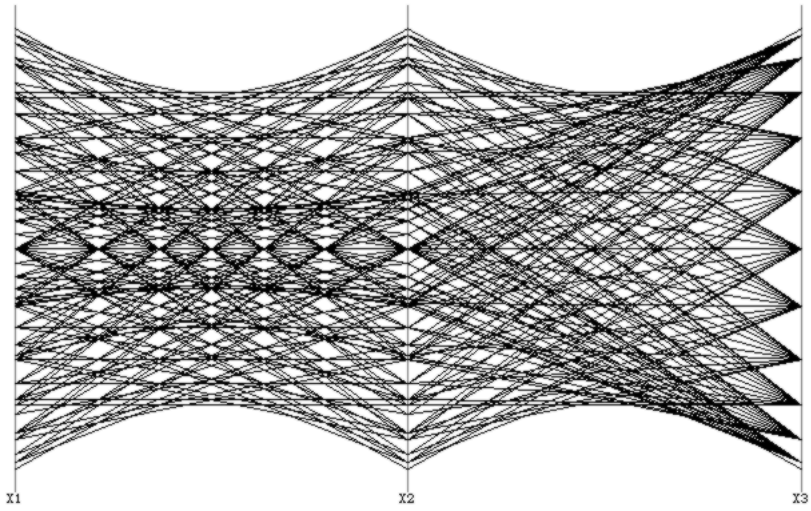
2.2.5. Metody wykorzystujące osie współrzędnych

Ponownie bierzemy pod uwagę tylko jedną metodę, czyli współrzędne równoległe. Polega ona na zaznaczeniu kolejnych wartości zmiennych na odpowiadających im, równoległych do siebie osiach (rys. 2.9).



2.9 Punkt C = (c_1, c_2, c_3, c_4, c_5) przedstawiony za pomocą współrzędnych równoległych

Powyższy przypadek jest bardzo prosty. Mamy dany punkt (wektor) składający się z pięciu zmiennych (cech), każdą z pionowych osi traktujemy jako przestrzeń kolejnych zmiennych. Współrzędne równoległe są bardzo istotnym narzędziem. Pozwalają one na wizualizację całego zbioru danych, co z kolei pozwala na odkrycie zależności pomiędzy przypadkami (wektorami) jak i cechami (zmiennymi). Nie jest to jednak takie proste ze względu na nakładanie się linii. Na poniższym rysunku (rys. 2.10) przedstawiona jest 5-wymiarowa kula. Gdy chcielibyśmy przyjrzeć się tylko jednemu wektorowi, okazuje się to praktycznie niemożliwe, nawet przy tak określonej strukturze zbioru.



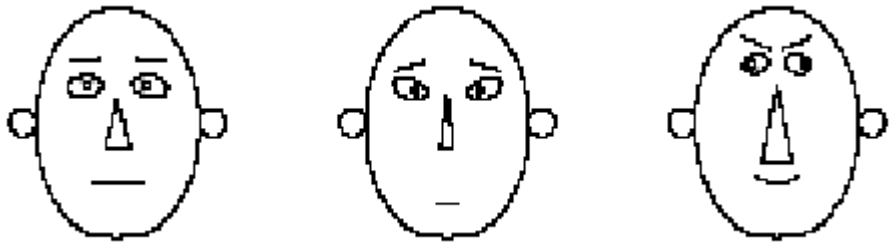
2.10 Trójwymiarowa kula przedstawiona za pomocą współrzędnych równoległych

2.2.6. Wykresy wykorzystujące predyspozycje człowieka

Zajmiemy się metodami, które aby ułatwić ich analizę, wykorzystują elementy dobrze, przez ludzi, rozpoznawalne. Dotyczy to przede wszystkim schematycznych rysunków, które człowiek często poddaje analizie, co powoduje zwiększoną wrażliwość na zmiany w ich strukturze.

Twarze Chernoffa jest to metoda zaproponowana w 1973 roku przez Hermana Chernoffa (rys. 2.11). W tym wypadku wartości różnych wymiarów prezentowane są przez wielkość, kształt bądź rozmieszczenie poszczególnych elementów twarzy (nos, oczy, brwi itd.). Większość ludzi przez całe życie musi rozpoznawać spoglądając na twarze: rodzinę, znajomych lub osoby publiczne. Powoduje to, iż w mózgu tworzą się struktury odpowiedzialna za rozpoznawanie twarzy. Oczywiście istnieje wiele takich struktur (zdolności manualne, umiejętność prowadzenia samochodu, ogólnie pojęte poczucie estetyki itd.), jednak czynność odróżniania od siebie twarzy wydaje się być jedną z najbardziej powszechnych umiejętności. Dzięki czemu osoba analizująca z łatwością

dostrzeże różnice pomiędzy wykresami (twarzami). Jednak podobnie jak w przypadku wykresów gwiazdowych analiza jednocześnie kilkuset przypadków może okazać się niemożliwa.



2.11 Twarze Chernoffa

Metody tego typu pokazują przede wszystkim, że można próbować bardziej niekonwencjonalnych rozwiązań, starać się wykorzystać naturalne ludzkie predyspozycje.

2.2.7. Podsumowanie metod graficznych

Wymienione wyżej techniki wizualizacji danych pokazują zarówno różnorodne podejście do problemu, jak i tak naprawdę szeroką gamę zadań, z którymi metody te muszą sobie radzić. Istnieje oczywiście jeszcze wiele innych metod reprezentacji graficznej. Jednak wybrane metody, z jednej strony pozwalają stworzyć funkcjonalny i kompletny pakiet wizualizacyjny (wykresy pudełkowe, histogramy, wykresy rozproszone, współrzędne równoległe) jak i pokazać inne podejście do problemu (twarze Chernoffa, wykresy gwiazdowe, histogramy dwuwymiarowe przedstawione w trzech wymiarach itd.). Oczywiście spoglądając na poszczególne techniki musimy zwrócić uwagę nie tylko na ich

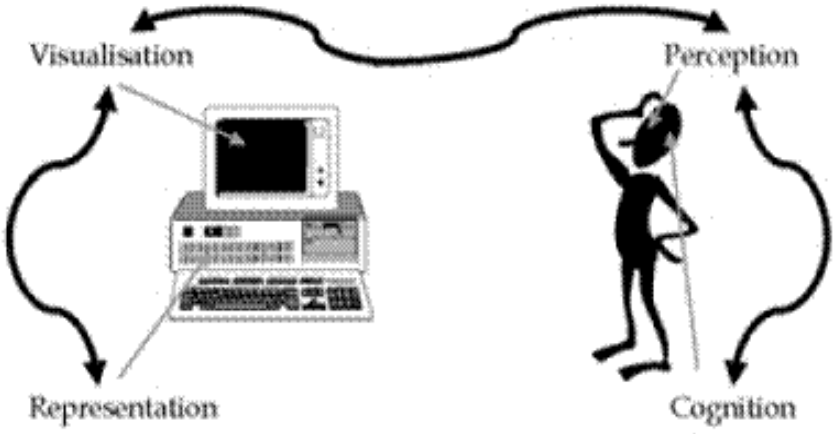
przydatność (nawet dużą) w określonych przypadkach, ale bardziej na uniwersalność. Metoda jest wtedy efektywna i warta zastosowania, gdy możemy jej użyć zarówno do danych dyskretnych jak i ciągłych, wielowymiarowych jak i niskowymiarowych, z dużą ilością przypadków oraz gdy jest ich kilka itd. no i oczywiście gdy za każdym razem można wyciągnąć przydatne wnioski.

3. Problem danych wielowymiarowych

3.1. Psychologiczne podstawy wizualizacji danych

Techniki wizualizacji mają na celu zaprezentowanie człowiekowi danych w sposób dokładnie przekazujący informacje w nich zawarte oraz wymagający jak najmniejszego wysiłku do ich zrozumienia. Powoduje to, że obrazy graficzne użyte w procesie wizualizacji powinny powstawać w oparciu o dobre zrozumienie ludzkiego układu wzrokowego. Techniki wizualizacji mają także umożliwić zarówno analizowanie jak i operowanie na danych. Dlatego struktura informacji powinna być zgodna z wymaganiami reprezentacyjnymi oraz preferencjami ludzkich procesów poznawczych. Jest to pierwszy powód, dla którego techniki modelowania danych użyte podczas wizualizacji powinny być oparte o zrozumienie działania ludzkiej pamięci oraz reprezentacji kognitywnej. Drugi mówi o silnym związku między percepcją, a procesami poznawczym, przez co percepcja jest bardzo wrażliwa na strukturalną budowę ludzkiej pamięci.

W oparciu o powyższe stwierdzenia psychologia często nakazuje nam najpierw uzyskać odpowiednią reprezentację danych, a dopiero potem ich wizualizację (rys. 3.1).



3.1 Schemat przedstawiający podstawowe aspekty wizualizacji danych.

W przypadku danych wielowymiarowych często nie jesteśmy w stanie wybrać odpowiednich cech potrzebnych do wizualizacji (metody wykorzystujące mniejszą ilość wymiarów), bądź żadne nie są odpowiednie i należy stworzyć nowe. Wówczas występuje najbardziej znany problem przygotowania danych, czyli zbyt duża ilość wymiarów (cech) w zbiorze. Rozwiązanie polega na redukcji wymiarowości.

3.2. Redukcja wymiarowości

Główny problemem związanym z wizualizacją są dane wielowymiarowe. Oznacza to, że każdy przypadek opisany jest dużą ilością cech. Oczywiście istnieją techniki, które są w stanie zobrazować wszystkie wymiary jednocześnie (np. współrzędne równoległe), jednak metody te sprawdzają się tak naprawdę dla niewielkiej liczby cech. W takim przypadku należy przygotować dane do wizualizacji, tak aby później uzyskana reprezentacja graficzna pozwalała wyciągnąć odpowiednie wnioski (należy stworzyć

odpowiednią reprezentację danych). Rozwiązaniem jest redukcja wymiarowości. Problem stanowi jedynie wybór wymiarów (bądź stworzeniu nowych w oparciu o już istniejące), tak aby zachowały one informacje istotne dla osoby analizującej dane. Często taką informacją jest miara podobieństwa, dystans, bądź wariancja punktu wewnętrznego. Najczęściej używane w tym celu narzędzia to: Analiza Głównych Składowych (PCA – Principal Component Analysis), Analiza Składowych Niezależnych (ICA – Independent Component Analysis), Liniowa Analiza Dyskryminacyjna (LDA – Lineał Discriminant Analysis), Analiza Dyskryminacyjna Fishera (FDA – Fisher Discriminant Analysis) oraz filtry cech.

3.2.1. Analiza Głównych Składowych (PCA)

Analiza Głównych Składowych (PCA – Principal Component Analysis) oprócz redukcji wymiarowości pozwala na odkrycie wzorców zawartych w danych. Jest to szczególnie przydatne w przypadku danych wielowymiarowych, gdzie nie możemy sobie pozwolić na reprezentację graficzną całego zbioru. Gdy znajdziemy odpowiednie wzorce możemy zredukować ilość wymiarów minimalizując stratę informacji.

Załóżmy, że mamy pewien zbiór danych, składający się z *m* przypadków, gdzie każdy z nich jest opisany przez *n* cech. Aby wyznaczyć główne składowe (ang. *principal components*) należy:

- wyznaczyć wartość średnią dla każdej z cech:

$$\bar{X} = \sum_{i=1}^{n} X_i \tag{3.1}$$

- obliczyć kowariancje dla każdej pary cech:

$$cov(X,Y)=\frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{(n-1)} \tag{3.2}$$

- po obliczeniu wszystkich kowariancji (dla każdej pary cech) tworzymy ich macierz:

$$cov=\begin{pmatrix} cov(X_1,X_1) & cov(X_1,X_2) & \dots & cov(X_1,X_n) \\ cov(X_2,X_1) & cov(X_2,X_2) & \dots & cov(X_2,X_n) \\ \dots & \dots & \dots & \dots \\ cov(X_n,X_1) & cov(X_n,X_2) & \dots & cov(X_n,X_n) \end{pmatrix} \tag{3.3}$$

Oczywiście kowariancje znajdujące się na przekątnej są równe wariancjom poszczególnych cech. Kolejnym krokiem jest obliczenie wektorów oraz wartości własnych macierzy kowariancji. Wektor własny macierzy jest to wektor, który po pomnożeniu z lewej przez tą macierz w wyniku daje swoją wielokrotność. Wielkość przez jaką musimy pomnożyć wektor własny, aby otrzymać wynik wyżej opisanego mnożenia nazywamy wartością własną np. dla macierzy:

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix}, \text{ mamy } \begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 12 \\ 8 \end{pmatrix} = 4 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

wektor $\begin{pmatrix} 3 \\ 2 \end{pmatrix}$ jest wektorem własny, a 4 wartością własną dla tego wektora. Wektory własne przedstawiamy w postaci jednostkowej (nie jest to problemem, ponieważ o tym, że wektor jest wektorem własnym świadczy kierunek, a nie jego długość). Macierz kowariancji jest kwadratowa i ma wymiar *n x n,* więc posiada *n* wektorów własnych.

