Podstawowe metody analizy, klasyfikacji i grupowania danych – ćwiczenia w środowisku R

Marcin Iwanowski

14 grudnia 2015

Spis treści

1 Wprowadzenie

W ramach ćwiczenia wykorzystywane będzie szereg pakietów pakietu R. Pakiety te należy zainstalować (opcja "pakiety–zainstaluj pakiety", oraz wczytać komendą `library(nazwa_pakietu)`). Wymagane pakiety to:

- psych
- stats
- class
- scatterplot3d
- matrixStats
- e1071
- party

W celu ujednolicenia sposobu korzystania z wybranych metod eksploracji przygotowany został zestaw funkcji znajdujący się w pliku `eksplo.R`, który należy wczytać komendą `source("eksplo.R")`. Wywołanie tej komendy skutkuje wczytaniem odpowiednich funkcji oraz pakietów zewnętrznych.

Zestaw funkcji wraz z plikami danych wykorzystywanymi w ćwiczeniu (pliki z rozszerzeniem ".dat") znajdują się w spakowanym pliku. Plik ten należy rozpakować do nowo utworzonego katalogu, zaś po uruchomieniu pakietu R uczynić ten katalog katalogiem roboczym. Wówczas pliki znajdujące się w katalogu staną się widoczne z przestrzeni roboczej.

2 Eksploracyjna analiza danych

Celem eksploracyjnej analizy danych jest określenie podstawowych własności i zależności charakterystycznych dla analizowanego zbioru danych zapisanego w postaci macierzy danych. W pakiecie R przewidziano specjalną strukturę danych do przechowywania tego typu dwuwymiarowych danych zawierających informacje o obiektach (wiersze) i ich atrybutach (kolumny) – ramkę danych (data frame). W zdaniach znajdujących się w dalszej części instrukcji wykorzystywane będzie standardowy (dostępny bezpośrednio z linii komend R) zbiór danych `iris` oraz zbiory danych `md1`,...`md11` znajdujące się w dostarczonym spakowanym katalogu. Dane te będą przedmiotem badań z wykorzystaniem funkcji zapisanych w pliku `eksplo.R`. Funkcje te mogą także służyć do badania dowolnych innych macierzy danych spełniających trzy warunki: 1. dane muszą mieć postać ramki danych (typ "data frame"), 2. atrybuty opisujące powinny byc ilościowe (typ "numeric"), oraz 3. atrybut decyzyjny być kategoryczny (typ "factor").

Do eksploracyjnej analizy danych przeznaczone są funkcje: `eksploruj`, `pokaz`, `histbox`.

2.1 Wizualizacja danych

W celu interpretacji i wstępnej jakościowej oceny danych wykorzystuje się często prezentację graficzną w postaci różnego rodzaju wykresów, z których najczęściej spotykanymi są: wykres punktowy, histogram i wykres pudełkowy.

2.1.1 Wykres punktowy i macierz wykresów punktowych

Wykres punktowy, nazywany także wykresem rozrzutu, jest bardzo wygodnym sposobem prezentacji wartości dwóch atrybutów ilościowych. Każdy punkt tego wykresu odpowiada pojedynczemu obiektowi, którego współrzędne „x" i „y" reprezentują wartości jego dwóch atrybutów ilościowych. Na wykresie punktowym moga byc przedstawione ponadto kategorie poszczególnych obiektów. Wykorzystuje się w tym celu kształt i/lub kolor poszczególnych punktów.

Przykład 1 *Wykresy punktowe*

```
s <- read.table("samochody.dat")
head(s)
plot(s$predkosc,s$zpaliwa) # wykres bez rozróżnienia klas
dev.new() # nowe okno wykresu
plot(s$predkosc,s$zpaliwa, col=s$typ, pch = 16)
# wykres z rozróżnieniem klas $
```

Wywołanie w postaci `plot(s$predkosc,s$zpaliwa)` wyświetla wykres punktowy, którego oś „x" odpowiada pierwszemu atrybutowi (prędkośc maksymalna), zaś oś „y" – drugiemu (zużycie paliwa). Wywołanie drugie – `plot(s$m[,1],s$m[,2],col = s$d,pch=16)` pokazuje atrybuty odpowiednio drugi i trzeci, trzy dotatkowe argumenty oznaczają kolejno: wektor etykiet (`col = s$d`) i rodzaj wyświetlanego punktu (`'pch = 16'`). Na wykresie kolor punktu odpowiada wartości atrybutu decyzyjnego obiektu macierzy danych, zaś współrzędne tego punktu – wartościom dwóch atrybutów opisujących.

Wykres punktowy może być także trójwymiarowy, odpowiednie wywołanie pokazuje przykład 2.

Przykład 2 *Wykresy punktowe trójwymiarowe*

```
s <- read.table("samochody.dat") # jeśli wczytane wcześniej można pominąć
scatterplot3d(s$predkosc,s$zpaliwa, s$moc, pch=16) # wykres bez rozróżnienia klas
dev.new()
scatterplot3d(s$predkosc,s$zpaliwa, s$moc, color=as.numeric(s$typ), pch = 16)
# wykres z rozróżnieniem klas $
```

Dobrym narzędziem eksploracyjnej analizy danych o *większej* liczbie atrybutów pozwalającym na wizualną identyfikację prostych zależności między atrybutami jest **macierz wykresów punktowych**. Elementami tej macierz są wykresy punktowe wszystkich par atrybutów.

Przykład 3 *Macierze wykresów punktowych*

```
s <- read.table("samochody.dat") # jesli wczytane wczesniej mozna pominac
pairs(s[,1:3]) # wykres bez rozroznienia klas
dev.new()
pairs(s[,1:3], pch=21, bg=s[,4]) # wykres z rozroznieniem klas
```

W bibliotece `eksplo.R` dostępna jest funkcja `pokaz`, która wyświetla wykresy punktowe dla zadanej macierzy danych, dla wszystkich lub wybranych jej atrybutów.

Przykład 4 *Funkcja* `pokaz`

```
s <- read.table("samochody.dat") # jesli wczytane wczesniej mozna pominac
pokaz(s) # pokazuje macierz wykresow punktowych z podzialem na klasy
pokaz(s, dec=0) # pokazuje macierz wykresow punktowych bez podzialu na klasy
pokaz(s, opis=c(1,2)) # pokazuje wykres punktowy atrybutow 1 i 2
pokaz(s, opis=c(2,3), dec =0)
```

Uwaga: przykład 4 należy wykonać kolejno wywołując poszczególne linie powodujące wyświetlanie odpowiednich wykresów i obserwując wynik. Wywołanie całego przykładu metodą kopiuj-wklej spowoduje, ze widoczny będzie jedynie wykres wyświetlony jako ostatni. Uwaga ta dotyczy także wszystkich podobnych przykładów zamieszczonych w dalszej części skryptu.

Wywołanie funkcji `pokaz` bez wskazania, które atrybuty są opisujące (argument `opis`), zaś które – decyzyjne (argument `dec`), oznacza przyjęcie zasady, zgodnie z którą argument decyzyjny jest

ostatnim (kolumna macierzy danych o najwyższym indeksie), zaś wszystkie atrybuty o niższych indeksach są atrybutami opisującymi. Inne przyporządkowanie atrybutów wymaga odpowiedniego ustawienia argumentów `opis` i `dec`. Przypisanie `dec=0` oznacza, że wizualizacja danych następuje bez przypisywania każdej klasie unikalnego koloru punktu na wykresie. Takie przypisanie argumentu można wykorzystać jeśli atrybut decyzyjny nie jest zdefiniowany lub nie jest uwzględniany w analizach.

Przykład macierzy danych `iris` wykorzystuje standardowe dane o nazwie „Fisher's Iris data set", zawierające dane (długość i szerokość) kwiatów i łodyg trzech gatunków kwiatów irysa. Obiekty składają się w tym przypadku z czterech atrybutów ilościowych (wyników pomiarów kwiatów i łodyg) i jednego kategorycznego nominalnego (gatunek irysa) o wartościach odpowiadających trzem odmianom irysa: *virginica*, *setosa* oraz *versicolor*. Każdy obiekt odpowiada więc pojedynczemu kwiatowi, wartości atrybutów opisujących są wynikami jego pomiarów, zaś atrybut decyzyjny to gatunek, który reprezentuje. Cały zbiór danych składa się ze 150 obiektów (kwiatów irysa).

Przykład 5 *„Fisher's Iris data set"*

```
# wczytujemy zbiór danych 'iris'
i = iris
# wyświetlamy strukture zmiennej 'i'
str(i)
# wyświetlamy jeden z atrybutów opisujących
i$Petal.Width
# wyświetlenie wyswietlenie wartości atrybutu dla klasy 'setosa' $
i$Petal.Width[i$Species == "setosa"]
```

Zadanie 1 *Obejrzyj macierz danych* `fishersiris` *przy pomocy wszystkich omówionych wykresów. Które odmiany irysa są łatwiejsze do rozróżnienia na podstawie analizy wartości romiarów ich płatków i łodyg ? Które spośród czterech atrybutów opisujących pozwalają na takie rozróżnienie lepiej, a które gorzej ?*

2.1.2 Histogram i wykres pudełkowy

Do generowania histogramów w środowisku R służy funkcja `hist()`. Funkcja ta może być użyta bez argumentu wyjściowego i wówczas wyświetla okno pokazujące histogram. Wywołanie jej w następujący sposób (np. `h <- hist(s[,1],plot=FALSE)`) nie powoduje wyświetlenia histogramu lecz zwrócenie poprzez ten argument wektora wartości histogramu.