Kolejnym krokiem jest uporządkowanie jednostkowych wektorów własnych (o długości równej jeden) według odpowiadających im wartości własnych, od największej do

najmniejszej oraz umieszczenie ich w macierzy, którą nazwiemy wektorem cech:

$$WektorCech = (wekw_1 \;\; wekw_2 \;\; ... \;\; wekw_n) \;\; , \tag{3.4}$$

gdzie $warw_1 < warw_2 < ... < warw_n$

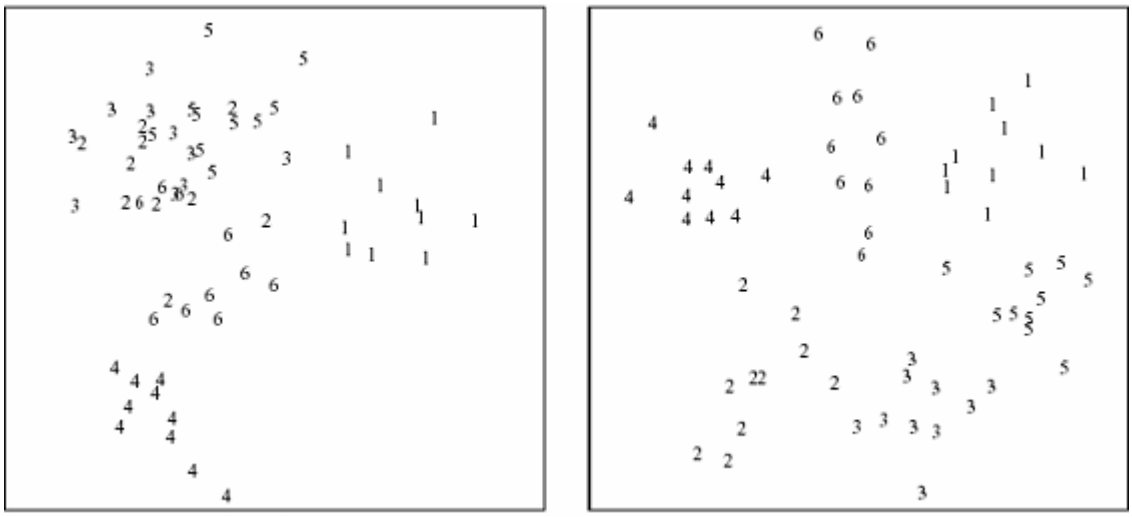
Następnie należy wybrać *k* pierwszych wektorów własnych, aby otrzymać transformację do danych *k*-wymiarowych. Ostatnim krokiem jest obliczenie macierzy:

$$DaneKońcowe^T = WektorCech^T \times Dane^T \;\; , \tag{3.5}$$

gdzie $Dane^T$, jest to macierz z danymi wejściowymi, gdzie każda kolumna oznacza pojedynczy przypadek, a każdy wiersz jeden z wymiarów (jedną cechę). W ten oto sposób powstaje nowa macierz *DaneKońcowe* wymiaru *m x k*.

Dzięki zastosowaniu techniki PCA powstają nowe, nieskorelowane cechy. Redukcja wymiarowości jest uzyskana poprzez odrzucenie cech z małą wariancją. Kolejną zaletą Analizy Głównych Składowych jest możliwość rekonstrukcji macierzy kowariancji w przypadku danych nisko wymiarowych.

Z drugiej strony podczas stosowania PCA największe wariancje decydują, które cechy zostaną użyte, co nie gwarantuje nam dobrej klasteryzacji danych. Po zastosowaniu kombinacji liniowych tracimy znaczenie cech. Kolejnym ograniczeniem redukcji wymiarowości przez PCA jest brak przystosowania do wizualizacji danych o nieliniowej strukturze (rys. 3.2). Jak widać na poniższym rysunku zastosowanie PCA na zbiorze „simplex5” (po lewej stronie) nie jest efektywne, ponieważ klasy 2, 3 oraz 5 nakładają się. Dopiero odwzorowanie nieliniowe przynosi pożądany efekt (rysunek po prawej stronie).



3.2 Wizualizacja zbioru danych „simplex5" poprzez liniowe i nieliniowe odwzorowania

3.2.2. Liniowa Analiza Dyskryminacyjna (LDA) oraz Analiza Dyskryminacyjna Fishera (FDA)

Liniowa analiza dyskryminacyjna (LDA – Linear Discriminant Analysis) jest to kolejna metoda, która nie tylko służy do redukcji wymiarów, ale również do klasyfikacji. LDA znajduje optymalną macierz transformacji, która zachowuje jak najwięcej informacji pozwalających rozdzielić poszczególne klasy.

Załóżmy, że mamy zbiór testowy składający się z N próbek, a każdą z nich opisuje p cech i są one podzielone na g klas. Aby sformułować procedurę optymalizacji musimy najpierw wyznaczyć wartości średnie (3.6) oraz macierze kowariancji (3.7) poszczególnych klas:

$$\bar{x}_j = \frac{1}{N_j} \sum_{i=1}^{N_j} x_i \quad , \tag{3.6}$$

$$\bar{W}_j = \frac{1}{N_j} \sum_{i=1}^{N_j} (x_i - \bar{x}_j)(x_i - \bar{x}_j)^T \quad , \tag{3.7}$$

a następnie zrobić to samo dla całego zbioru:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i \quad , \tag{3.8}$$

$$\bar{T} = \frac{1}{N} \sum_{i=1}^{N} (x_i - \bar{x})(x_i - \bar{x})^T \quad , \tag{3.9}$$

gdzie N_j ilość próbek klasy J. Możemy więc zdefiniować kryterium optymalizacji:

$$c = arg \max_{c_p} \frac{\left| c_p^T \bar{T} c_p \right|}{\left| c_p^T \bar{W} c_p \right|} \quad , \tag{3.10}$$

gdzie

$$\bar{W} = \frac{1}{N} \sum_{j=1}^{J} N_j \bar{W}_j \tag{3.11}$$

Kryterium pozwala jednocześnie zmaksymalizować „odległość" między klasami oraz zminimalizować „wielkość" każdej z nich. Gwarantuje to zachowanie większości, istotnych w kontekście separacji, informacji w nowej przestrzeni cech.

Analiza Dyskryminacyjna Fishera (FDA – Fisher's Discriminant Analysis) różni się od LDA tylko nielicznymi szczegółami. Najbardziej istotna jest różnica użytego kryterium. Dla FDA jest to kryterium Fishera (3.12):

$$\frac{c_i^T B c_i}{c_i^T \bar{W} c_i} = \max_{c \in R^p, c \neq 0} \frac{c^T B c}{c^T \bar{W} c} \quad , \tag{3.12}$$

gdzie B oznacza macierz kowariancji pomiędzy klasami:

$$B = \frac{1}{g-1} \sum_{j=1}^{g} N_j (\bar{x}_j - \bar{x})(\bar{x}_j - \bar{x})^T \tag{3.13}$$

Podobnie jak PCA zarówno LDA jak i FDA są mało skuteczne jeżeli mamy do czynienia ze zbiorami o strukturze nieliniowej. Z drugiej strony dużym plusem tych metod jest możliwość klasteryzacji.

3.2.3. Analiza Składowych Niezależnych (ICA)

Analiza Składowych Niezależnych (ICA – Independent Components Analysis) to technika dzięki której możemy odkryć w danych ukryte cechy. W przypadku ogólnym mamy dane reprezentowane przez wektory $x = (x_1,...,x_m)$**,** a składowe jako losowy wektor $s = (s_1,...s_n)$**.** Naszym zadaniem jest transformacja danych x, przy użyciu transformacji liniowej $s = Wx$, do postaci maksymalnie niezależnych składowych s. Niezależność mierzymy przy użyciu funkcji niezależności $F(s_1,...,s_n)$. Komponenty x_i wektora x^T są generowane przez niezależne składowe s_k $(k = 1,...n)$, przy użyciu wag $a_{i,k}$**:**

$$x_i = a_{i,1}\, s_1 + ... + a_{i,k}\, s_k + ... + a_{i,n}\, s_n \tag{3.14}$$

Więc każdy wektor **x** można zapisać w postaci:

$$x = \sum_{k=1}^{n} a_k\, s_k \tag{3.15}$$

Gdy złożymy wszystkie wektory bazowe wektorów x, postaci $a_k = (a_{1,k},...a_{m,k})^T$, w macierz $A = (a_1,...,a_n)$ otrzymamy równanie $x = As$, gdzie $s = (s_1,...s_n)^T$. Musimy wyznaczyć wektory s poprzez obliczenie kolejnych wektorów w oraz ustalanie funkcji kosztu, która albo maksymalizuje „*niegaussowość*" obliczonego $s_k = (w^T{*}x)$ lub

minimalizuje informację wzajemną. Czasami wiedza a priori na temat rozkładu funkcji prawdopodobieństwa danych wejściowych może być wykorzystana do skonstruowania funkcji kosztu. Wektory *s* możemy obliczyć mnożąc wektory danych wejściowych *x* przez macierz $W = A^{-1}$.

Powszechny problem, dla którego stosowane jest ICA to tzw. problem „ślepej separacji źródeł". Mamy do czynienia z danymi (wektor *x* – np. zapis dźwięku z różnych mikrofonów) , które są mieszaniną statystycznie niezależnych sygnałów (wektor *s* – np. głosy kilku jednocześnie mówiących osób). Za pomocą analizy składowych niezależnych możemy odseparować sygnały wektora *s*.

3.2.4. Filtry cech

Kolejną metodą są filtry cech. Są to funkcje zwracające indeks istotności *J(S|D)*, który pozwala stwierdzić jak istotny jest podzbiór cech *S* zbioru danych *D* dla zadania *Y*. Zazwyczaj dane *D* oraz zadanie są ustalone więc funkcję możemy zapisać w postaci *J(S)*. Aby wyznaczyć istotność danej cechy, bądź cech możemy używać nie tylko prostych funkcji, jak korelacje, ale nawet skomplikowanych algorytmów. Indeksy możemy wyznaczać dla pojedynczych cech X_i, $i = 1...N$, co pozwala ustalić ranking $J(X_{i1}) \leq J(X_{i2}) \leq ... J(_{Fin})$. Następnie możemy usunąć cechy plasujące się na końcu rankingu. Takie podejście jest właściwe tylko, gdy poszczególne cechy są niezależne. Zazwyczaj jednak tak nie jest. W przypadku, gdy cechy są skorelowane wybranie pary najważniejszych nie jest równoznaczne z wybraniem dwóch najwyżej sklasyfikowanych według rankingu.

Jest wiele sposobów na sprawdzenie istotności cech m. in. korelacja, odległość między

rozkładami prawdopodobieństwa, teoria informacji lub drzewa decyzji. Ciężko jest orzec, która jest z nich najlepsza, ponieważ wszystko zależy od danych wejściowych oraz klasyfikatora. Jeżeli jakaś metoda działa w przypadku dużej ilości klas, cech i próbek prawdopodobnie nie sprawdzi się w sytuacji gdy ilość klas, cech i próbek jest mała.

Filtry są jedną z najtańszych metod selekcji cech. W przypadku dużych zbiorów danych są one niezbędne. Dopiero po odrzuceniu większości cech przez filtry realne jest zastosowanie bardziej wyrafinowanych technik. Ich wielką zaletą jest prostota konstrukcji, właściwie wszystko zależy od testu istotności cech. Jeżeli jest to korelacja metoda ta będzie bardzo szybka.

Oprócz wyżej opisanych metod istnieją również techniki neuronowe (uczenie konkurencyjne – competitive learning, SOM – Self-Organizing Maps), statystyczne (skalowanie wielowymiarowe – multidimensional scaling) oraz kernelowe. Są to metody nieliniowe, duża bardziej zaawansowane niż PCA, LDA, FDA czy ICA.