Przykład 6 *Histogram rozkładu normalnego*

```
# utworzenie wektora wartosci (rozkł. normalny)
y <- rnorm(10000)
# histogram z domyslną liczbą przedziałów
hist(y)
# histogram z zadaną liczbą przedziałów
hist(y,40)
```

Zadanie 2 *Dla macierzy danych* `iris` *wyświetl histogramy poszczególnych atrybutów.*

Zadanie 3 *Dla trzeciego atrybutu opisującego macierzy danych* `iris` *(przykład 5) wyznacz histogramy obiektów należących do poszczególnych klas tj. virginica,setosa i versicolor. Patrząc na te trzy histogramy oceń czym róznią się irysy należące do poszczególnych odmian ?*

Wykres pudełkowy jest wygodnym i czytelnym sposobem prezentacji rozkładu wartości atrybutu ilościowego w zbiorze danych. Za pomocą symbolu graficznego przedstawiane są podstawowe miary rozkładu (opisane w dalszej części skryptu):

- górna i dolna krawędź „pudełka" reprezentuje odpowiednio pierwszy i trzeci kwartyl
- linia w centralnym rejonie „pudełka" reprezentuje medianę (drugi kwartyl).
- pionowe odcinki odchodzące od linii pierwszego i trzeciego kwartyla reprezentują minimalną i maksymalną wartość atrybutu w próbce.

Na podstawie wykresu pudełkowego można również wnioskować o skrzywieniu danych (rozumianego jako znacząca asymetyria rozkładu i dominacja jego obszarów skrajnych). Przykładowo, jezeli linia mediany jest wyraźnie zbliżona do jednego z kwartyli, jest to oznaką skrzywienia danych w próbce.

Przykład 7 *Wykres pudełkowy*

```
x1 = 8*runif(100) + 10
x2 = 2*runif(100) + 5
boxplot(x1,x2)
```

Pytanie 1 *Czy możliwe jest, aby dwa różne zbiory danych miały identyczne wykresy pudełkowe? Jeżeli tak - podać przykład, jeżeli nie - uzasadnić*

Zadanie 4 *Narysuj wykresy pudełkowe atrybutów w poszczególnych kategoriach macierzy danych* `samochody`*. Czym różnią się poszczególne kategorie samochodów, oceń na podstawie wykresów ?*

Zadanie 5 *Wykorzytując wartości średnie, odchylenia standartowe, kwartyle znajdź naistotniejsze różnice pomiędzy trzema odmianami irysa w zbiorze* `iris`*. Skonfrontuj wyniki z obserwacjami poczynionymi w zadaniu 1.*

W bibliotece `eksplo.R` dostępna jest funkcja `histbox`, która wyświetla, dla danej macierzy danych histogramy i wykresy pudełkowe wszystkich (lub wybranych) atrybutów z podziałem na klasy.

Przykład 8 *Funkcja* `histbox`

```
m <- read.table("md1.dat")
histbox(m)
```

Funkcja `histbox` z pojedynczym (`histbox(m)`) wyświetla wykresy dla wszystkich atrybutów opisujących znajdujących się w macierzy `m`, z klasami określonymi przez atrybut decyzyjny. Zakłada się przy tym, że atrybut decyzyjny jest umieszczony na końcu wszystkich atrybutów (najwiekszy indeks) zaś pozostałe atrybuty (o niższych indeksach) są opisujące. Pełne wywołanie funkcji(`histbox(m,opis=...,dec=...,gl=...)`) zawiera trzy parametry, oznaczające kolejno: wektor indeksów atrybutów opisujących, indeks atrybutu decyzyjnego oraz parametr `gl`, który określa stopień wygładzenia histogramu (możliwe wartości między 0 a 1).

Zadanie 6 *Przy pomocy funkcji* `pokaz` *i* `histbox` *obejrzyj macierze danych* `md1.dat` *...* `md11.dat` *oraz zbiór* `iris`*. Jakie wnioski można wyciągnąć z obserwacji wykresów dla poszczególnych zbiorów ? Co można o każdym z tych zbiorów powiedzieć ?*

2.2 Miary danych

2.2.1 Miary tendencji centralnej

Miary tendencji centralnej określają pozycję „środka" zbioru danych. Do miary tych zalicza się wartości średnie, medianę i modę.

Wyróżnia się trzy podstawowe **rodzaje średnich**: arytmetyczną, geometryczną i harmoniczną.

Średnia arytmetyczna jest ilorazem sumy wartości atrybutu i liczebności badanego zbioru danych. Jest to najczęściej wykorzystywana miara tendencji centralnej, charakteryzująca przeciętną wartość atrybutu w zbiorze obiektów. Opisuje się ją wzorem:

$$s = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{1}$$

Średnia geometryczna jest pierwiastkiem k-tego stopnia z iloczynu n wartości:

$$g = \sqrt[n]{\prod_{i=1}^{n} x_i} \tag{2}$$

Średnią harmoniczną wyliczamy ze wzoru:

$$h = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}} \tag{3}$$

Wykorzystanie funkcji obliczających wartości średnie pokazuje przykład 9.

Przykład 9 *Średnie*

```
x = c(1,1,1,1,1,100)
barplot(x)
y = c(1,2,3,3,4,2,1)
dev.new()
barplot(y)
z = c(1,1,1,100,1,1,1)
dev.new()
barplot(z)
v = c(1,2,4,7,5,2,2,1)
dev.new()
barplot(v)
(s1 = c(geometric.mean(x),harmonic.mean(x),mean(x)))
(s2 = c(geometric.mean(y),harmonic.mean(y),mean(y)))
(s3 = c(geometric.mean(z),harmonic.mean(z),mean(z)))
(s4 = c(geometric.mean(v),harmonic.mean(v),mean(v)))
```

Pytanie 2 *Jak kształtują się średnie dla danych silnie zróżnicowanych (x, z), a jak dla słabo zróżnicowanych (y, v) ?*

Pytanie 3 *Czy kolejność elementów macierzy x, y oraz z wpływa na wartości średnich?*

Przykład 10 *Wartości średnie dla macierzy danych* `samochody`*.*

```
s <- read.table("samochody.dat")
mean(s[,1])  # wartość średnia atrybutu 1
mean(s[s$typ=="limuzyna",1]) # j.w. ale dla samochodów kategorii ,,limuzyna''
mean(s[s$typ=="sportowy",1]) # j.w. ale dla samochodów kategorii ,,sportowy''
```

Pytanie 4 *Jakie wnioski dotyczące poszczególnych kategorii samochodów mógłbyś wysnuć na podstawie obserwacji wartości średnich parametrów samochodów ?*

Zadanie 7 *Wyświetl wykresy punktowe komendą* `pokaz`*. Zwróć uwagę na znajdujące się na nich gwiazdki oznaczające średnie wartości atrybutów dla poszczególnych klas. Wykonaj to zadanie także dla innych dostępnych macierzy danych.*

Mediana (wartość środkowa) jest to wartość dzieląca zbiór wartości na dwie równe części w taki sposób, że 50% jednostek w zbiorowości ma wartości mniejsze-równe (lub mniejsze) medianie, a 50% większe (lub większe-równe). Jest to środkowa liczba w uporządkowanej niemalejąco próbce: $x_1 \leq x_2 \leq \mathbf{x_3} \leq x_4 \leq x_5$. Medianą jest wartość $M = x_3$. Jeżeli wyliczamy medianę z szeregu o parzystej liczbie elementów, np.: $x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6$ wówczas jest ona średnią arytmetyczną dwóch elementów środkowych: $M = \frac{x_3+x_4}{2}$.

Przykład 11 *Mediana*

```
a = c(11,9,5,10,3)
median(a)
b = c(11,5,10,9,3)
median(b)
c = c(100,5,9,11,1)
median(c)
```

Pytanie 5 *Wektory a, b i c mają identyczną medianę. Wymień trzy cechy, którymi się różnią a które nie mają wypływu na medianę. Które z tych cech miałyby wpływ na wartość średnią, a które nie ?*

Moda (wartość modalna, dominanta) jest to najczęściej występujaca wartość atrybutu w zbiorze danych. Można ją wyznaczyć zarówno dla atrybutów ilościowych jak i jakościowych. Moda nie może być wartością skrają w uporządkowanej próbie. Jeżeli istnieje więcej niż jedna moda, rozkład taki nazywamy **wielomodalnym** (rozkład z kilkoma „garbami").

Przykład 12 *Moda*

```
a=c(1,2,7,4,2,5,11,2,7)
modal(a)
```

Zadanie 8 *Zmień jeden i tylko jeden element macierzy a tak, aby moda przybrała inną wartość.*

2.2.2 Miary rozrzutu

Miary rozrzutu (rozproszenia, dyspersji) opisują rozrzut danych wokół ich środka określonego miarą tendencji centralnej. Najpopularniejszymi miarami tego typu są odchylenie standardowe, wariancja i kwartyle.

Najprostszą miarą rozrzutu jest miara **rozstępu**. Jest on definiowany jako różnica wartości maksymalnej i minimalnej wektora danych. Wartości minimalne, maksymalne oraz inne przydatne miary dla wszystkich atrybutów są wyznaczane przy pomocy komendy `summary`.

Przykład 13 *Rozstęp*

```
s <- read.table("samochody.dat")
summary(s)  # wyświetla kilka podstawowych miar, w tym wartości min i max
```

Zadanie 9 *Oceń rozstęp poszczególnych atrybutów macierzy danych* `iris`*.*

Podstawową miarą rozrzutu jest **odchylenie standardowe** opisujące stopień rozproszenia danych wokół wartości średniej:

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - s)^2} \tag{4}$$

przy czym x_i są kolejnymi wartościami wektora danych, zaś s jest ich średnią arytmetyczną.

Odchylenie standardowe jest pierwiastkiem **wariancji** w, zatem:

$$w = s^2 \tag{5}$$

$$\frac{1}{n}\sum_{i=1}^{n}(m_i - s)^2 \tag{6}$$

Miary te są tym większe, im większe jest rozproszenie wartości atrybutu wokół wartości średniej. Odchylenie standardowe jest bowiem średnią kwadratową różnicą wartości danego atrybutu i ich średniej, zaś wariancja — jej kwadratem.

Komendami pozwalającymi na wyznaczanie odchylenia standartowego i wartiancji są odpowiednio `sd` oraz `var`.

Przykład 14 *Wariancja*

```
x=c(1,5,9,7,15,16)
sd(x)
var(x)
y=c(1,5,9,7,7,7)
sd(y)
var(y)
z=c(1,2,4,4,2,100)
sd(z)
var(z)
```

Pytanie 6 *Jakie wnioski można wyciągnąć z porównania odchyleń standardowych dla macierzy x, y oraz z ?*

Miary dyspersji stanowią istotne uzupełnienie miar tendencji centralnych. Wartość średnia nie uwzględnia rozproszenia wartości atrybutu(ów). Dwa zupełnie różne wektory mogą mieć tę samą wartość średnią. Wówczas możliwe jest ich rozróżnienie np. przy pomocy odchylenia standardowego. Pokazuje to przykład 16.

Przykład 15 *Wartość średnia, odchylenie standardowe, wariancja*

```
m1 = c(5,6,4,5,6,4)
m2 = c(20,-28,0,10,32,-4)
mean(m1)
mean(m2)
```

```
 sd(m1)
 sd(m2)
```

Przykład 16 *Miary rozrzutu*

```
md <- read.table("md4.dat")
plot(md[,1],md[,2],col=md$klasa, pch=16)
summary(md[md$klasa=="klasa 1",]) # podsumowanie obiektów w klasie 1
sd(md[md$klasa=="klasa 1",1]) # odchyl.stand. pierwszego atrybutu w klasie 1
sd(md[md$klasa=="klasa 1",2]) # i drugiego
summary(md[md$klasa=="klasa 2",]) # podsumowanie obiektów w klasie 2 $
# ... itd
```

Pytanie 7 *Co możesz powiedzieć o każdej z trzech kategorii z przykładu 16 na podstawie wartości odchyleń standardowych ? Czy obserwacja wykresu punktowego potwierdza te wnioski ?*

Zadanie 10 *Wykonaj podobna analizę dla innych zbiorów danych* `md`.

Kwantyl jest to parametr statystyczny służący do opisu zbioru danych, niosący informację o ich rozproszeniu. Dla wektora m i rzędu r jest to taka liczba k, że $r \cdot 100\%$ elementów wektora m ma wartość nie większą niż k. Dla pewnych szczególnych wartości r kwantyle noszą nazwy **kwartyli**:

- dla $r = 0,25$ – pierwszy kwartyl
- dla $r = 0,5$ – drugi kwartyl – mediana (szczególny kwartyl centralny)
- dla $r = 0,75$ – trzeci kwartyl

Miarą rozproszenia danych jest **odstęp międzykwartylowy**, który definiuje się jako różnicę pomiędzy pierwszym i trzecim kwartylem.

Przykład 17 *Odstęp międzykwartylowy*

```
  m = floor(5*runif(30)) # wektor wartości losowych
  barplot(m)    # i jego wizualizacja
  m1 = sort(m)  # sortowanie
  barplot(m1)   # wizulalizacja wektora posortowanego
  summary(m)
IQR(m)  # odstep między kwartylowy
```

Pytanie 8 *Zlokalizuj poszczególne kwartyle na wykresie posortowanych wartości wektora m. Co możesz powiedzieć o ich położeniu ?*

2.2.3 Współmierność atrybutów – normalizacja i standaryzacja

W przypadku atrybutów ilościowych na szczególną uwagę zasługują zakresy zmienności poszczególnych atrybutów. Często bowiem różne atrybuty ilościowe tego samego obiektu odnoszą się do różnych pojęć. I tak, przykładowo, przy atrybutach obiektu będącego samochodem (np. macierz danych `samochody`), dwa atrybuty ilościowe mógłby opisywać prędkość maksymalną w km/h oraz zużycie paliwa w l/100km. W takim przypadku, zakresy zmienności obu atrybutów są różne – prędkość maksymalna może przyjmować wartości pomiędzy 100 a 200 km/h, zaś zużycie paliwa – pomiędzy 5 a 20 l/100km. Porównując dwa obiekty charakteryzujące się takimi atrybutami okazałoby się, że wpływ atrybutu o większej zmienności dopuszczalnych wartości (jak np. prędkość

maksymalna) jest nieproporcjonalnie większa od atrybutu o mniejszej zmienności i przyjmującego mniejsze wartości (jak np. zużycie paliwa). W celu ujednolicenia zakresów zmienności atrybutów stosuje się jedno z dwóch podejść: normalizację lub standaryzację. W przypadku atrybutów ilościowych na szczególną uwagę zasługują zakresy zmienności poszczególnych atrybutów. Często bowiem różne atrybuty ilościowe tego samego obiektu odnoszą się do różnych pojęć. I tak, przykładowo, przy atrybutach obiektu będącego samochodem (np. macierz danych `samochody`), dwa atrybuty ilościowe opisują prędkość maksymalną w km/h oraz zużycie paliwa w l/100km. W takim przypadku, zakresy zmienności obu atrybutów są różne – prędkość maksymalna może przyjmować wartości pomiędzy 100 a 200 km/h, zaś zużycie paliwa – pomiędzy 5 a 20 l/100km. Porównując dwa obiekty charakteryzujące się takimi atrybutami okazałoby się, że wpływ atrybutu o większej zmienności dopuszczalnych wartości (jak np. prędkość maksymalna) jest nieproporcjonalnie większa od atrybutu o mniejszej zmienności i przyjmującego mniejsze wartośći (jak np. zużycie paliwa). W celu ujednolicenia zakresów zmienności atrybutów stosuje się jedno z dwóch podejść: normalizację lub standaryzację.

Standaryzacja jest przekształceniem zmiennej x na standaryzowaną zmienną x'_i według wzoru:

$$x_i = \frac{x_i - s}{\sigma}, \tag{7}$$

przy czym s jest jej średnią arytmetyczną a σ - odchyleniem standardowym.

Średnia arytmetyczna ze zbioru standaryzowanego wynosi 0, zaś odchylenie standardowe σ wynosi 1.

Normalizację danych rozumiemy jako procedurę przekształcenia zmiennych w taki sposób, aby ich wartości znalazły się w pewnym z góry określonym przedziale ¡0,1¿ i całkowicie ten przedział pokrywały.

$$x''_i = \frac{x_i - x_{min}}{x_{max} - x_{min}} \tag{8}$$

Przykład 18 *Normalizacja min-max i standaryzacja*

```
x1 = 8*runif(100) + 10
x2 = 2*runif(100) + 5
x1a = (x1 - min(x1))/(max(x1) - min(x1))
x1b = (x1 - mean(x1))/sd(x1)
x2a = (x2 - min(x2))/(max(x2) - min(x2))
x2b = (x2 - mean(x2))/sd(x2)
```

Zadanie 11 *Na podstawie przykładu 18 wykonaj standaryzację macierzy danych „samochody".*

W bibliotece `eksplo.R` za normalizację i standaryzację odpowiada funkcja `normalizuj(m,typ,atryb)`. Poszczególne parametry oznaczają odpowiednio macierz danych, typ operacji (`typ='norm'` dla normalizacji, `typ='std'` dla standaryzacji), oraz wektor indeksów kolumn (atrybutów) macierzy danych, które są poddawane wybranej operacji.

Zadanie 12 *Wykonaj normalizację i standaryzację macierzy danych „iris" korzystając z funkcji* `normalizuj`*. Obejrzyj i zinterpretuj wyniki, policz podstawowe miary.*

2.3 Identyfikacja obserwacji oddalonych

Do poprawnej standaryzacji danych niezwykle istotna jest identyfikacja i korekcja zaburzeń wprowadzanych przez **obserwacje (punkty) oddalone**, czyli pojedyncze elementy zbioru danych, które dostały się do niego przypadkowo i nie odzwierciedlają faktycznego stanu rzeczy. Punkty takie zwykle pojawiają się na skutek błędu (na przykład przy przepisywaniu wyników ankiet,

awarii czujnika, błędu oprogramowania) i powinny być wyeliminowane na etapie wstępnej obróbki danych, ponieważ utrudniają analizę i silnie zaburzają wartości miar takich jak średnia arytmetyczna czy wariancja. Nawiązując do wcześniejszego przykładu samochodu atrybut „zużycie paliwa w l/100km" nie może przyjąć wartości 200 tak jak atrybut „prędkość maksymalna" nie może przyjąć wartości 8 (choć zamieniając obie te dane miejscami uzyska się już rozsądne wartości). Istotną wskazówką w identyfikacji elementów oddalonych jest fakt, że punkty takie są bardzo nieliczne w porównaniu do zbioru danych, leżą z dala od głównych skupisk punktów obserwacyjnych i nie mają zwykle żadnych bliskich sąsiadów. W rozkładach normalnych (jednomodalnych, symetrycznych) ponad 60% obserwacji odchyla się od średniej arytmetycznej o mniej niż jednokrotność odchylenia standardowego, ok. 95% - o mniej niż dwukrotność odchylenia standardowego.

Zidentyfikowane punkty oddalone można usunąć ze zbioru danych przed przystąpieniem do zadań eksploracji.

Pytanie 9 *Dane są dwa wektory:*
`A=[1,2,4,5,5,6,2,3,7,6,110]`,
`B=[2,4,5,5,54,48,51,110,109,113]`.
Które punkty można uznać za oddalone a które nie? Odpowiedź uzasadnić.