4. Zaimplementowany pakiet metod wizualizacji

Pakiet narzędzi służących do wizualizacji danych został stworzony na potrzeby nowego narzędzia do analitycznej eksploracji danych – Intemi. Aplikacja została napisana w języku C# w środowisku .Net z wykorzystaniem pakietu do tworzenia wykresów TeeChart firmy Steema. Uzyskane rozwiązania są efektem analizy metod wizualizacji danych proponowanych przez EDA, oraz przez dostępne pakiety takie jak: GhostMiner, Yale oraz WEKA. W oparciu o wnioski powstała aplikacja składająca się z ośmiu różnych modułów (każdy ukazuje dane w inny sposób). W tym rozdziale przedstawione zostaną kolejne metody składające się na stworzony pakiet do wizualizacji danych. W celu prezentacji tych rozwiązań zostały wykorzystane dane *„Iris Plants Database”*. Są one podzielone na trzy klasy (Setosa, Virginica, Versicolor), a każdy przypadek (wektor) określają cztery cechy (sepal lenght, sepal width, petal lenght, petal width).

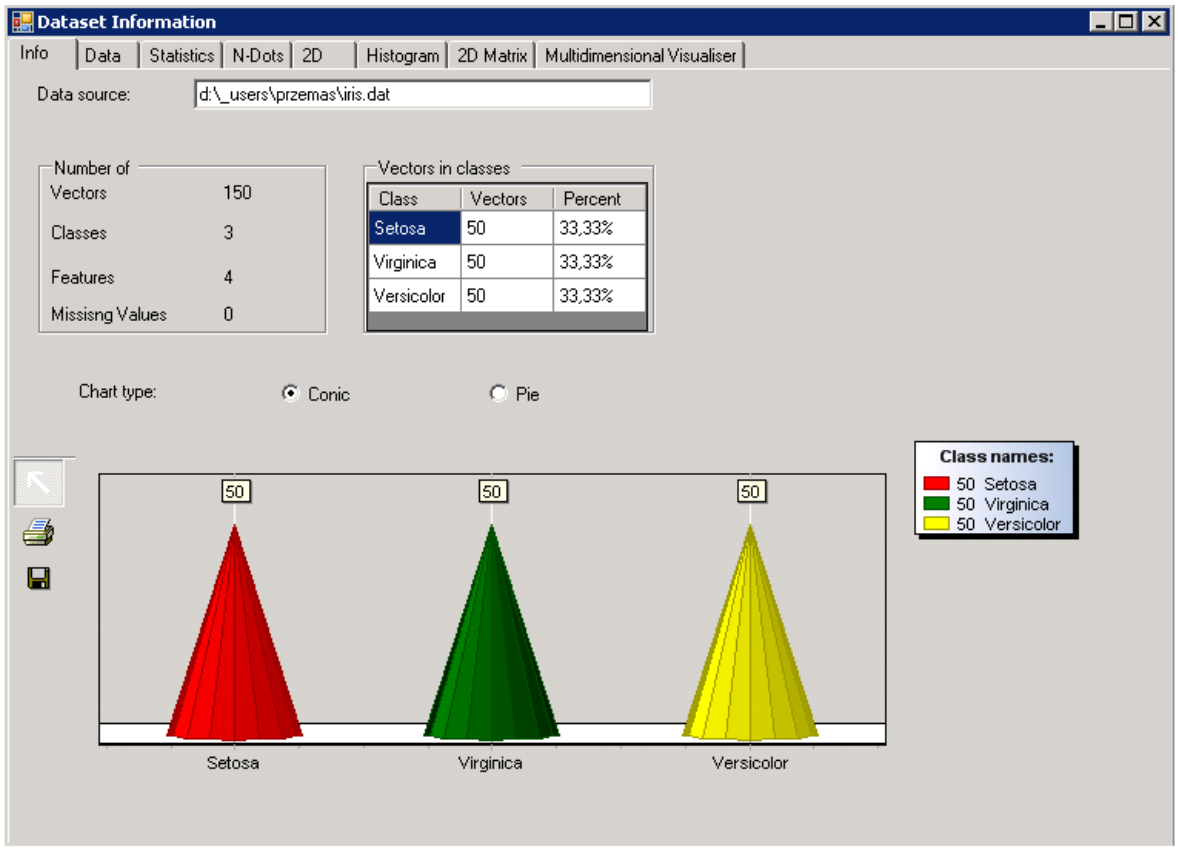
4.1. Struktura aplikacji

Aplikacja składa się, z ośmiu modułów. Każdy z nich zamieszczony jest na osobnej zakładce (rys. 4.1). Program zawiera nie tylko same wykresy, ale również dane numeryczne takie jak: korelacje, wartości średnie, wariancje, wartości brakujące itd. Zaimplementowane zostały także dodatkowe narzędzia, które mają na celu ułatwienie użytkownikowi analizę danych. Pozwalają one osobie korzystającej z pakietu na m.in.

modyfikowanie ustawień, dodawanie nowych wykresów lub działanie na podzbiorze danych.

4.1.1. Podstawowe informacje o zbiorze danych

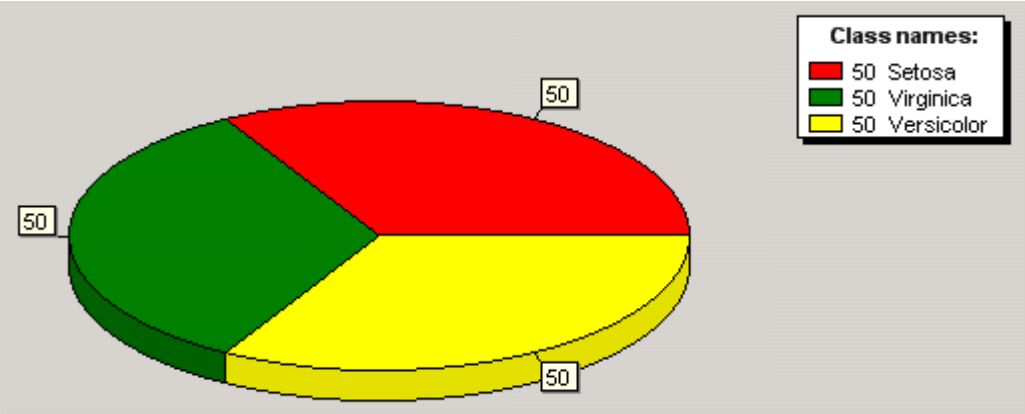
Podstawowe informacje o zbiorze danych są zawarte w dwóch zakładkach. Przeglądając zakładkę *Info* użytkownik może poznać: ścieżkę dostępu do zbioru danych, oraz ilości wektorów, klas, cech, brakujących wartości (rys. 4.1).



4.1 Informacje ogólne o zbiorze danych „Irirs Plants Database”

Zawiera ona również dwa wykresy, które przedstawiają ilość przypadków (wektorów) należących do poszczególnych klas. Zakładka służy do przybliżenia użytkownikowi danych, z którymi będzie pracował. Zastosowane metody wizualizacji, mimo iż są bardzo proste, w znacznym stopniu ułatwiają wstępne „spojrzenie” na dane. Osoba korzystająca z

aplikacji nie musi analizować liczb, ponieważ na wykresach są one wyraźnie wyeksponowane i wystarczy krótkie spojrzenie, aby dostrzec różnicę. Pierwszy wykres jest widoczny na dole rysunku 4.1, a drugi na rysunku 4.2.



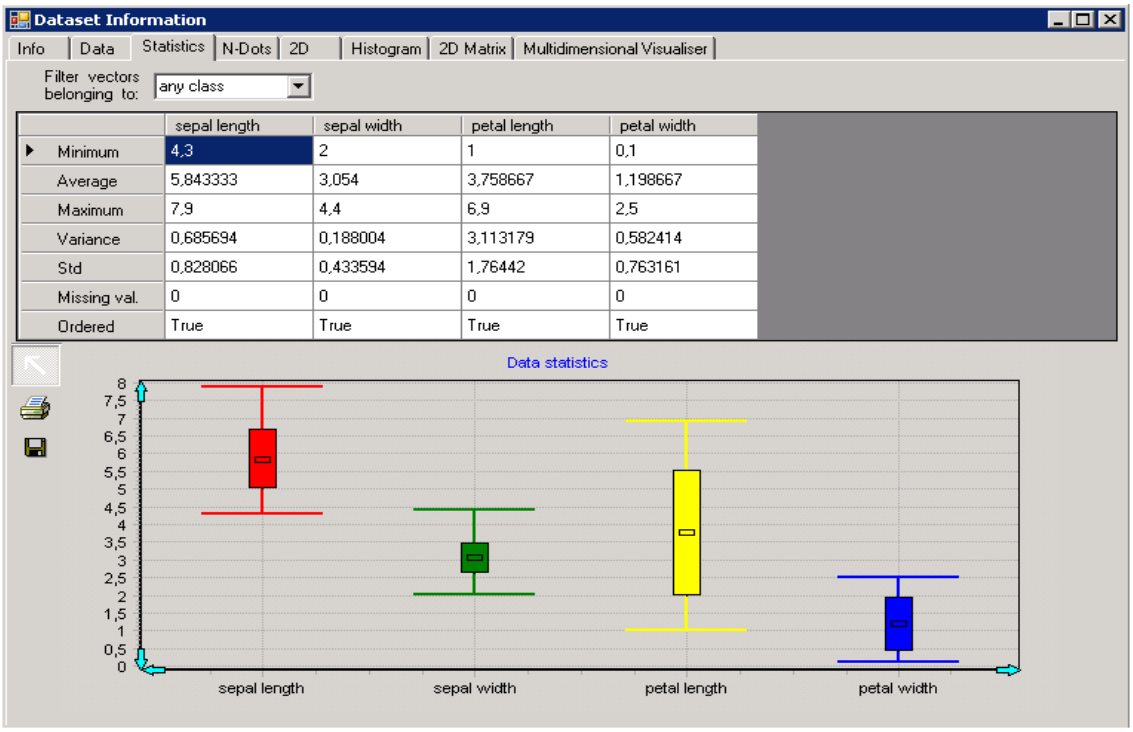
4.2 Alternatywny sposób wyświetlania ilości wektorów należących do poszczególnych klas.

Zakładka *Data* zawiera zbiór danych przedstawiony w postaci tabeli. Użytkownik może obejrzeć zarówno cały zbiór, jak i tylko wektory należące do jednej z klas. Umożliwione zostało również sortowanie przypadków według wybranej cechy.

Dzięki zakładkom *Info* oraz *Data* osoba korzystająca z pakietu może wstępnie zapoznać się ze zbiorem danych, co pozwala na dobranie odpowiednich metod do dalszej pracy. Możliwe jest również analizowanie wektorów w wersji oryginalnej. Jest to dość przydatne w sytuacji gdy na podstawie niżej przedstawionych metod odkrywamy interesujące nas przypadki i chcemy się im przyjrzeć w postaci numerycznej.