Istnieje wiele sposobów identyfikacji obserwacji oddalonych. Jedne z nich polega na wykorzystaniu odchylenia standardowego. Za punkty oddalone uznawane są te, dla których bezwzględna wartość różnicy wartości punktu i wartości średniej jest większa niż pewna krotność odchylenia standardowego.

Przykład 19 *Wykrywanie punktów oddalonych*

```
md <- read.table("md5.dat")
plot(md[,1],md[,2],col=md[,3],pch=16) # wszystkie klasy
md1=md[md$klasa=="klasa 1",1:2]
plot(md1[,1],md1[,2],pch=16) # jedna z klas
md1oind = abs(md1[,1] - mean(md1[,1])) < 2*sd(md1[,1])
# indeksy punktów oddalonych ze względu na atrybut 1
plot(md1[,1],md1[,2],pch=16,col=md1oind+1)  # wyświetlamy punkty oddalone $
```

Zadanie 13 *Pozmieniaj wielokrotność odchylenia standardowego i sprawdź jak wpłynie to na efekt wykrywania punktów oddalonych.*

2.3.1 Korelacja i kowariancja

Wymienione dotychczas miary służyły do opisu pojedynczych atrybutów. Jednak kluczową rolę w analizie danych odgrywają wzajemne zależności pomiędzy atrybutami. Atrybuty mogą być ze sobą powiązane w takim sensie, że wzrost wartości jednego powoduje proporcjonalny wzrost (lub – odwrotnie – spadek) wartości innego. O takich atrybutach mówimy, że są ze sobą skorelowane. W takim przypadku uwzględnianie obu atrybutów nie wnosi wiele nowej informacji w porównaniu z sytuacją gdy brany jest pod uwagę tylko jeden atrybut. Najprostszym narzędziem do wizualnej identyfikacji zależności między atrybutami jest wykres punktowy (dla dwóch atrybutów) lub macierz wykresów punktowych (dla większej liczby atrybutów).

Zadanie 14 *Wczytaj macierz danych* `md10` *(atrybuty opisujące znajdują się w macierzy* `m`*, zaś wartości atrybutów decyzyjnych, klasy – w macierzy* `c`*), wyświetl wykres punktowy z wyróżnieniem obiektów należących do poszczególnych klas, a następnie oceń czy istnieje zależność między atrybutami opisującymi w każdej klasie oraz – jeśli istnieje – to jaki ma charakter.*

Zadanie 15 *Oceń, przy pomocy macierzy wykresów punktowych, zależności między atrybutami opisującymi macierzy danych* `samochody`*.*

Przydatnymi narzędziami statystycznym pozwalającymi na ocenę liniowych zależności pary atrybutów są kowariancja i korelacja. Kowariancja dwóch atrybutów „x" i „y" jest definiowana następująco:

$$\sigma_{x,y} = \frac{1}{n}\sum_{i=1}^{n}(x_i - s_x)(y_i - s_y), \tag{9}$$

przy czym s_x jest wartością średnią atrybutu x, zaś s_y jest wartością średnią atrybutu y.

Miara kowariancji jest wykorzystywana do określania poziomu liniowej zależności między dwoma atrybutami. Poziom ten może być mierzony przy pomocy współczynnika korelacji, nazywanym też współczynnikiem Pearsona, który jest ilorazem kowariancji i iloczynu odchyleń standardowych obu atrybutów:

$$r_{x,y} = \frac{\sigma_{x,y}}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{n}(x_i - s_x)(y_i - s_y)}{\sqrt{\sum_{i=1}^{n}(x_i - s_x)^2 \sum_{i=1}^{n}(y_i - s_y)^2}}, \tag{10}$$

Współczynniki korelacji przyjmują wartości z przedziału domkniętego między -1 a 1. Jego wartość dodatnia oznacza dodatnią zależność liniową dwóch atrybutów, ujemna zaś – zależność ujemną. Wartość 0 oznacza brak korelacji. Im większa jest wartość bezwzględna współczynnika, tym większy jest poziom wzajemnej korelacji obu atrybutów.

W przypadku większej liczby atrybutów, ich wzajemne powiązania można wykryć stosując macierz kowariancji, która zawiera wzajemne kowariancje wszystkich par atrybutów:

$$C = \begin{bmatrix} \sigma_1 & \sigma_{12} & \sigma_{13} & \cdots & \sigma_{1k} \\ \sigma_{21} & \sigma_2 & \sigma_{23} & \cdots & \sigma_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sigma_{k1} & \sigma_{k2} & \sigma_{23} & \cdots & \sigma_k \end{bmatrix} \tag{11}$$

przy czym k jest liczbą atrybutów, σ_i jest wariancją i-tego atrybutu, zaś σ_{ij} jest kowariancją atrybutów i-tego oraz j-tego. Macierz kowariancji jest macierzą symetryczną o nieujemnym wyznaczniku. Dla atrybutów nieskorelowanych macierz kowariancji jest równa macierzy jednostkowej tj. takiej, której elementy położone na głównej diagonali są równe 1, zaś pozostałe – 0.

Analogicznie definiuje się macierz korelacji:

$$R = \begin{bmatrix} 1 & r_{12} & r_{13} & \cdots & r_{1k} \\ r_{21} & 1 & r_{23} & \cdots & r_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{k1} & r_{k2} & r_{23} & \cdots & 1 \end{bmatrix} \tag{12}$$

Macierz korelacji zawiera współczynniki korelacji poszczególnych par atrybutów. Podobnie do macierzy kowariancji jest to macierz symetryczna, jej wyznacznik zawiera się w przedziale domkniętym $< 0, 1 >$. Elementy na głównej przekątnej są równe 1, ponieważ współczynnik korelacji każdego atrybutu z nim samym wynosi zawsze 1.

Do wyznaczania wartości kowariancji służy w R funkcja `cov`, zaś korelacji – `cor`.

Przykład 20 *Wyznaczanie korelacji i kowariancji*

```
md <- read.table("md10.dat")
plot(md[,1],md[,2],col=md[,3],pch=16)
cov(md[md$klasa=="klasa 1",1:2])
cov(md[md$klasa=="klasa 2",1:2])
cov(md[md$klasa=="klasa 3",1:2])
cor(md[md$klasa=="klasa 1",1:2])
cor(md[md$klasa=="klasa 2",1:2])
cor(md[md$klasa=="klasa 3",1:2])
```

Uwaga – w przykładzie 20, pomimo występowania jedynie dwóch atrybutów, wyznaczane są macierze kowariancji i korelacji. Dla dwóch atrybutów w istocie ważna jest jedynie dwukrotnie powtarzająca się wartość poza główną przekątną.

Pytanie 10 *Zinterpretuj wyniki z przykładu 20 odnosząc je do wykresu punktowego z zadania 14.*

Pytanie 11 *Jaki rodzaj zależności między atrybutami opisują miary kowariancji i korelacji ?*

Pytanie 12 *Która z miar: kowariancja czy korelacja jest bardziej uniwersalnym narzędziem oceny zależności atrybutów ? Dlaczego ?*

Zadanie 16 *Oceń stopień zależności pomiędzy atrybutami opisującymi macierzy danych* `samochody` *(nie uwzględniając podziału na dwie klasy) wyznaczając odpowiednie macierze. Czy interpretacja wartości w tych macierzach jest zgodna z obserwacją macierzy wykresów punktowych z zadania 15 ?*

W bibliotece `eksplo.R` została zaimplementowana funkcja `eksploruj(m,opis,dec)` o parametrach: `m` – macierz danych, `opis` wektor indeksów atrybutów opisujących, `dec` – indeks atrybutu decyzyjnego. Funkcja ta wyświetla podstawowe statystyki opisowe oraz macierze korelacji dla całej macierzy, oraz następnie dla każdej klasy z osobna.

Zadanie 17 *Przy pomocy funkcji* `eksploruj` *zbadaj macierze danych* `md1.dat` *...* `md11.dat` *oraz zbiór* `iris`*. Jakie wnioski można wyciągnąć z obserwacji wykresów dla poszczególnych zbiorów ? Co można o każdym z tych zbiorów powiedzieć na podstawie uzyskanych miar ? Spróbuj określić związek między odpowiednimi parametrami, a wykresami uzyskanymi w zadaniu 6.*

3 Klasyfikacja

Zadanie *klasyfikacji* wymaga macierzy danych zawierającej dane o specyficznej strukturze. Atrybuty obiektów w niej zawartych dzielą się na dwie grupy: atrybuty opisujące (zwykle wiele) i decyzyjne (przeważnie jeden). Zakłada się przy tym, że wartość atrybutów decyzyjnych zależy od wartości atrybutów opisujących. Typowym przykładem tego typu danych jest macierz zawierająca objawy różnych chorób. Obiektami w takiej macierzy są przypadki chorobowe, atrybuty opisujące reprezentują objawy, zaś atrybut decyzyjny – postawioną diagnozę (chorobę). Macierz tego typu może zawierać historyczne dane dotyczące diagnozowania różnych chorób przez lekarzy. Na jej podstawie można dokonywać automatycznego rozpoznania nowego przypadku chorobowe na podstawie objawów. Proces taki wymaga określenia – w tym przypadku nieznanej – wartości atrybutu decyzyjnego (choroby) na podstawie znanych wartości atrybutów opisujących (objawy chorobowe).

Proces *klasyfikacji* można zdefiniować jako proces określania nieznanej wartości atrybutu decyzyjnego na podstawie wartości atrybutów opisujących. *Klasyfikator* jest metodą, algorytmem realizującym proces klasyfikacji. Poprawna klasyfikacja konkretnych danych wymaga zwykle uprzedniego *nauczenia* klasyfikatora, które polega na wyznaczeniu pewnych jego parametrów (zależnych od rodzaju klasyfikatora). Tylko właściwie przygotowany klasyfikator jest bowiem zdolny do przeprowadzenia poprawnej klasyfikacji. Do nauczenia klasyfikator wykorzystuje się *zbiór uczący*, który zawiera obiekty opisane atrybutami obojga rodzajów – opisujących i decyzyjnych.