4.1.2. Statystyki cech

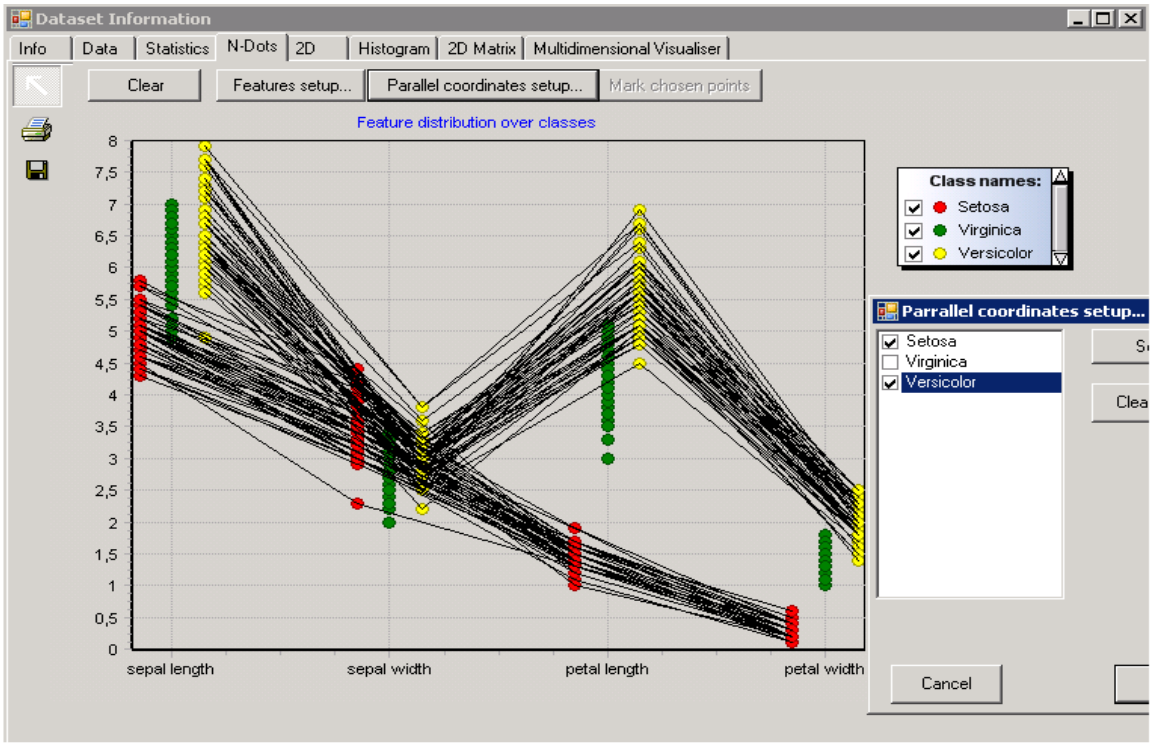
Statystyki cech, opisujących wektory należące do zbioru danych, są przedstawione w zakładce *Statistics* (rys. 4.3). Możliwe jest ich wyświetlanie zarówno dla całego zbioru jak i poszczególnych klas. W ostatnim wierszu tabeli, o nazwie *Ordered, jest* określone czy, dana cecha jest uporządkowana (wartość *true*), czy nie (*false)*. Jest to istotna informacja, która może wpłynąć na dalszą obróbkę danych (np. metoda *Multidimensional Visualiser* korzysta tylko i wyłącznie z cech nieuporządkowanych). Zastosowany wykres jest specyficzną postacią wykresu pudełkowego, zaznaczone na nim wartości to: wartość maksymalna, minimalna, średnia, a wielkość pudełka jest równa odchyleniu standardowemu w każdą ze stron licząc od wartości średniej. Jest to kolejny sposób na dosyć ogólne spojrzenie na dane, aczkolwiek zostają ujawnione pewne właściwości, które pozwalają wyciągnąć wnioski na temat budowy zbioru.



4.3 Statystyki cech zbioru „Iris Plant Database”.

4.1.3. Zbiór danych przedstawiony za pomocą współrzędnych równoległych

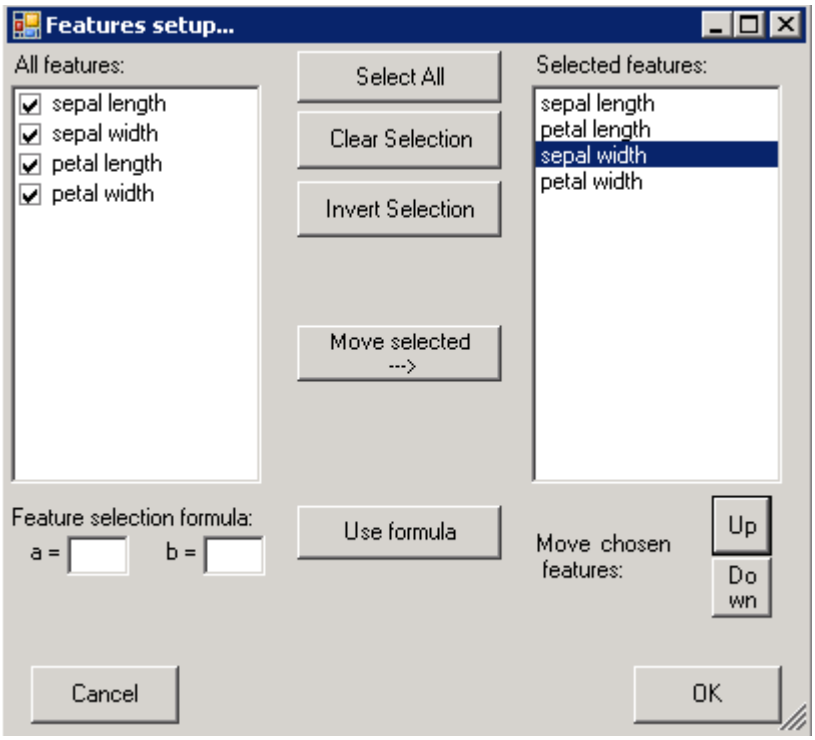
W zakładce *N-Dots* znajduje się wykres zawierający cały zbiór danych przedstawiony za pomocą współrzędnych równoległych (metoda omówiona w rozdziale 2.2.5.). Bezpośrednio po wczytaniu danych generowane są jedynie punkty. Wynika to z faktu, iż tworzenie kompletnego wykresu współrzędnych równoległych dla dużych zbiorów danych może być zbyt czasochłonne (zarówno ze względu na dużą ilość wektorów jak i cech je opisujących). Są dwa sposoby na utworzenie linii, które łączą odpowiednie punkty w wektory. Możemy wejść w menu o nazwie *Parallel coordinates setup* (rys. 4.4), co pozwoli wygenerować linie dla określonych klas. Drugi sposób polega na naciśnięciu lewego przycisku myszy na interesującym nas punkcie. Zostanie wtedy wygenerowana linia wektora, do którego należy ten punkt, a na legendzie zostanie podany jego numer (rys 4.5).



4.4 Współrzędne równoległe ustawione dla dwóch klas (Setosa, Versicolor), oraz menu konfiguracji współrzędnych równoległych

Po naciśnięciu przycisku *Features setup* pojawia się menu wyboru cech. Jest ono bardzo przydatne gdy mamy do czynienia z wieloma wymiarami, gdyż możemy wybrać tylko te, które nas interesują oraz uporządkować je w wybrany przez nas sposób. Pomagają nam w tym narzędzia takie jak *Select All* (zaznacza wszystkie cechy), *Clear Selection* (odznacza wszystkie), *Invert Selection* (zamienia zaznaczone na odznaczone i na odwrót) oraz *Use Formula*. Ostatnie z nich pozwala nam stworzyć prostą funkcję liniową postaci $aX + b$, która jest przydatna przede wszystkim, gdy mamy do czynienia z dużą ilością cech. Poprzez parametr *a* oznaczamy, co który element chcemy zaznaczyć, a poprzez parametr *b* oznaczamy, od którego zaczynamy wybór. Gdy zostaną wybrane interesujące cechy należy nacisnąć przycisk *Move selected,* następnie wybrane przez użytkownika wymiary zostaną przeniesione do okna znajdującego się po prawej stronie menu. Tam za pomocą klawiszy *Up* oraz *Down* możemy ustawić odpowiednią kolejność w jakiej zostaną one zamieszczone na wykresie.

4.5 Współrzędne równoległe dla dwóch wybranych wektorów o numerach 108 i 98



4.6 Menu wyboru cech

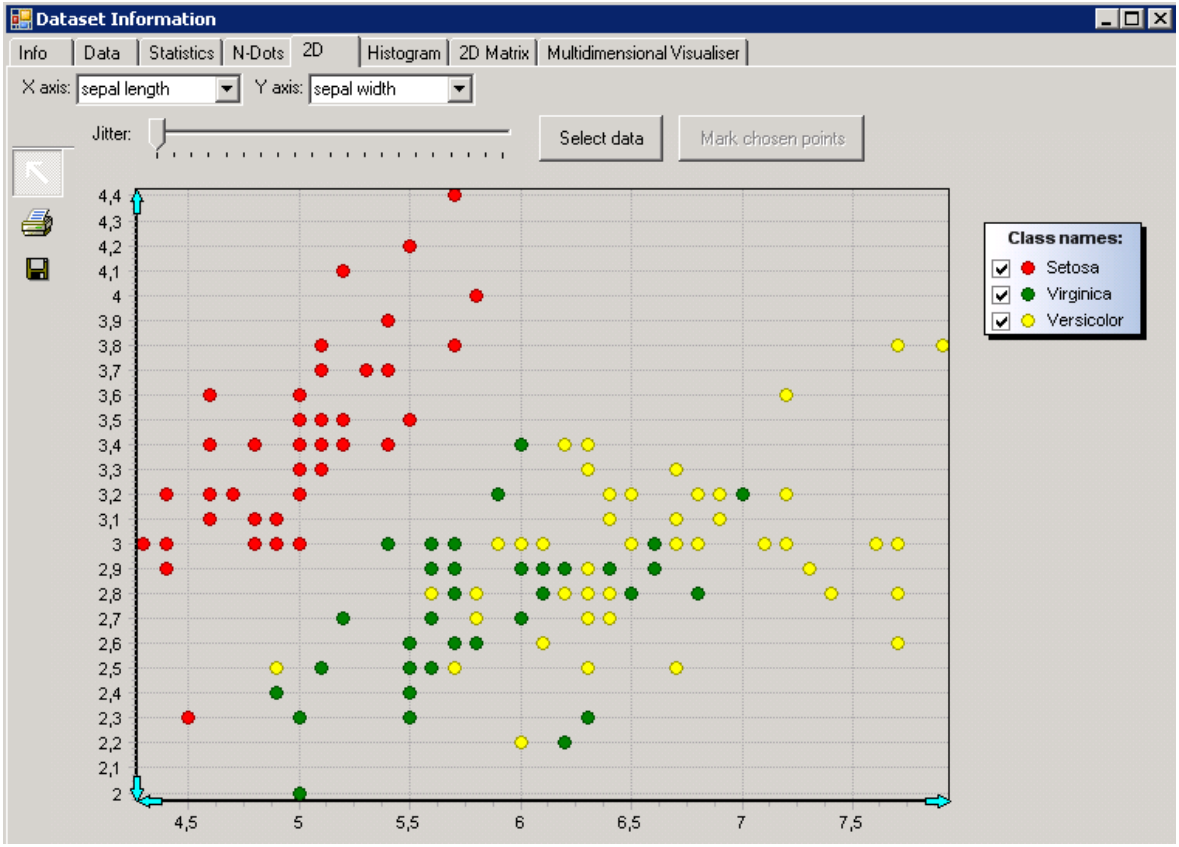
Istnieje jeszcze przycisk *Mark chosen points* (widoczny na rys. 4.4), który służy do wyznaczenia wektorów wybranych podczas analizowania wykresów dwuwymiarowych (dokładniej wyjaśnione w 4.1.4.).

Współrzędne równoległe pozwalają osobie analizującej dane zobaczyć jak gęsto rozkładają się wartości poszczególnych cech. Dzięki zaimplementowanym narzędziom można również podejrzeć tylko interesujące wymiary oraz zauważyć wiele prawidłowości. Mamy do wyboru różne poziomy szczegółowości, ponieważ można oprzeć badania zarówno na całym zbiorze, tylko na określonych klasach bądź wręcz na kilku wektorach. Istnieje również możliwość działania na wybranych wymiarach. Metoda pozwala na odkrycie cech mocno skorelowanych i wykluczenie ich z dalszych badań. Dzięki linią użytym w tej metodzie możemy zauważyć prawidłowości dla poszczególnych klas bądź cech (np. po wyświetleniu linii dla całego zbioru łatwo dostrzec wartości cech krzyżujące

się, czyli gdy dla cechy pierwszej liczba prezentująca dany wymiar jest bliska maksymalne to dla cechy drugiej liczba jest bliska minimalnej).