Zwykle początkowy zbiór danych (macierz danych) jest dzielony na dwa podzbiory: właściwy *zbiór uczący* wykorzystywany do właściwego uczenia klasyfikatora oraz *zbiór testowy* wykorzystywany do jego testowania. Testowanie polega na dokonywaniu klasyfikacji kolejnych obiektów z tego zbioru i porównywaniu jej wyniku z faktyczną wartością atrybutu decyzyjnego – znaną dla obiektów ze zbioru testowego. Dzięki temu możliwa jest ocena jakości procesu klasyfikacji.

3.1 Metodyka badania klasyfikatorów

3.1.1 Podział wejściowej macierzy danych, prezentacja danych

Celem tej części ćwiczenia jest praktyczne zaznajomienie się z wynikami działania 5 klasyfikatorów: najbliższego sąsiada, k-najbliższych sąsiadów, najbliższego prototypu, naiwnego klasfikatora Bayesa oraz drzewa decyzyjnego.

Po wczytaniu pliku danych pierwszą operacją powinien być jego podział na zbiór uczący i testowy. Komenda `ut <- podziel(m,p,dec)` dzieli w sposób losowy macierze wartości atrybutów opisujących i decyzyjnego na dwie części będące zbiorem uczącym i testowym. Danymi wejściowymi są macierz danych `m`, stosunek podziału `p` oraz indeks atrybutu decyzyjnego – numer kolumny macierzy danych, w której atrybut ten się znajduje. Do zbioru uczącego trafia $100\% \cdot p$ obiektów, zaś do testowego – $100\% \cdot (p-1)$. Ustalony stosunek podziału jest zachowany w każdej klasie (tj. klasy są dzielone w zadanym stosunku niezależnie). Wyniki są zwracane przez wektor `ut` o długości równej liczbie obiektów w macierzy `m`, indeks każdego elementu wektora odpowiada indeksowi obiektu w macierzy danych. Wartość 1 określa przynależność obiektu o danym indeksie do zbioru uczącego, zaś wartość 0 wskazuje na przynależność obiektu do zbioru testowego. Pominięcie atrybutu `p` oznacza, ze przyjmowany jest typowy stosunek `p=0.7`, czyli 70% obiektów trafia do zbioru uczącego, zaś pozostałe 30% zasila zbiór testowy. Pominięcie indeksu atrybutu decyzyjnego wskazuje, że za taki atrybut uznawany jest ostatni spośród znajdujących się w macierzy danych. Jeśli `dec = 0` to podział jest wykonywany bez uwzględnienia klas tj. wskazany procent obiektów do każdego ze zbiorów jest wybierany z całości zbioru, a nie z każdej klasy niezależnie. Taki sposób realizacji podziału może skutkować zmianą proporcji liczebności obiektów w klasach po podziale w stosunku do sytuacji w zbiorze oryginalny. W skrajnym przypadku może się okazać, że pewna klasa (lub klasy) zostaną w całości przypisane do jednego ze zbiorów – uczącego lub testowego, pozostawiając drugi spośród nich bez obiektów danej klasy. Dlatego zalecany jest podział z zachowaniem zadanego stosunku podziału w każdej klasie (argument `dec` większy od zera lub pominięty).

Funkcja `pokaz(m,ut)` wyświetla wykres punktowy dla macierzy danych `m` z podziałem na część uczącą i testową zgodnie z wektorem `ut`. Punkty wypełnione oznaczają elementy zbioru uczącego, punkty puste w środku - zbioru testowego, zaś gwiazdki wskazują na położenie średnich wektorów atrybutów każdej z klas.

Przykład 21 *Funkcja* `pokaz` *z podziałem na zbiór uczący i testowy*

```
i <- iris # jesli dane zostaly wczytane wczesniej mozna pominac
ui <- podziel(i)
pokaz(i,ui) # macierz wykresow punktowych z podzialem na zbior uczący i testowy
pokaz(i,ui, dec=0)
pokaz(i,ui, opis=c(1,2))
pokaz(i,ui, opis=c(3,4), dec = 0)
```

3.1.2 Testowanie klasyfikatora

Do testowania klasyfikatorów służą dwie funkcje `weryfikuj` oraz `granice`.

Komenda `w = weryfikuj(typ, m, ut, opis, dec, k, pokaz)` weryfikuje działanie klasyfikatora `typ` z wykorzystaniem macierzy danych `m` podzielonej na część uczącą i testową zgodnie z wektorem `ut`, dla atrybutów opisujących o indeksach zapisanych w `opis` i atrybucie decyzyjnym w `dec` oraz klasyfikatora `typ` (`k` i `pokaz` są parametrem dodatkowym wykorzystywanymi przez niektóre klasyfikatory). Efektem działania funkcji jest wyświetlenie dwóch macierzy kontyngencji odpowiadających wynikom testowania klasyfikatorów na zbiorach uczącym i testowym. Macierz kontyngencji (tabela krzyżowa) zawiera informację o liczbie obiektów w sklasyfikowanych do każdej z klas dla poszczególnych wartości atrybutu decyzyjnego. Przykładowo, macierz kontyngencji:

	1	2	3
1	39	3	0
2	12	78	1
3	1	1	56

należy czytać w ten sposób, że 39 obiektów należących (wg. wartości atrybutu decyzyjnego) do klasy 1 zostało do tej klasy sklasyfikowanych, 3 obiekty należące do klasy 1 zostały sklasyfikowane do klasy 2, i żaden do trzeciej; 12 obiektów klasy 2 zostało przypisanych do klasy 1, 78 obiektów trafiło do prawidłowej w tym przypadku klasy 2, oraz 1 do trzeciej i analogicznie dla ostatniego wiersza. Podsumowując, na łączną liczbę 191 (suma wszystkich elementów) obiektów, 173 (czyli 39+78+56) zostało prawidłowo sklasyfikowanych co daje 90,5% skuteczność.

Komenda `granice(typ, m, ut, atrx, atry, dec, k)` wyświetla wykres punktowy pokazujący granice decyzyjne dla klasyfikatora `typ`. Znaczenie poszczególnych parametrów jest analogiczne do funkcji `weryfikuj`. Jedyną różnicą jest ograniczenie liczby dopuszczalnych atrybutów opisujących do dwóch, których indeksy (numery kolumn) w macierzy danych są określone przez parametry `atrx` i `atry`. Granice decyzyjne oddzielają obszary w przestrzeni atrybutów, które odpowiadają poszczególnym klasom. Sposób podziału przestrzeni atrybutów zależy przy tym od rodzaju klasyfikatora. Ponieważ granice decyzyjne są pokazywane na wykresie punktowym dwuwymiarowym może być wywołana jedynie dla macierzy danych o dwóch atrybutach opisujących, to w tym przypadku klasyfikacja następuje jedynie z wykorzystaniem dwóch atrybutów opisujących. W przypadku większej liczby takich atrybutów należy więc wybrać dwa spośród nich i dla niech przeprowadzić operację wyznaczania i wizualizacji granic decyzyjnych.

W obu funkcjach możliwe są następujące wartości parametru `typ`

- `knn` — klasyfikator k-najbliższych sąsiadów - parametr `k` oznacza liczbę sąsiadów, pominięcie tego argumentu oznacza przyjęcie k=1 (najprostszy klasyfikator minimlnoodległościowy)
- `knn` — klasyfikator najbliższego prototypu
- `bayes` — naiwny klasyfikator Bayesa
- `drzewo` — drzewo decyzyjne, parametr $0 < k < 100$ determinuje stopień szczegółowości drzewa, im niższa jego wartość tym drzewo jest bardziej rozbudowane. W przypadku drzewa brany jest także pod uwagę parametr `pokaz`. Przypisanie mu wartości 1 umożliwia wraz z wyświetlaniem granic decyzyjnych wyświetlenie także drzewa w formie graficznej w osobnym oknie. Wywołując funkcję `weryfikacja` drzewo jest zawsze wyświetlane w formie tekstowej (zestaw reguł) na ekranie konsoli.

Przykład 22 pokazuje przykładowe badanie klasyfikator k-najbliższych sąsiadów dla różnych wartości k.

Przykład 22 *Badanie klasyfikatora k-NN*

```
md <- read.table("md2.dat")
ut <- podziel(md)
pokaz(md)
pokaz(md,ut)
granice('knn',md,ut,1,2,3)
weryfikuj('knn',md,ut,c(1,2),3)
granice('knn',md,ut,1,2,3,k=3)
weryfikuj('knn',md,ut,c(1,2),3,k=3)
granice('knn',md,ut,1,2,3,k=5)
weryfikuj('knn',md,ut,c(1,2),3,k=5)
```

Pytanie 13 *Jak można ocenić wynik klasyfikacji na podstawie otrzymanych wykresów i macierzy kontyngencji ? Na podstawie jakich czynników możemy stwierdzić która wersja klasyfikatora jest lepsza a która gorsza ?*

Zadanie 18 *Sprawdź w podobny sposób pozostałe klasyfikatory. W przypadku drzewa decyzyjnego sprawdź i oceń efekty dla różnych poziomów szczegółowości drzewa zmieniając k w granicach od 1 do 99.*

3.2 Klasyfikatory

Klasyfikatory najbliższych sąsiadów należą do najprostszych metod klasyfikacji. Ich działanie polega na znajdowaniu obiektu *najbliższego* aktualnie rozpoznawanemu spośród obiektów znajdujących się w zbiorze uczącym. Klasa (wartość atrybutu decyzyjnego) charakterystyczna najbliższego obiektu jest wówczas przypisywana obiektowi rozpoznawanemu. W swoich najprstszych wersjach, klasyfikatory minimalnoodległościowe nie wymagają uczenia.