4.1.4. Rzutowanie danych na dwa wymiary

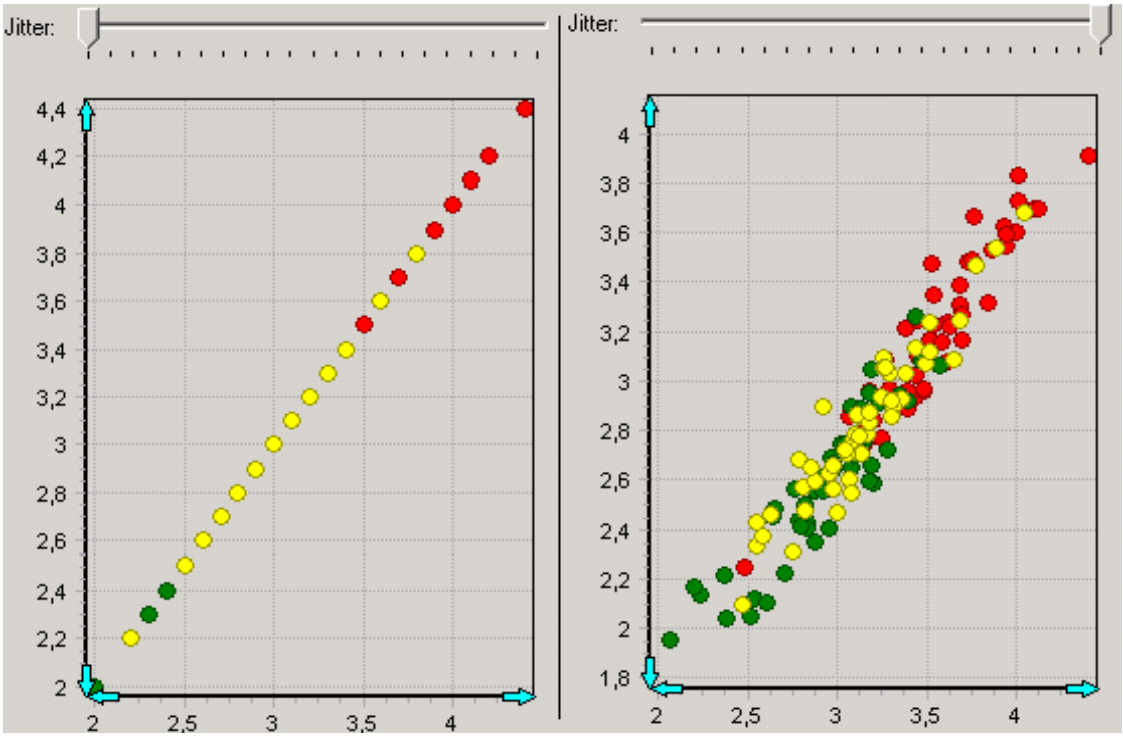
W tym przypadku zastosowane zostały wykresy rozproszone. Jak widać na rysunku 4.7 zaimplementowane zostały również dodatkowe funkcje. Ułatwiają one analizę utworzonego wykresu jak i pozwalają na jego modyfikację oraz wybór danych.



4.7 Zakładka wyświetlająca wykresy rozproszone

Jak już było wspomniane w rozdziale drugim, podczas kreowania wykresów rozproszonych często mamy do czynienia z nakładającymi się punktami. Wówczas nie zawsze jesteśmy w stanie stwierdzić, gdzie skupia się większa ilość danych. W celu rozwiązania tego problemu został stworzony suwak odpowiedzialny za stopień „drżenia”

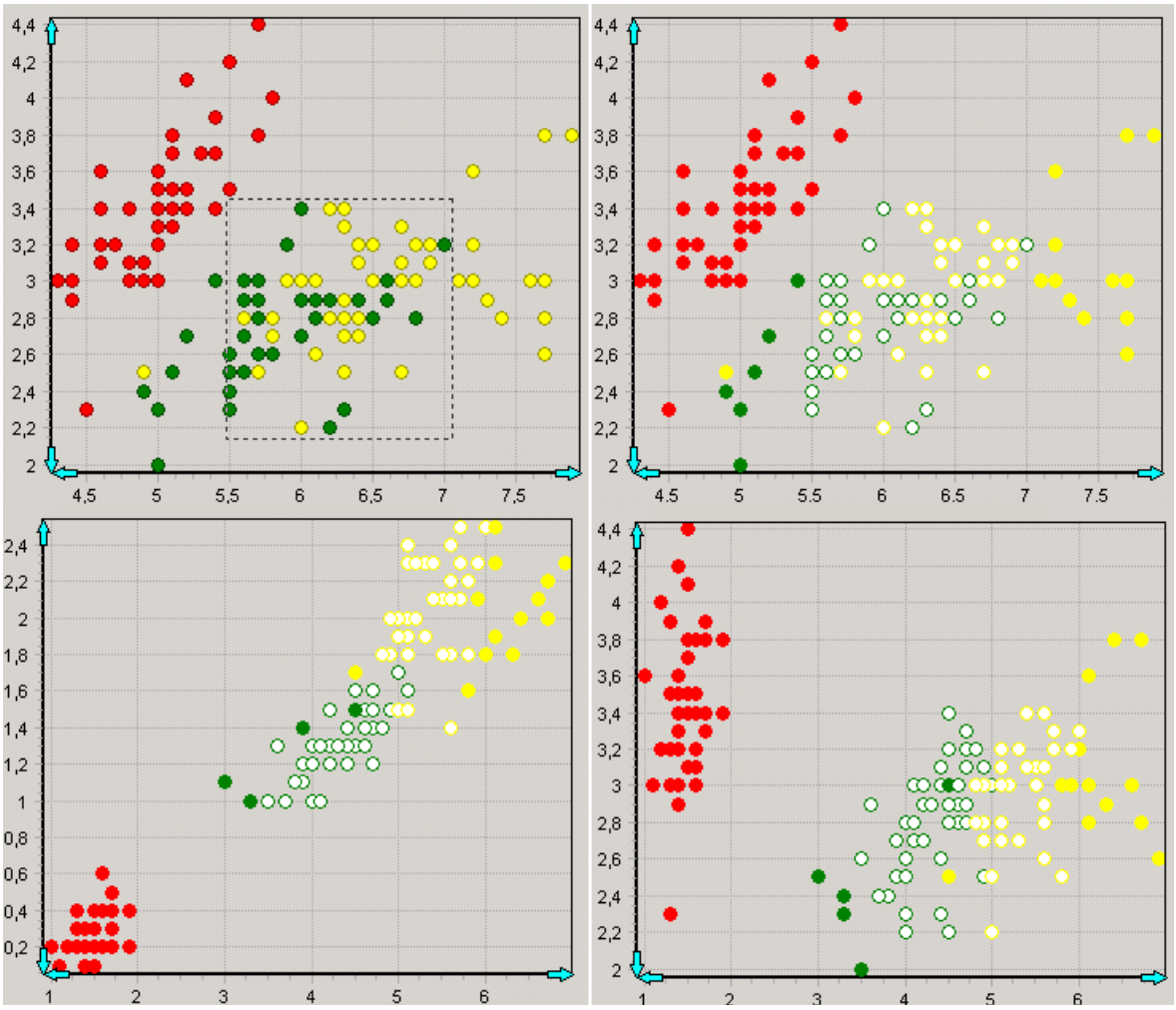
danych (ang. *jitter*). Za każdym razem gdy zostaje on przesunięty kolejne punkty na wykresie są przesuwane w różnych (losowych) kierunkach. Z każdym przesunięciem suwaka w prawo rozproszenie punktów staje się większe, a w lewo mniejsze. Efekt jego działania najlepiej obrazuje przykład zilustrowany na rysunku 4.8. Na wykresie po lewej stronie wydaje się, że punktów jest kilkanaście, a w rzeczywistości jest ich dużo więcej (wykres po prawej).



4.8 Zastosowanie suwaka odpowiedzialnego za „drżenie" danych.

Istnieje również możliwość wyboru danych, które chcemy poddać dalszej wizualizacji. W tym celu należy wybrać (przybliżyć) interesujący nas obszar po czym nacisnąć przycisk *Select. S*powoduje to zapisanie wybranych wektorów do pliku (co umożliwi przegląd wybranego podzbioru). Można również podejrzeć zaznaczone przez użytkownika przypadki w kontekście całego zbioru. Wystarczy naciśnięcie przycisku *Mark chosen points* wtedy wybrane wektory zostaną wyselekcjonowane w sposób widoczny na rysunku

4.9. Może to dosyć istotnie pomóc w wyborze dalszych metod analizy oraz stworzeniu nowych reguł klasyfikacji danych. Wektory te możemy również obejrzeć w zakładce *N-Dots* po naciśnięciu przycisku o takiej samej nazwie – *Mark chosen points* (4.1.3.).



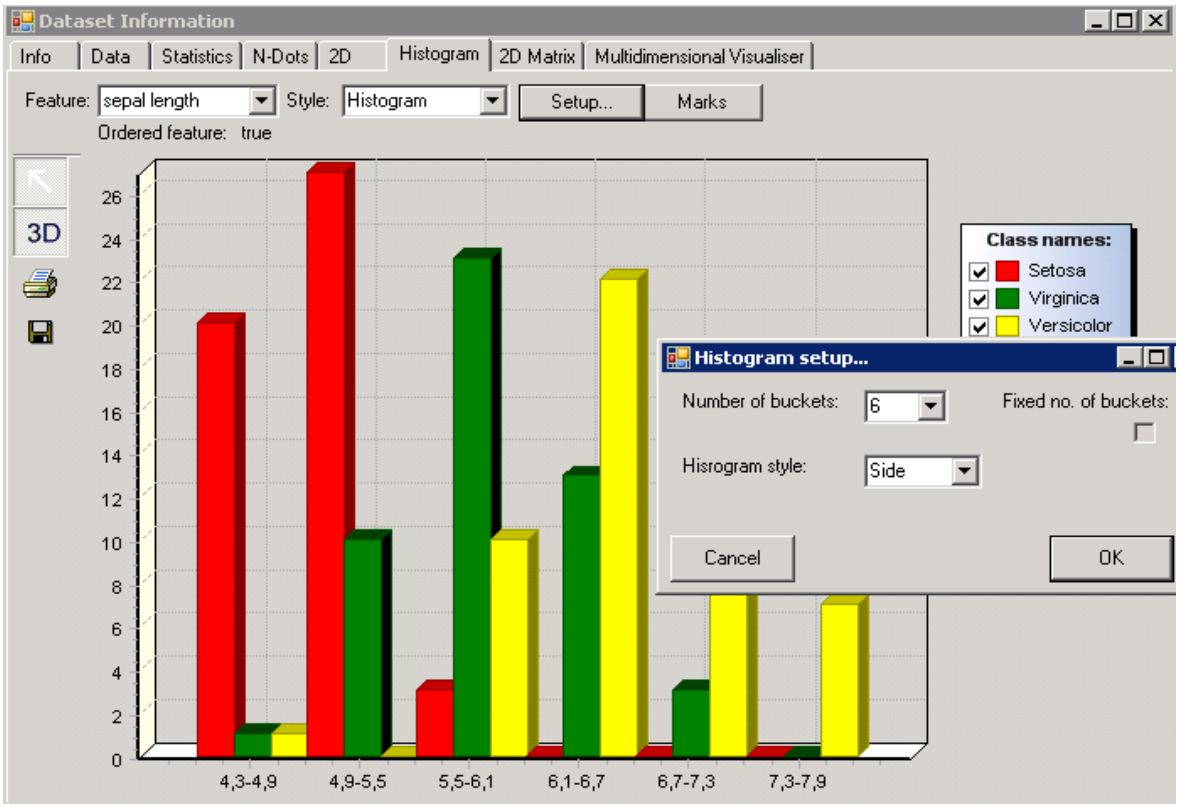
4.9 Wykresy rozproszone pokazujące zastosowanie przycisku Select

Powyżej zostało zaprezentowane przykładowe użycie opisanej wcześniej metody. Na wykresie po lewej górnej stronie (oś X: *sepal lenght*, oś Y: *sepal width*) zaznaczono problematyczne dane (ciężko w tym obszarze na podstawie wybranych cech stwierdzić czy wektor należy do klasy *Virginica* czy *Versicolor*). Dlatego przypadki te zostały wyszczególnione prawa górna strona rysunku. Z kolei poniżej wybrane zostały inne cechy, aby znaleźć dobry klasyfikator dla wektorów znajdujących się w wybranym obszarze. Od

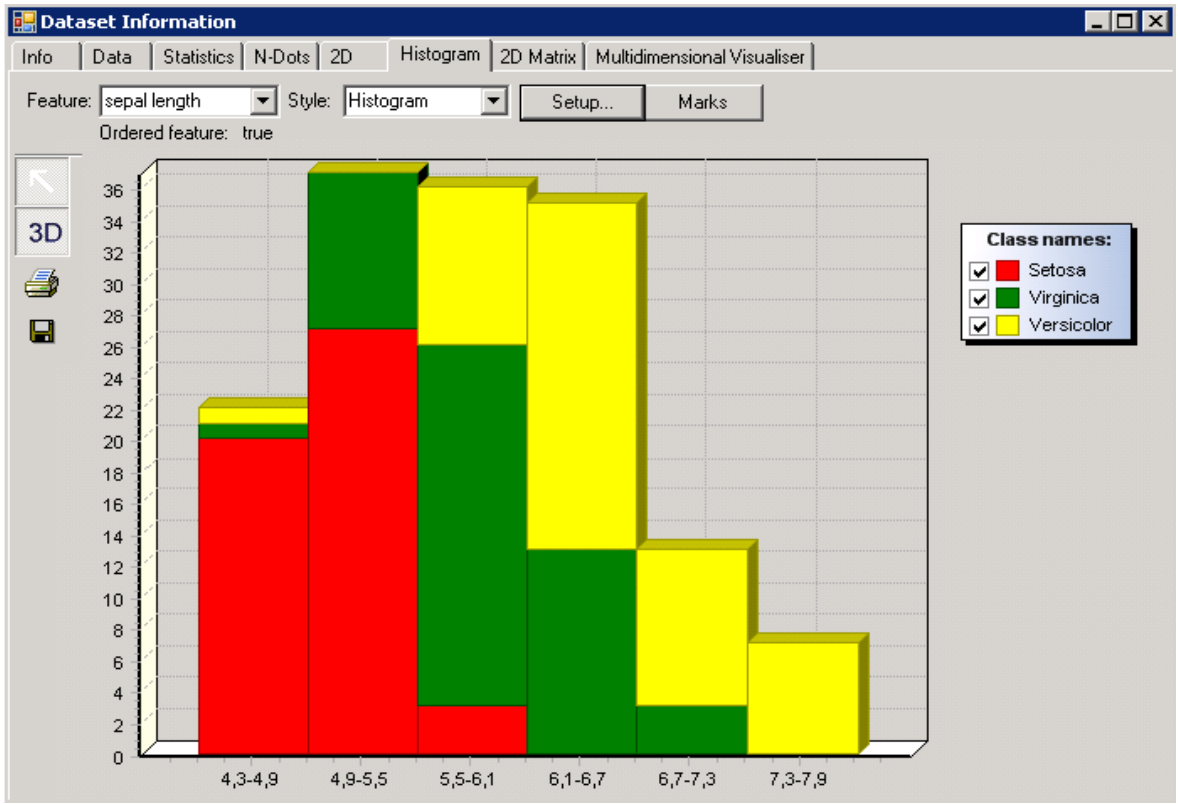
razu można zauważyć, że lepiej jest wybrać cechy tworzące wykres z lewej (oś X: *petal lenght*, oś Y: *petal width*) niż ten z prawej (oś X: *petal lenght,* oś Y: *sepal width*). Możemy również skorzystać z utworzonego podzbioru, na który składają się wyłącznie wybrane przypadki i znaleźć klasyfikator wyłącznie dla interesującego nas obszaru bez obawy, iż dane do niego nie należące przeszkodzą w analizie.

4.1.5. Histogramy

W zakładce *Histogram* zaimplementowane zostały dwie metody wizualizacji. Obydwie mają na celu przybliżenie rozkładu wybranego wymiaru. Wszystko zależy od tego, czy określona przez użytkownika cecha jest uporządkowana, czy nieuporządkowana. W pierwszym przypadku tworzony jest histogram z liczbą przedziałów równą liczbie wartości jakie mają wektory w podanym wymiarze. Istnieje możliwość samodzielnego wyboru ilości przedziałów. Może okazać się to niezbędne w przypadku, gdy cecha przyjmuje bardzo dużo różnych wartości, ponieważ wykres ze zbyt dużą ilością „słupków" może okazać się nieczytelny. Kolejnym ułatwieniem jest możliwość wyboru dwóch sposobów wyświetlania histogramów: *stacked* oraz *side*. Domyślnie ustawiony jest *side* i wyświetla on osobny „słupek" dla każdej z klasy (rys. 4.10). Z kolei *stacked* tworzy klasyczny histogram z podziałem na klasy (rys. 4.11).



4.10 Histogram dla cechy sepal lenght z ustaloną liczbą sześciu przedziałów i stylem side



4.11 Histogram z rysunku 4.10 w stylu stacked

W przypadku gdy cecha jest ciągła utworzony zostaje wykres liniowy (rys. 4.12), który przybliża jej rozkład dla każdej klasy. W celu uzyskania takiego efektu zaimplementowany został algorytm opierający się na metodzie „okienek Parzena". Polega ona na stworzeniu przedziału o, określonej przez użytkownika lub domyślnej, szerokości. Następnym krokiem jest ustalenie środka pierwszego przedziału (wynosi on wartość minimalną dla danej cechy plus połowa wielkości przedziału). „Okienko Parzena" przesuwamy w dość prosty sposób (według niżej opisanej w pseudokodzie procedury), który pozwala dosyć dokładnie przybliżyć rozkład cechy:

```
count_Center(array, min, max, center, win_Size)
        minimum = center – 0,5*win_Size
        maximum = center + 0,5*win_Size
        difference1 = array[min] – minimum
        difference2 = array[max +1] – maximum
        if (difference1 < difference2)
                center = center + difference1
                return center
        else
                center = center + difference2
                return center
```

Funkcja odpowiedzialna za ustalanie kolejnych centrów okien na wejściu dostaje pięć parametrów: posortowaną tablicę kolejnych wartości cechy, indeks najmniejszego elementu mieszczącego się w oknie, indeks największego elementu mieszczącego się w oknie, środek okna oraz rozmiar okna. Procedura polega na obliczeniu dwóch odległości. Pierwsza jest to dystans między początkiem okna (minimum), a najmniejszą wartością do niego należącą. Druga to odległość między końcem okna (maksimum), a następną wartością znajdującą się w tablicy. Następnym krokiem jest wybór mniejszej z nich oraz przesunięcie o właśnie tą odległość środka okna. Dzięki tak skonstruowanej procedurze przedział jest przesuwany tak aby podczas każdego kolejnego kroku wpadała bądź była usuwana z niego tylko jedna wartość (w przypadku równych odległości mogą te dwa zdarzenia nastąpić jednocześnie).

Aplikacja oferuje użytkownikowi również trzy miary określające ilość wektorów wpadających do takiego przedziału (wyboru dokonujemy w menu *Parzen Window setup* – rysunek 4.12). Pierwsza „kwadratowa" polega na zliczeniu przypadków, które znajdują się w oknie. W pozostałych dwóch każdemu wektorowi wpadającemu do odpowiedniego przedziału jest przypisywana waga, która zależy bezpośrednio od jego odległości od środka okna. I tak w mierze „trójkątnej" każda z wag jest obliczana wzorem 4.1:

$$w = \frac{array[i] - center + 0{,}5 * winSize}{center - 0{,}5 * winSize} \quad , \tag{4.1}$$

jeżeli $array[i] < center$, w przeciwnym przypadku

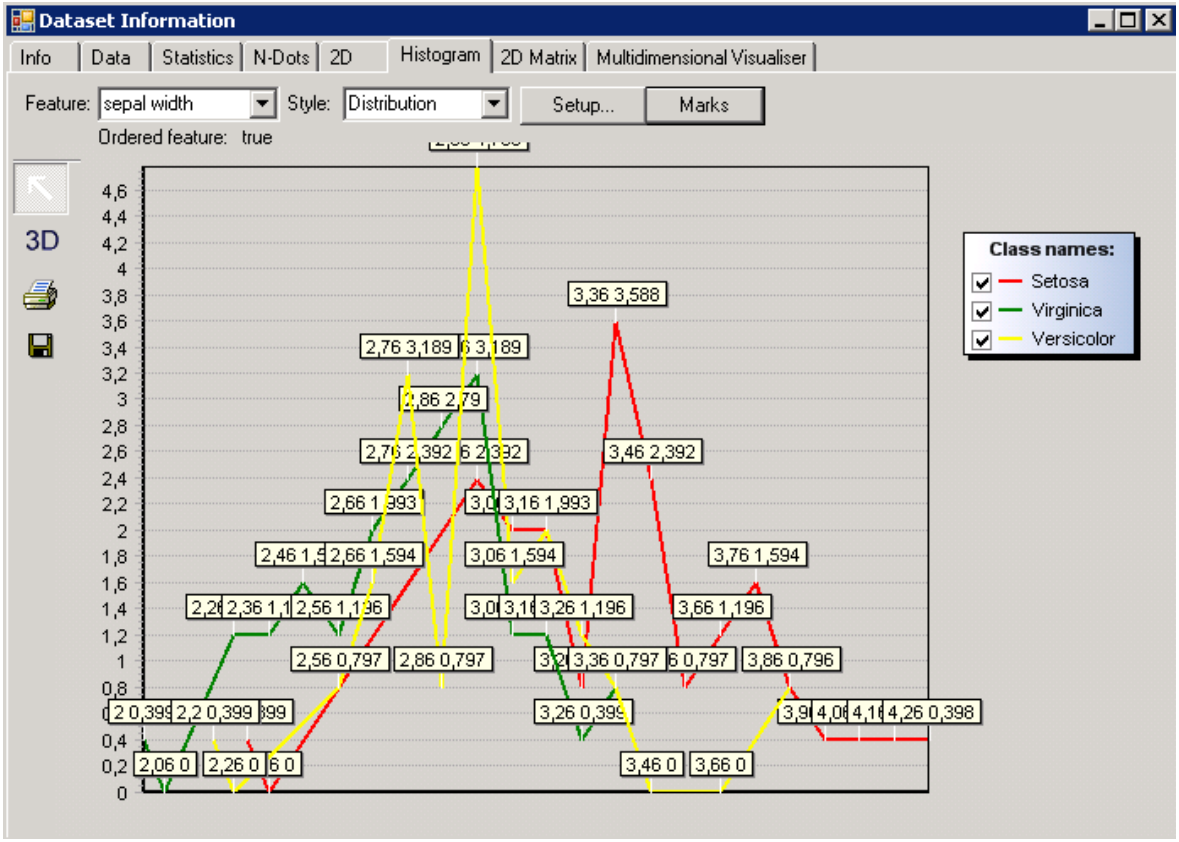
$$w = \frac{center + 0{,}5 * winSize - array[i]}{center - 0{,}5 * winSize} \quad , \tag{4.2}$$

gdzie *w* oznacza obliczoną wagę, *center* środek okna, dla którego jest robione obliczenie, *winSize* wielkość okna, a *array[i]* element, dla którego obliczamy wagę.

Miarę „gaussowską” obliczamy według wzoru 4.3:

$$w = \frac{1}{\sqrt{2\pi}} e^{\frac{-(array[i]-\mu)^2}{2}}, \tag{4.3}$$

gdzie za μ podstawiamy liczbę równą środkowi okna, dla którego waga jest liczona. Posiadanie trzech różnych miar pozwala użytkownikowi na samodzielne ustalenie jak bardzo jest dla niego istotna odległość wektora od centrum przedziału. Dzięki temu sami decydujemy o stopniu wygładzenia wykresu i ilości wykonywanych obliczeń.