W przeciwieństwie do klasyfikatorów najbliższych sąsiadów, klasyfikatory wykorzystujące funkcję dyskryminacji wymagają fazy uczenia. W jej trakcie wyznaczane są parametry tej funkcji. W zależności od metody parametry te są wyznaczane w róznych sposób. Tok postępowania jest jednak w większości takich metod podobny. Funkcje dyskryminacji są wyznaczane dla wszystkich istniejących w danym zadaniu klasyfikacji klas. Argumentem tych funkcji jest wektor atrybutów rozpoznawanego obiektu. W procesie klasyfikacji wyznaczane sa wartości tych funkcji dla wszystkich klas, a następnie wyszukiwana jest wartość najmniejsza. Ostatecznie rozpoznawanemu obiektowi jest przypisywana klasa charakterystyczna dla funkcji dyskryminacji o najmniejszej wartości.

Ostatnią grupę przedstawianych klasfyikatorów tworzą drzewa decyzyjne. W procesie uczenia wyznaczane jest struktura grafowa – drzewiasta, której węzły odpowiadają warunkom nałożonym na atrybuty, zaś liście odpowiadają wynikom klasyfikacji. Proces klasyfikowania obiektu sprowadza się do wykonania serii testów atrybutów zgodnie z zapisami w drzewie decyzyjnym.

3.2.1 Najprostszy klasyfikator minimalnoodległościowy

Działanie *najprostszego klasyfikatora minimalnoodlegościowego* polega na tym, że rozpoznając obiekt, wyznacza się kolejno odległości między nim, a wszystkimi obiektami ze zbioru uczącego. Ściślej rzecz ujmując, odległości te są wyznaczane pomiędzy wektorami atrybutów opisujących obiektów. Wyznaczając te odległości znajduje się jednocześnie odległość najmniejszą. Po przeanalizowaniu wszystkich obiektów ze zbioru uczącego, za wynik klasyfikacji obiektu nieznanego przyjmuje się wartość atrybutu decyzyjnego najbliższego obiektu ze zbioru uczącego. Ten typ klasyfikatora nosi także nazwę *klasyfikatora najbliższego sąsiada.*

Zadanie 19 *Wykonaj klasyfikację na zbiorze* `md1`*, spójrz na granice decyzyjne i wyniki weryfikacji. Oceń skuteczność. Kilkakrotnie powtórz podział na zbiór uczący i testowy, także w innych proporcjach. Jak zmieniają się wyniki ?*

Zadanie 20 *Wykonaj podobne czynności na zbiorach* `md2`*,*`md7`*. Oceń wyniki. Czy są lepsze czy gorsze niż dla* `md1`*. Zastanów się skąd się biorą różnice.*

Zadanie 21 *Poeksperymentuj w analogiczny sposób z pozostałymi macierzami danych. Oceń przydatność klasyfikatora dla każdego zbioru danych.*

Zaletą klasyfikatora minimalnoodległościowego jest jego prostota. Jego cechą charakterystyczną jest brak fazy uczenia – w procesie klasyfikacji jest przeglądany cały zbiór uczący w poszukiwaniu najbliższego obiektu.

3.2.2 Klasyfikator k-najbliższych sąsiadów

Jedną z istotniejszych wad klasyfikatora najbliższego sąsiada jest jego wrażliwość na obecność przypadkowych obiektów należących do danej klasy znajdujących się w obszarze zajmowanym przez obiekty z innych klas. Aby uniezależnić się od takich sytuacji stosuje się klasyfikator k-najbliższych sąsiadów. Jego działanie polega na wyszukiwaniu – zamiast dokładnie jednego – zadanej liczby k sąsiadów charakteryzujących się najmniejszą odległością do obiektu klasyfikowanego. W najprostszym wariancie tego klasyfikatora, klasa do której rozpoznawany obiekt przynależy jest wyznaczana jako klasa dominująca wśród k najbliższych sąsiadów ze zbioru uczącego. Dzięki takiemu podejściu istnienie pojedynczego błędnie klasyfikowanego obiektu ze zbioru uczącego w otoczeniu obiektu rozpoznawanego nie wpłynie na wynik klasyfikacji ponieważ wadliwie klasyfikowany obiekt będzie w mniejszości w stosunku do pozostałych k najbliższych sąsiadów.

Zadanie 22 *Powtórz zadania 19 oraz 20 dla klasyfikatora k-NN. Wskaż na obrazie granic decyzyjnych miejsca, w których klasyfikator k-najbliższych sąsiadów daje lepsze wyniki klasyfikacji niż prosty klasyfikator minimalnoodległościowy. Dlaczego w tym przypadku wyniki są lepsze ?*

Pytanie 14 *Czy wszystkie obszary na wykresie granic decyzyjnych mają jednoznaczne przypisanie do konkretnej klasy ? Czy widzisz obszary o niejednoznacznej przynależności ? Jeśli tak to zastanów się jaka jest przyczyna ich powstania.*

Zadanie 23 *Sprawdź jak przebiega proces klasyfikacji tych samych obiektów dla rożnych wartości k. Sprawdź jak zmieniają się wyniki dla różnych podziałów na zbiór uczący i testowy.*

Pytanie 15 *Jak zmieniają się wyniki klasyfikacji wraz ze wzrostem k ?*

Zadanie 24 *Poeksperymentuj w analogiczny sposób z pozostałymi macierzami danych. Oceń przydatność klasyfikatora dla każdego zbioru danych.*

3.2.3 Klasyfikator najbliższych prototypów

Wadą wszystkich opisanych wyżej klasyfikatorów najbliższych sąsiadów jest konieczność wykorzystania podczas procesu właściwej klasyfikacji z całego zbioru uczącego. Niesie to ze sobą istotne konsekwencje. Dla dużej liczby obiektów zbioru uczącego proces klasyfikacji staje się czasochłonny, często zachodzi ponadto konieczność przechowywania całego tego zbioru w pamięci.

Rozwiązaniem tego problemu jest redukcja zbioru uczącego do zbioru składającego się z obiektów reprezentatywnych dla każdej z rozpatrywanych klas – *prototypów klas*. Wówczas, zamiast rozpatrywania całego zbioru uczącego, rozpatrywany jest jedynie zbiór prototypów. Typowym rozwiązaniem jest wybór po jednym prototypie na klasę, choć stosowane jest także rozwiązanie polegające na wyborze większej liczby prototypów każdej klasy.

Pytanie 16 *Na czym polega faza uczenia klasyfikatora najbliższych prototypów ?*

Pytanie 17 *Jak zmierzyć stopień rozproszenia wartości atrybutu wokół wartości średniej ?*

Zadanie 25 *Powtórz zadania 19 oraz 20 dla klasyfikatora najbliższego prototypu. Jak zakłócenia wpływają na wyniki klasyfikacji ?*

Zadanie 26 *Podziel zbiór wejściowy* `md2` *na uczący i testowy, wg. typowych proporcji. Dla tak wykonanego podziału wykonaj klasyfikację najprostszym klasyfikatorem minimalnodległościowym, klasyfikatorem k-najbliższych sąsiadów oraz najbliższych prototypów. Wykonaj weryfikację wyni-*

ków klasyfikacji na zbiorze uczącym i testowym dla każdego z trzech klasyfikatorów. Oceń wyniki.

Pytanie 18 *Zastanów się jak zinterpretować błędne wskazania klasyfikatora dla obiektów ze zbioru uczącego ? Dlaczego może się zdarzyć, że nie wszystkie obiekty zbioru uczącego są poprawnie klasyfikowane? Na wykresie punktowym wskaż takie obiekty. Czy – w tym konkretnym przypadku – niepoprawna klasyfikacja jest efektem pozytywnym czy negatywnym ? O jakiej własności klasyfikatora ona świadczy ?*

Zadanie 27 *Wykonaj klasyfikację klasyfikatorem najbliższego prototypu dla macierzy danych* `md8`, `md9`, `md10`, `md11`. *Oceń wyniki i porównaj je z wynikami wcześniej poznanych klasyfikatorów.*

Zadanie 28 *Poeksperymentuj w analogiczny sposób z pozostałymi macierzami danych. Oceń przydatność klasyfikatora dla każdego zbioru danych.*

3.2.4 Naiwny klasyfikator Bayesa

Podstawą klasyfikacji Bayesowskiej jest twierdzenie Bayesa. W przypadku klasyfikacji, zdarzenia losowe, które są brane pod uwagę przy wyznaczaniu prawdopodobieństw dotyczą dwóch faktów związanych z rozpoznawanymi obiektami: posiadania przez obiekt konkretnego zbioru wartości atrybutów opisujących zapisanego zwykle w formie wektora wartości atrybutów oraz przynależności tego obiektu do poszczególnych klas. i-ta funkcja dyskryminacji dla obiektu o wektorze atrybutów opisujących jest w tym przypadku tożsama prawdopodobieństwu warunkowemu przynależności obiektu do i-tej klasy pod warunkiem posiadania przez obiekt wektora atrybutów opisujących. Wygodnym założeniem jest brak zależności między poszczególnymi atrybutami opisującymi. Dzięki niemu można przyjąć, że zdarzenia losowe polegające na posiadaniu przez obiekt konkretnych wartości poszczególnych atrybutów są od siebie niezależne. Klasyfikatory spełniające to założenie noszą nazwę naiwnych klasyfikatorów Bayesowskich.