4.12 Rozkład cechy sepal width przy miarze „gaussowskiej” z włączoną opcją Marks

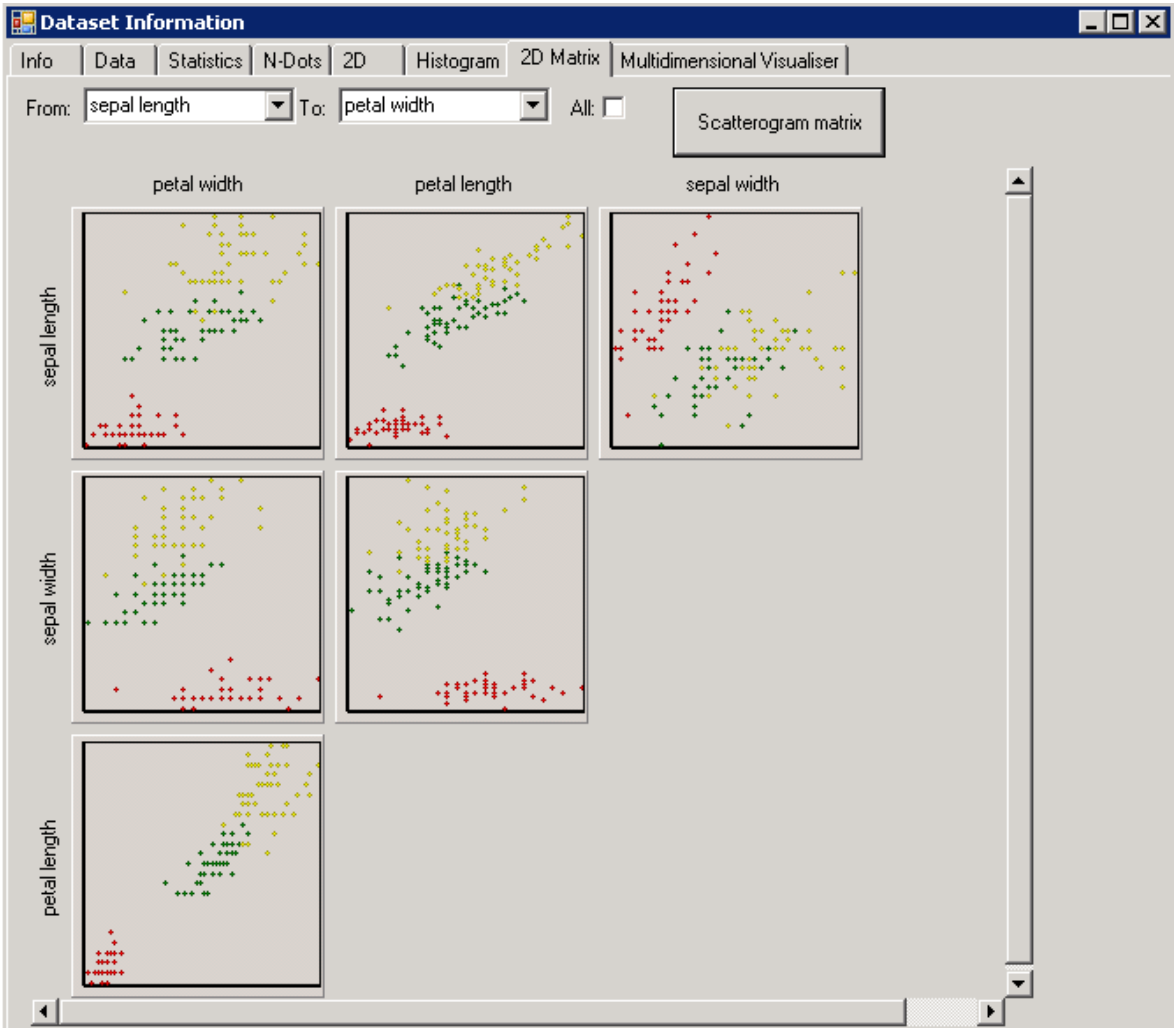
Jeżeli cecha jest ciągła aplikacja umożliwia korzystanie zarówno z histogramów jak i wykresów liniowych, w przeciwnym przypadku dostępne są jedynie histogramy (nie ma

sensu stosowania drugiej metody). Dzięki przyciskowi *Marks* możliwe jest wyświetlanie informacji na temat wykresu bezpośrednio na nim np. środki przedziałów i przyporządkowane im liczby (rys. 4.12).

Metody zaimplementowane w module *Histogram* pozwalają na dość dokładne przybliżenia rozkładów poszczególnych wymiarów. Jest to bardzo istotne w kontekście dalszej analizy. Dzięki temu możemy „na oko" stwierdzić czy rozkład danej cechy jest np. normalny, co umożliwia zastosowanie wielu algorytmów. Znając rozkład można m.in. uzupełnić wartości brakujące.

4.1.6. Macierz wykresów rozproszonych

Macierz wykresów rozproszonych jest bardzo przydatnym narzędziem. Oglądając całość bądź część takiej macierzy możemy zauważyć, które cechy warto wybrać do stworzenia wykresu rozproszonego, które z nich dobrze separują klasy itd.. Jednak ze względu na często dużą ilość wymiarów w analizowanych zbiorach danych nie można sobie pozwolić na domyślną wizualizację takiej macierzy w całości. W przypadku tysiąca wymiarów użytkownik na pewno musiałby długo czekać na efekt. Dlatego stworzony pakiet pozwala na wybranie zakresu cech dla jakich chcemy stworzyć macierz, bądź zaznaczenie opcji *all* i wyświetlenie całości (rys.4.13).

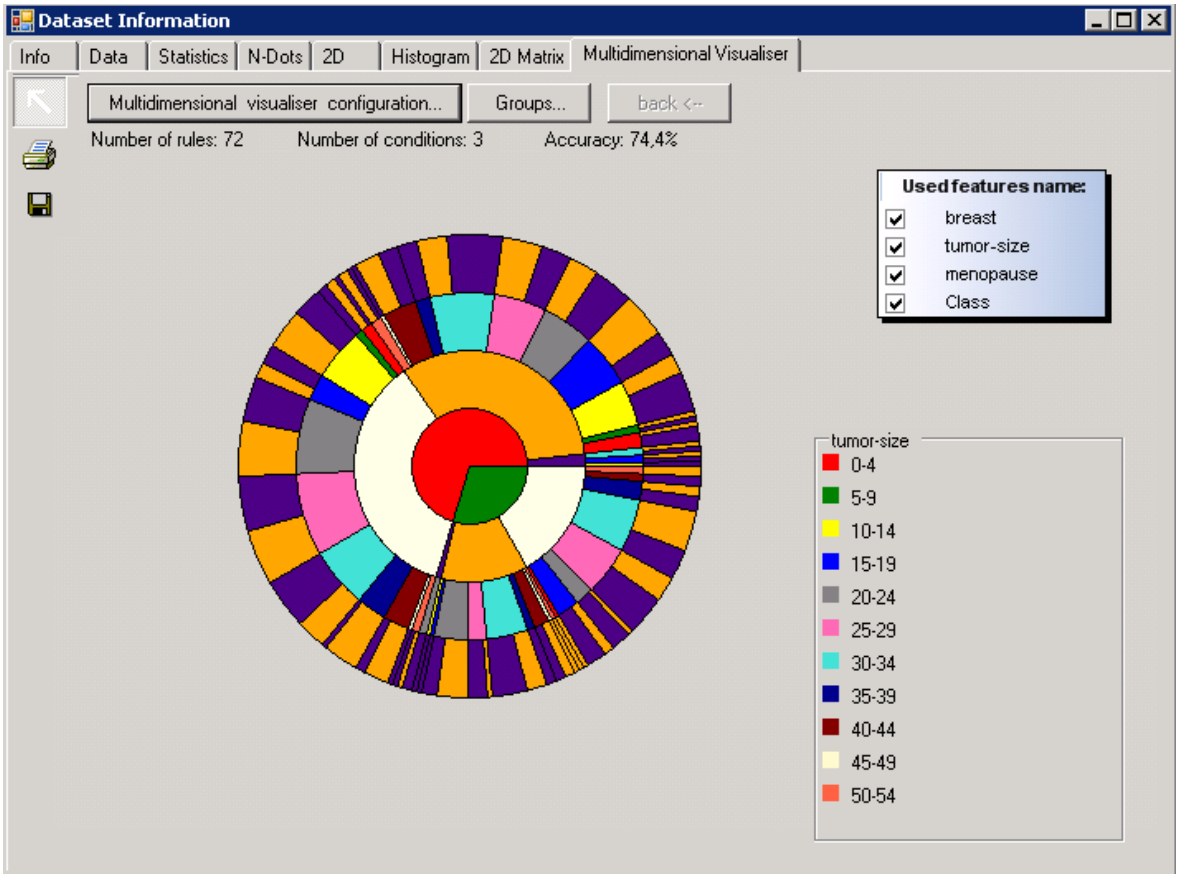


4.13 Macierz wykresów rozproszonych.

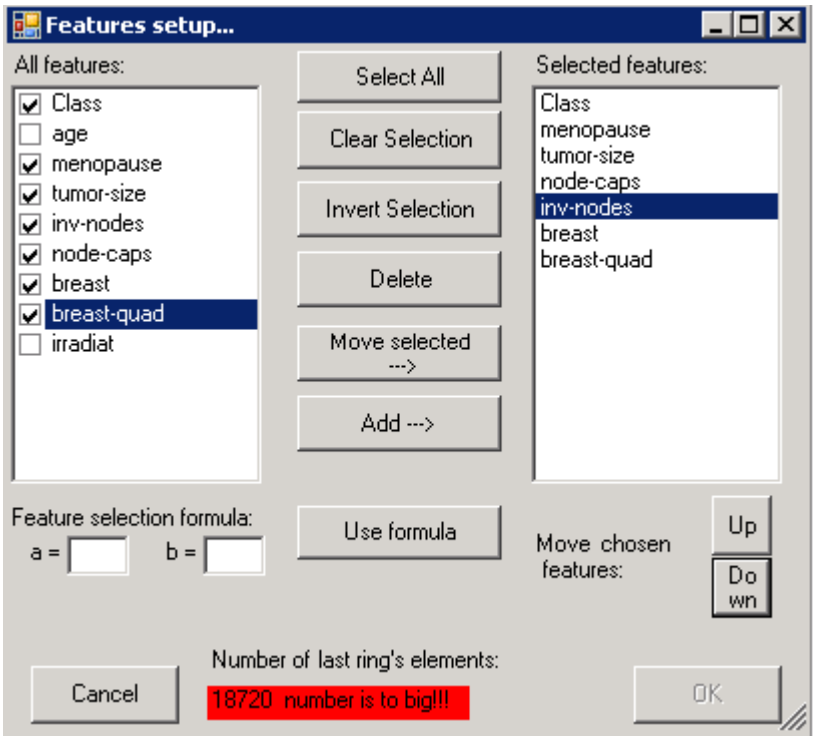
Jak widać powyżej tak naprawdę wizualizowana jest tylko połowa macierzy, ponieważ jest ona symetryczna i nie ma potrzeby pokazywania całości. Nazwy kolumn odpowiadają wartością znajdującym się na osiach współrzędnych X-ów, wierszy Y-ów. Po kliknięciu myszą na wybrany wykres zostanie od wyświetlony w zakładce *2D*. Pozwala to na płynne przemieszczanie się pomiędzy tymi dwoma, mocno ze sobą związanymi, modułami.

4.1.7. Wizualizator cech nieuporządkowanych

Zakładka *Multidimensional Visualiser* zawiera dość innowacyjne podejście do wizualizacji danych. Wyniki działania modułu zostały zaprezentowane na podstawie zbioru danych *Lbreast*, ponieważ zawierają one dużą ilość cech nieuporządkowanych. Metoda (rys.4.14) polega na rysowaniu okręgów. Po wciśnięciu przycisku *Multidimensional visualiser configuration...* ukaże się menu, po lewej stronie stworzona zostaje lista wszystkich cech nieuporządkowanych oraz klasy. Użytkownik ma możliwość wybrania kilku, ustawienie ich w określonym przez siebie porządku, dodawanie do utworzonego wykresu itd. (rys.4.15). W dolnej części menu wyświetlana jest informacja ile elementów będzie miał zewnętrzny okrąg, jeżeli będzie ich więcej niż tysiąc utworzenie wykresu nie jest możliwe.



4.14 Wykres kołowy oraz kolory odpowiadające wartością cechy „tumor-size".



4.15 Menu wyboru cech.

Po utworzeniu wykresu (jak widać na rysunku 4.14) zostają obliczone trzy wartości, które ułatwiają analizę: wartość *Number of rules* oznacza ilość reguł jakie zostały stworzone (liczba elementów okręgu zewnętrznego), *Number of condotions* jest to ilość przesłanek (okręgów wewnętrznych), oraz *Accuracy* czyli dokładność obliczamy wzorem 4.4:

$$Acc = \frac{\sum_{1}^{m} acc_i}{m} \tag{4.4}$$

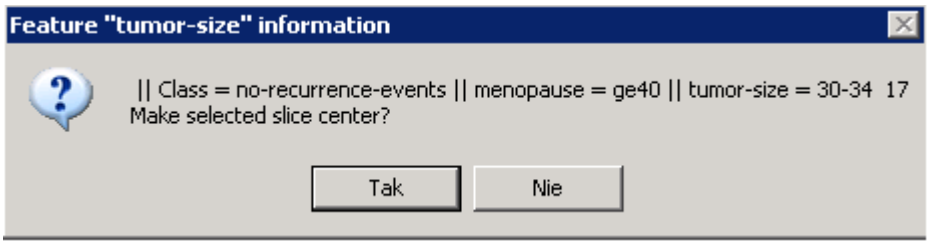
gdzie:

$$acc_j = \frac{r_k}{\sum_{0}^{R_j} r_i} \tag{4.5}$$

, przez Acc oznaczmy dokładność całkowitą (wzór 4.4) jest ona sumą dokładności cząstkowych obliczanych według wzoru 4.5. Aby obliczyć acc_j wyznaczamy zbiór reguł R_j , które spełniają określone przesłanki $c_{1j} \wedge c_{2j} \wedge ... \wedge c_{nj}$. Wybieramy regułę,

która zawiera najwięcej przypadków r_k oraz dzielimy przez ilość przypadków należących do reguł z R_j .