W przypadku atrybutów ilościowych niezbędne prawdopodobieństwa szacuje się z wykorzystaniem typowych rozkładów zmiennych losowych

Zadanie 29 *Wykonaj klasyfikację klasyfikatorem Bayesa wszystkich zbiorów, dla których wyniki klasyfikacji najbliższego prototypu były niezadowalające. Czy zastosowanie klasyfikatora Bayesa je polepsza ? Dlaczego ?*

Zadanie 30 *Znajdź zbiory danych, w przypadku których wyniki klasyfikacji Bayesowskiej są gorsze niż k-NN. Zastanów się dlaczego tak się dzieje.*

Pytanie 19 *Dla jakich dystrybucji obiektów w przestrzeni atrybutów (położenia zbiorów punktów na wykresie punktowym) klasyfikator Bayesa daje dobre wyniki, a dla jakich gorsze ? Dlaczego ?*

Zadanie 31 *Poeksperymentuj z pozostałymi macierzami danych. Oceń przydatność klasyfikatora dla każdego zbioru danych.*

3.2.5 Drzewa decyzyjne jako klasyfikatory

Drzewa decyzyjne są struktura grafową przedstawiającą zależności między atrybutami obiektów. Formalnie rzecz ujmując, drzewo jest acyklicznym grafem spójnym. Drzewa są wyznaczane dla macierzy danych zawierających atrybuty opisujące i atrybut decyzyjny. Drzewo decyzyjne opisuje w formie grafu zależność wartości atrybutu decyzyjnego od wartości atrybutów opisujących.

Dzięki hierarchicznej reprezentacji tych zależności drzewo nie tylko jest klasyfikatorem, ale także umożliwia analizę istotności poszczególnych atrybutów dla konkretnego procesu klasyfikacji.

Węzły drzewa decyzyjnego zawierają testy atrybutów. Gałęzie odchodzące od węzłów odpowiadają poszczególnym wynikom testów. Z każdego węzła odchodzi więc tyle gałęzi, ile jest wyników testu w nim umieszczonego. Gałęzie łączą węzły z innymi węzłami albo węzły z liśćmi. Liście są zakończeniami drzewa i reprezentują wartości atrybutu decyzyjnego. Droga łącząca pierwszy węzeł drzewa (korzeń) z poszczególnymi liśćmi zawiera sekwencję testów, które należy wykonać aby otrzymać odpowiadający danemu liściowi wynik klasyfikacji.

Działanie metody indukcji drzewa decyzyjnego polega na kolejnym doborze testów przeznaczonych do umieszczenia w kolejnych jego węzłach. Budowa drzewa decyzyjnego składa się z trzech podstawowych zadań:

- Wyboru typu testów atrybutów i kolejności ich wykonywania.
- Określenia reguły decydującej o tym, czy dany węzeł ma być węzłem końcowym, czy ma podlegać kolejnym podziałom.
- Sposobu, w jaki każdemu węzłowi końcowemu przyporządkowujemy etykietę klasy.

Częstym problemem przy klasyfikacji za pomocą drzew jest *przetrenowanie*. Nie zawsze stuprocentowo skuteczna klasyfikacja zbioru uczącego przekłada się na dobre wyniki klasyfikacji zbioru testowego, ponieważ drzewo, w którym jest zbyt wiele zbyt szczegółowych testów traci zdolność generalizacji. Problem przetrenowania może byc rozwiązany na dwa sposoby, poprzez:

- wstrzymanie budowy drzewa, zanim osiągnie maksymalne rozmiary (ograniczanie w trakcie rozrostu), lub
- przycinanie drzewa po jego wyznaczeniu (drzewa maksymalnego).

Zadanie 32 *Utwórz drzewa decyzyjne dla wszystkich macierzy danych* `md` *i zbioru* `iris`*. Dla jakich danych konstrukcja drzew jest prostsza, a dla jakich – bardziej skomplikowana ? Dlaczego ?*

Zadanie 33 *Poeksperymentuj z różnymi wartościami współczynnik k, w przypadku drzew wpływającego na złożoność (dopasowanie) drzewa*

Pytanie 20 *Jak wielkość drzewa wpływa na skuteczność klasyfikacji ?*

Zadanie 34 *Porównaj na wybranym zbiorze danych działanie klasyfikatora drzewiastego z wcześniej omówionymi ? Jakie widzisz różnice ?*

Zadanie 35 *Poeksperymentuj z pozostałymi macierzami danych. Oceń przydatność klasyfikatora dla każdego zbioru danych.*

Zadanie 36 *Poeksperymentuj z różnymi klasyfikatorami na zbiorze* `iris` *o czterech atrybutach decyzyjnych, wybierając tylko dwa z nich. Zwróć uwagę na to jak wybór dwóch z czterech atrybutów wpływa na wynik klasyfikacji. Wskaż najlepszą i najgorszą parę atrybutów z punktu widzenia poprawności klasyfikacji. Czy jesteś w stanie wskazać na macierzy wykresów punktowych dla tego zbioru danych, cechy rozkładu punktów, które potwierdzają ten wybór ? Porównaj wynik klasyfikacji dla najlepszej pary atrybutów z klasyfikacją z wykorzystaniem wszystkich czterech atrybutów.*

4 Grupowanie danych

Jednym z klasycznych zadań eksploracji danych jest problem organizowania obserwowanych danych w sensowne struktury i łatwe do interpretacji grupy. Obecnie zagadnienie to jest szczególnie istotne, gdyż coraz częściej mamy do czynienia z ogromnymi ilościami danych i niemożliwe jest analizowanie ich bez użycia komputerów i skutecznej metodologii eksploracyjnej. Szczególnie duże znaczenie ma na tym polu *grupowanie danych*.

Grupowanie danych (nazywane też analizą skupień lub klasteryzacją) jest dość szerokim pojęciem obejmującym szereg różnych algorytmów kojarzenia danych. Jej istotą jest podział zbioru obiektów na niepuste, rozłączne i możliwie jednorodne grupy: skupienia. Obiekty należące do jednego skupienia powinny być możliwie najbardziej „podobne" do siebie i możliwie najbardziej różnić się od obiektów innych grup.

Grupowanie danych znalazło szerokie zastosowanie w biznesie, szczególnie w branży telekomunikacyjnej, ubezpieczeniowej i bankowej. Przykładowe zastosowania obejmują między innymi segmentację klientów na podstawie ich zachowań konsumenckich, marketing kierowany, wykrywanie oszustw, szacowanie ryzyka czy profilowanie.

W przeciwieństwie do klasyfikacji wzorcowej (znanej z poprzedniego ćwiczenia), polegającej na przyporządkowywaniu przypadków do jednej ze z góry określonych klas, tu klasy nie są znane ani w żaden sposób scharakteryzowane przed przystąpieniem do analizy. Nie istnieje żaden zbiór uczący, na bazie którego dobierany jest klasyfikator, nie mamy więc żadnej wiedzy *a priori* o analizowanym zbiorze i próbujemy niejako „odgadnąć" prawidłowości i związki między danymi jedynie na podstawie nich samych. Różne metody grupowania mogą dawać różne, jednak równie użyteczne i sensowne wyniki, które mają za zadanie dać raczej ogólny obraz wewnętrznej struktury danych niż stanowić podstawę do precyzyjnej klasyfikacji.

Aby móc porównywać obserwacje między sobą i określać, na ile są one do siebie podobne, konieczne jest wprowadzenie pojęcia *miary podobieństwa*.

Podobieństwo jest zwykle wyznaczane jako odwrotności odległości. Im bowiem obiekty są bliższe, zgodnie z przyjętym sposobem wyznaczania odległości, tym są bardziej do siebie podobne.

W bibliotece `eksplo.R` grupowanie danych jest realizowane przy pomocy funkcji `grupuj`. Składnia funkcji jest następująca: `grupuj(typ, m, k, opis)`, kolejne argumenty oznaczają metodę grupowania, macierz danych, liczbę skupień oraz wektor zawierający indeksy atrybutów opisujących.

4.1 Metoda k-średnich

Metoda k-średnich jest jednym z najpopularniejszych algorytmów grupowania danych. Jego zaletą jest stosunkowo nieduży koszt obliczeniowy i prosta konstrukcja. Jest to metoda iteracyjna, wymagająca wcześniejszego podania liczby k skupień, do których mają zostać zakwalifikowane obiekty. Wybór liczby skupień ma duży wpływ na jakość uzyskanej segmentacji. Podanie zbyt dużej liczby skupień może spowodować, że wyznaczone grupy będą co prawda wewnętrznie jednorodne, jednak bardzo utrudniona będzie ich interpretacja i praktyczne wykorzystanie wyników. Z drugiej strony, im mniejsza liczba skupień, tym są one mniej jednorodne wewnętrznie, co także jest efektem niepożądanym.

Standardowo algorytm k-średnich polega na przenoszeniu obiektów ze skupienia do skupienia w celu zminimalizowania zmienności wewnątrz skupień, maksymalizując jednocześnie zmienności między skupieniami. Zasada działania algorytmu k-średnich jest następująca -– kolejno wykonujemy kroki:

1. Ustalamy pożądaną liczbę skupień, liczba tych skupień zapisywana jest w zmiennej k.

2. Ustalamy wstępnie środki skupień. Najczęściej na wstępne środki skupień wybieranych jest losowo k obiektów ze zbioru danych lub k dowolnych punktów w przestrzeni atrybutów).

3. Obliczamy odległości obiektów od środków skupień, wykorzystując wybraną miarę odległości.

4. Przypisujemy obiekty do skupień – dla danego obiektu porównujemy odległości do wszystkich środków skupień (obliczonych w punkcie 3) i przypisujemy go do tego skupienia, którego środek położony jest najbliżej w przestrzeni atrybutów.