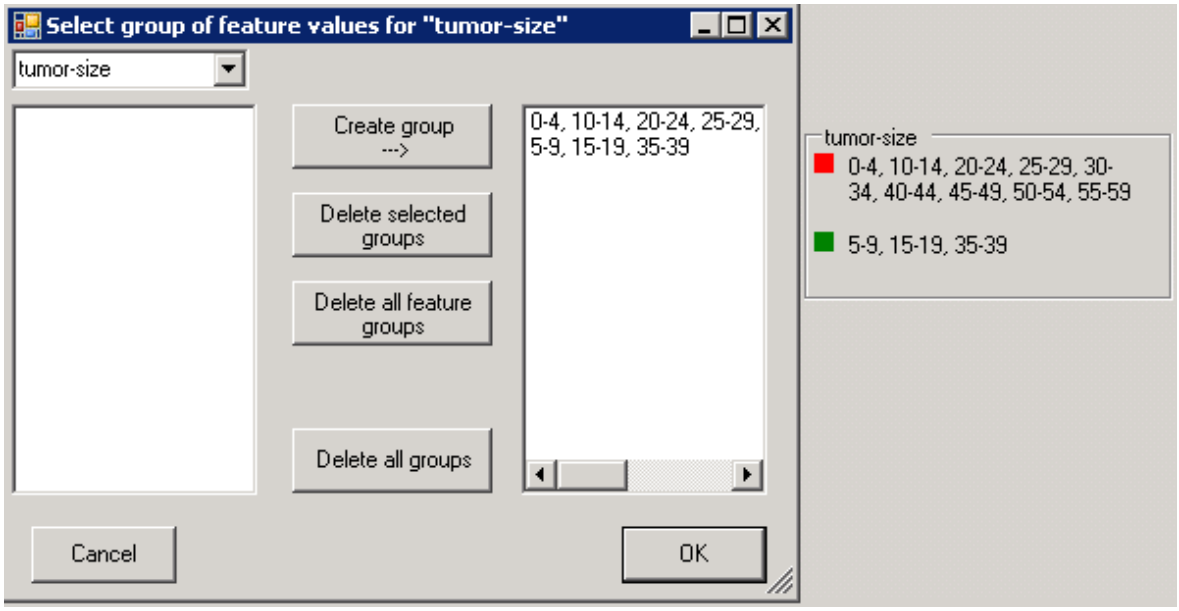
Tworzenie wykresu zaczynamy od pierwszej wybranej przez użytkownika cechy, okrąg reprezentuje cały zbiór i jest podzielony na części, które odpowiadają różnym wartością wybranej cechy (pojedynczym lub kilku naraz, jeżeli zostały utworzone grupy cech). Wizualizacja kolejnej cechy jest zależna od cechy poprzedniej ze względu na to, iż dzielimy wcześniej utworzone podzbiory na ilość części równą liczbie wartości kolejnego wymiaru. Jeżeli użytkownik chce zobaczyć ile jest elementów w danym podzbiorze wystarczy kliknąć myszą na określony kawałek, a ukaże okno dialogowe z dokładną informacją jakie cechy są brane pod uwagę, jakie przyjmują wartości oraz ile dokładnie ich jest w podzbiorze (rys 4.16).



4.16 Okno dialogowe z informacją o określonym kawałku trzeciego okręgu z rys. 4.14.

Oprócz informacji okno dialogowe zawiera pytanie „*Make selected slice center?*”, poprzez naciśnięcie przycisku *Tak* możemy na wykresie obejrzeć tylko wybrany przez nas podzbiór. Przycisk „*back* <--”, staje się aktywny i dzięki niemu możemy wrócić do poprzedniego wykresu. Jeżeli naciśniemy *Nie* wartości wybranego przez nas okręgu zostaną wyświetlone wraz z kolorami, które im odpowiadają (widoczne na rys 4.14). Jak już wcześniej zostało wspomniane wartości cech można grupować dzięki menu *Groups*. Po lewej stronie okna dialogowego znajdują się wartości jakie przyjmuje cecha, poprzez wybranie kilku i naciśnięcie przycisku *Create Group* łączymy cechy i na wykresie będą

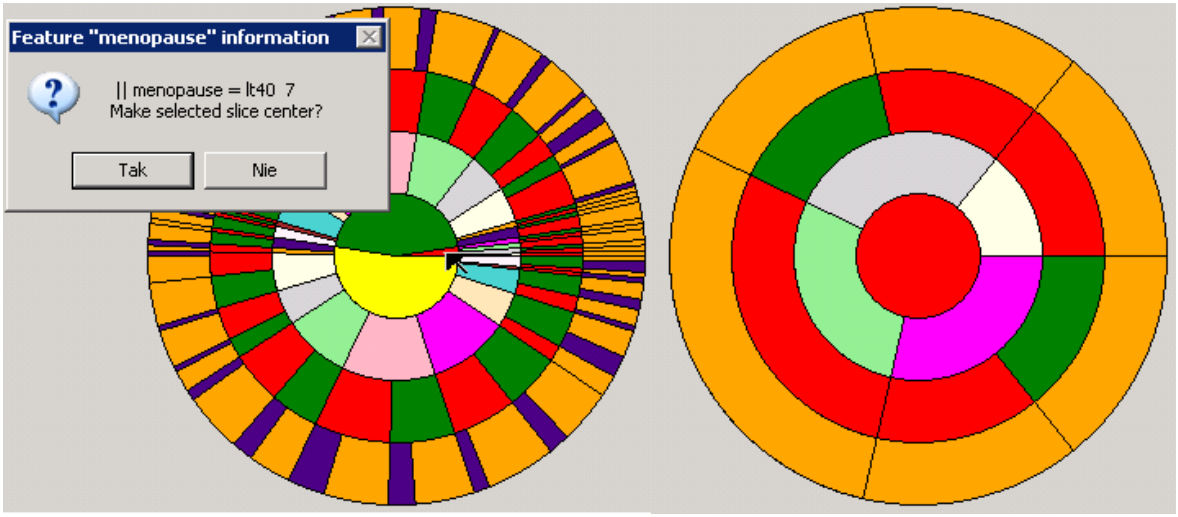
oznaczane jako jedna (rys 4.17).



4.17 Menu wyboru grup wartości dla cechy „tumor-size”.

Jak widać powyżej zostały wybrane stworzone dwie grupy, pierwsza na wykresie zostanie oznaczona kolorem czerwonym, a druga zielonym.

Jest to nowy sposób na wizualizację, którego inspiracją były drzewa decyzji. Metoda wizualizuje cały zbiór i dzięki niej możemy zobaczyć kilka wymiarów naraz. Można dostrzec również jak rozkładają się wartości poszczególnych cech. Łatwo znaleźć zarówno cechy są skorelowane, jak i nieskorelowane. Badania można prowadzić na różnych poziomach szczegółowości, istnieje możliwość podglądania tylko części zbioru (rys. 4.18).



4.18 Dwa wykresy kołowe.

Powyższe wykresy ukazują możliwości utworzonego narzędzia do wizualizacji. Wykres po lewej jest kombinacją czterech cech (zaczynając od najmniejszego okręgu: *„menopause”, „tumor-size”, „breast”, „irradiant”*). Jeżeli cecha *„menopause”*, przyjmuje wartości oznaczone kolorami żółtym oraz zielonym, dalszy rozkład jest dosyć czytelny jednak wyciągnięcie wniosków na temat siedmiu wektorów jest prawie niemożliwe. Wykres po lewej stronie został stworzony poprzez kliknięcie myszą na element oznaczony strzałką. Pozwala on na bardziej szczegółową analizę określonego podzbioru danych. Od razu można zauważyć, że jeżeli cecha *„menopause”* przyjmuje wartość *„lt40”* to *„irradiant”* jest równa *„no”*.

5. Podsumowanie

Stworzenie pakietu wizualizacyjnego na potrzeby nowego systemu służącego do eksploracji danych – Intemi zostało poprzedzone, zarówno analizą metod dostępnych w tego typu aplikacjach, jak i licznymi konsultacjami z osobami z nich korzystającymi. Dzięki temu zaimplementowane metody są połączeniem technik istniejących i sprawdzających się, oraz nowych rozwiązań zaproponowanych w trakcie przygotowania się do tworzenia projektu (np. nowa metoda do wizualizacji *Multidimensiona visuliser* do pokazania zależności między cechami nieuporządkowanymi).

Metody opierają się na technikach Eksploracyjnej Analizy Danych, która jest podstawowym podejściem do problemu wizualizacji. Każda z stworzonych metod dostarcza użytkownikowi istotnych informacji w kontekście analizy danych. Dzięki narzędziom oferowanym przez stworzoną aplikację można uzyskać podstawowe informacje na temat badanego zbioru oraz przeprowadzić bardziej szczegółowe badania. Każda z metod (histogramy, wykresy pudełkowe, wykresy rozproszone, współrzędne równoległe, wizualizator do cech nieuporządkowanych) pozwala na wizualizację zarówno całego zbioru jak i pojedynczych klas. Można zająć się kilkoma wektorami (współrzędne równoległe, wykresy rozproszone), lub delikatnie zmienić strukturę danych w celu lepszej analizy. Z kolei macierz wykresów rozproszonych pozwala na większy przegląd sytuacji. Dzięki tym narzędziom analiza danych jest w wielu przypadkach intuicyjna, ponieważ wystarczy chwilowe spojrzenie na wykres bądź kilka z nich, aby odkryć jaki jest model

danych i jakie metody należy zastosować do dalszych badań. Struktura aplikacji pozwala także na użycie jej jako osobnego narzędzia. Jedynym wejściem są dane i nie ma znaczenia czy są one w postaci oryginalnej czy zmodyfikowanej przez wcześniejsze operację (jak chociażby PCA, LDA itp.). Bardzo ważną cechą jest możliwość podejrzenia tylko części zbioru danych w celu analizy przypadków spornych. Wszystkie te właściwości sprawiają, że aplikacja jest zarówno praktyczna jak i kompletna. Brak bardziej wyszukanych, aczkolwiek mniej efektywnych rozwiązań, zapewnia szybkość działania i małą niezawodność.

Nawet na podstawie dwóch zbiorów danych, które posłużyły do zaprezentowania możliwości stworzonego programu, widać jak przydatną metodą jest wizualizacja. Można nawet zaryzykować stwierdzenie, iż w większości przypadków jest ona niezbędna do kompletnej analizy, a na pewno zawsze jest bardzo pomocna. W bardzo rozległej dziedzinie jaką jest „*data mining”* potrzebna jest aplikacja, skupiająca zarówno metody numeryczne jak i wykorzystujące naturalne zdolności człowieka (percepcja). Program, jak już wcześniej zostało wspomniane, został stworzony jako rozbudowany moduł do aplikacji Intemi, która ma być systemem łączącym wiele metod do analizy danych. Dzięki temu możemy wykorzystać możliwości stworzonego programu po wcześniejszej „obróbce” danych.

Spis ilustracji

Bibliografia

Literatura:

1) T. Hill, P. Lewicki. *STATISTICS Methods and Applications.* Wydawnictwo StatSoft Inc., Tulsa, 2006.

2) S. Balakrishnama, A. Ganapathiraju, J. Picone. *Linear discriminant analysis for signal processing problems* . Southeastcon '99. Proceedings. IEEE, Lexington, 1999.

3) M. Lee, D. Vickers. *Psychological Approaches to Data Visualization.* Wydawnictwo DSTO Electronics and Surveillance Research Laboratory, Salisbury Lipiec 1998.

4) A. Naud. *Neural and Statistical Methods for the Visualization of Multidimensional Data.* Katedra Metod Komputerowych, Uniwersytet Mikołaja Kopernika, Toruń, 2001. http://www.phys.uni.torun.pl/publications/kmk/01phd-an.pdf.

5) W. Duch, Y. Hayashi. *Computational Intelligence: Methods and Applications.* Katedra Informatyki Stosowanej, Uniwersytet Mikołaja Kopernika w Toruniu, Katedra Informatyki, Uniwersytet Meiji. http://www.fizyka.umk.pl/publications/kmk/00koszyce.pdf

6) C. Ware. *Information Visualization, Second Edition: Perception for Design.* Wydawnictwo Morgan Kaufmann, San Francisco, Kwiecień 2004.

Strony internetowe:

7) *NIST/SEMATECH e-Handbook of Statistical Methods*, http://www.itl.nist.gov/div898/handbook/

8) http://pl.wikipedia.org/