5. Ustalamy nowe środki skupień – przyjmuje się, że są to punkty, których współrzędnymi są średnie arytmetyczne współrzędnych obiektów, na danym etapie działania algorytmu należących do danego skupienia.

6. Jeżeli w punkcie 5 przesunęliśmy środki skupień, to powtarzamy kroki 3, 4, 5, w przeciwnym razie algorytm zatrzymuje się, a za ostateczny wynik przyjmujemy bieżący podział.

Jak już wspomniano, przed przystąpieniem do analizy należy określić, na ile skupień chcemy dzielić interesujące nas obiekty. Wybór liczby skupień może być dokonany na wiele sposobów. Wybór początkowych centrów skupień może być losowy, lub też dokonany przez użytkownika na podstawie wstępnej wizualizacji danych i ewentualna późniejsza zmiana tej liczby, aby otrzymać lepsze wyniki. Alternatywnym podejściem jest przeprowadzenie wstępnej analizy za pomocą metody hierarchicznej, oszacowanie za jej pomocą liczby skupień, a następnie dla tak wybranej liczby skupień wykonanie analizy metodą k-średnich.

Algorytm k-średnich jest zbieżny niezależnie od wybranych początkowych centrów skupisk, nie gwarantuje jednak zbieżności do globalnego rozwiązania optymalnego. W zależności od początkowego wyboru centrów, algorytm może zbiegać do różnych lokalnych rozwiązań optymalnych. W związku z tym, aby uzyskać zadowalający rezultat, zaleca się kilkukrotne wykonanie grupowania, dla różnych początkowych centrów skupisk. Inna metoda poszukiwań optymalnego podziału polega na dokonaniu wstępnego grupowania w oparciu o niewielką próbkę pobraną ze głównego zbioru (ok. 5-10% całkowitej liczby obiektów), a następnie wykorzystaniu centrów skupisk znalezionych w tej wstępnej procedurze jako początkowych centrów przy grupowaniu całego zbioru danych.

Istotne znaczenie dla jakości uzyskanych wyników ma wybór miary podobieństwa (odległości) pomiędzy między obiektami. W przypadku zmiennych ilościowych najczęściej stosuje się odległość euklidesową. Odległość euklidesowa, tak jak inne pokrewne jej miary, ma jednak pewną wadę: może silnie podlegać wpływowi jednej ze zmiennych, mianowicie tej, której zakres wartości jest największy. Jeśli wartości tej zmiennej są znacznie większe od wartości innych zmiennych, wtedy o różnicy bądź podobieństwie między obserwacjami będzie, w dużej mierze, decydowała tylko ta jedna zmienna (wynika to wprost z formuły, za pomocą której wyliczamy odległość euklidesową). Może to mieć miejsce na przykład, gdy zmienne wyrażone są w różnych jednostkach lub reprezentują różny rząd wielkości. Aby zapobiec takiej sytuacji, stosuje się normalizację min-max, lub standaryzację.

Grupowanie metoda k-średnich może zostać zrealizowane poprzez wywołanie funkcji `grupuj` z atrybutem `typ = "ksrednich"`, jak w przykładzie 21. Funkcja zwraca wektor zawierający przyporządkowania poszczególnych obiektów do skupień, który może być porównany z informacjami zawartymi w wektorze wartości atrybutu decyzyjnego (jeśli jest on dostępny).

Przykład 23 *Grupowanie metodą k-średnich*

```
md <- read.table("md2.dat")
pokaz(md)
dev.new()
pokaz(md,dec=0) # zapominamy o atrybucie decyzyjnym
g <-grupuj("ksrednich",md,3,c(1,2),3) # grupowanie (trzy grupy)
# wynik grupowania (kolor - wyn.grupowania)
table(md[,3],g) # macierz kontyngencji
```

Pytanie 21 *O czym mówi nam macierz kontyngencji ? Dlaczego w tym przypadku może nie być diagonalna ?*

Zadanie 37 *Poeksperymentuj z innymi liczbami skupień.*

Zadanie 38 *Wykonaj grupowanie dla macierzy danych* `md2`*. Czy w tym przypadku wyniki są równie dobre jak w przykładzie 21 ?*

Zadanie 39 *Poeksperymentuj z innymi macierzami danych* `md`*.*

4.2 Grupowanie hierarchiczne

Hierarchiczne metody grupowania danych dzielą się na dwie grupy: aglomeracyjne i rozdzielające. W ramach ćwiczenia zostaną pokazane metody aglomeracyjne. Działanie tych metod polega na kolejnym łączeniu najbardziej do siebie podobnych obiektów w grupy. Początkowo każdy obiekt stanowi odrębną grupę. W pierwszym kroku, obiekty najbardziej do siebie podobne (czyli najmniej odległe) są łączone w pojedynczą grupę. W ten sposób całkowita liczba grup jest redukowana o 1. Następnie znajdowane się kolejne dwie najbliższe grupy, które łączą się w jedną. Proces ten jest powtarzany aż do uzyskania docelowej liczby grup (skupień).

Kluczową rolę w procesie grupowania odgrywa odległość między obiektami poddawanymi grupowaniu. W przypadku grupowania hierarchicznego wyznaczenie odległości pomiędzy wszystkimi obiektami stanowi pierwszy krok metody. Odległości wyznaczone w tym kroku są umieszczane w wektorze odległości, zawierającym odległosci pomiędzy skupieniami. Początkowo wektor odległości zawiera odległości między wszystkimi obiektami w zbiorze.

Wektor odległości jest wykorzystywania w grupowaniu hierarchicznym do znajdowania najbliższych sobie obiektów czy grup obiektów. W przypadku wyszukiwania najbliższych grup obiektów wykorzystuje się różne podejścia, z których najpopularniejsze to:

- metoda *pojedynczego połączenia* ang. „single linkage" w której odległość między dwoma grupami obiektów jest równa najmniejszej spośród odległości pomiędzy obiektami należącymi do pierwszej i do drugiej grupy,
- metoda *całkowitego połączenia* ang. „complete linkage" w której odległość między dwoma grupami obiektów jest równa największej spośród odległości pomiędzy obiektami należącymi do pierwszej i do drugiej grupy,
- metoda *średniego połączenia* ang. „average linkage" w której odległość między dwoma grupami obiektów jest równa średniej odległości pomiędzy obiektami należącymi do pierwszej i do drugiej grupy.

Łącząc kolejno najbliższe grupy obiektów uzyskuje się w końcu pojedynczą grupę do której należą wszystkie obiekty macierzy danych. Kolejność łączenia w kolejnych iteracjach zależy od przyjętej metody połączenia. Kolejność łączenia grup przedstawia się graficznie w postaci struktury drzewiastej – *dendrogramu*.

Analiza dendrogamu pozwala na uzyskanie ostatecznego wyniku grupowania. Wyróżnia się dwa zasadnicze sposoby takiej analizy. W pierwszym ustala się zadaną liczbę skupień. Wówczas „odcinając" dendrogram na poziomie na którym występuje liczba skupień równa zadanej. W drugim podejściu nie ustala się z góry żądanej liczby skupień. Ustala się natomiast próg wartości odległości międzygrupowej. Często tę wartość progową podaje się jako procent maksymalnej wartości odległości występującej w dendrogramie (która jest odległością pomiędzy dwoma ostatnimi skupieniami – gałęziami odchodzącymi od korzenia dendrogramu).

Zadanie 40 *Przeanalizuj kolejność łączenia grup w każdym z typów połączeń, patrząc na odpowiedni dendrogram i wykres punktowy.*

Grupowanie metodami hierarchicznymi może zostać zrealizowane poprzez wywołanie funkcji `grupuj` z atrybutem `typ` o możliwych wartościach: `hie_pp` – dla metody pojedynczego połącze-

nia, `hie_cp` – dla metody całkowitego połączenia oraz `hie_sp` (alternatywnie: `hie`) – dla metody średniego połączenia. Przykładzie 24 pokazuje grupowanie hierarchiczne.

Przykład 24 *Grupowanie hierarchiczne*

```
md <- read.table("md1.dat")
pokaz(md)
dev.new()
pokaz(md,dec=0) # zapominamy o atrybucie decyzyjnym
g <-grupuj("hie",md,3,c(1,2)) # grupowanie (trzy grupy)
# wynik grupowania (kolor - wyn.grupowania)
table(md[,3],g) # macierz kontyngencji
```

Zadanie 41 *Zobacz jak zmienia się wynik dla różnych wartości parametrów* `hclust` *(wartości* `hie_pp`, `shie_cp`, `hie_sp`*).*

Zadanie 42 *Poeksperymentuj z innymi liczbami skupień.*

Zadanie 43 *Poeksperymentuj z innymi macierzami danych* `md`*.*

Zadanie 44 *Wykonaj grupowanie wg. przykładów 21 oraz 24 dla zbioru* `iris`*. Wykorzystaj zarówno wszystkie cztery jak i mniejszą liczbę atrybutów. Każdorazowo porównaj wyniki z wartościami atrybutu decyzyjnego. Dla jakiego zestawu atrybutów wyniki grupowania sa najlepsze ? O czym to świadczy ?*

Zadanie 45 *Funkcja* `m1 <- doklej(m,a)` *tworzy nowa macierz danych* `m1` *dołączając do macierzy danych* `m`*, jako ostatnią kolumnę, wektor* `a` *zawierający informacje o skupieniach do których poszczególne obiekty zostały przypisane np. w wyniku grupowania. Wykonanie tej operacji umożliwia wykorzystanie wyników grupowania jako atrybutu decyzyjnego w klasyfikacji. Dla wybranych przez siebie zbiorów danych wykonaj grupowanie i dobierz najlepszy klasyfikator przy założeniu, że wartość atrybutu decyzyjnego jest wynikiem grupowania.*