AGNIESZKA WNUK [1]
MARCIN KOZAK [1]
MAŁGORZATA TARTANUS [2]
[1] Katedra Doświadczalnictwa i Bioinformatyki, Szkoła Główna Gospodarstwa Wiejskiego, Warszawa
[2] Instytut Ogrodnictwa, ul. Konstytucji 3 Maja 1/3, 96-100 Skierniewice

Wprowadzenie do interaktywnej wizualizacji danych w środowisku R∗

Introduction to interactive data visualization in R environment

Wizualizacja danych jest bardzo skutecznym narzędziem wspomagającym proces analizy danych rolniczych na każdym jej etapie, od zapoznawania się z danymi, poprzez interpretację zjawisk, aż po wnioskowanie. Często pozwala na skuteczne zaobserwowanie struktury danych czy obserwacji odstających lub nietypowych, które w innym wypadku trudno byłoby dostrzec. Można jednak znacznie rozszerzyć jej możliwości poprzez dynamiczny wpływ użytkownika na postać wykresu za pomocą interaktywnej wizualizacji. W tym artykule przedstawiliśmy podstawowe sposoby wykorzystania tego rodzaju wizualizacji, w celu zwiększenia możliwości interpretacyjnych analizowanych danych. W ten sposób chcemy zwrócić uwagę czytelników na bogate możliwości, jakie daje użytkownikowi interaktywna wizualizacja. W pracy wykorzystaliśmy możliwości oferowane przez środowisko R; każdemu przykładowi towarzyszy kod w R, który może zostać użyty przez czytelnika do zapoznania się z przykładem, ale także może zostać wykorzystany w praktyce analizy danych.

Słowa kluczowe: analiza danych, metody graficzne, wizualizacja

Data visualization is a very useful tool supporting analyses of agricultural data at each stage, including data exploration, interpretation, and drawing conclusions. It frequently helps one to observe patterns in data, find outliers or untypical values, which otherwise would be difficult to detect. However, the usefulness of visualization can be greatly improved by the dynamic user's influence on the graph by interactive visualization. This paper introduces basic possibilities of applying such type of visualization to improve interpretational possibilities in data analysis. In this way we want to direct the readers' attention to rich opportunities that interactive visualization offers. The R environment is used in the paper; each example includes also the R code, which can be used by the reader to run the example, but can also be used in practical data analysis.

Key words: data analysis, graphical methods, visualization

∗ Praca była prezentowana w ramach I Warsztatów Biometrycznych, które odbyły się w IHAR-PIB w Radzikowie w dniach 14-15 września 2010 r.

WSTĘP

Wizualizacja dostarcza narzędzi niezwykle efektywnych przy analizie danych doświadczalnych (Cleveland, 1993 i 1994; Kozak, 2010). Co więcej, wizualizacja powinna być nieodłącznym etapem każdej analizy statystycznej, czy to przy zapoznawaniu się z danymi, ich interpretacją, sprawdzaniu założeń metod statystycznych czy też prezentowaniu wyników. Metody graficzne mają zastosowanie również w dydaktyce statystyki (Kozak i in., 2010).

Interaktywna wizualizacja jest jeszcze bardziej efektywna, i z pewnością o wiele bardziej efektowna. Interaktywność ta polega na dynamicznym wpływie użytkownika na postać wykresu poprzez użycie odpowiednich technik. Najprostszym spośród nich jest zwykłe wciśnięcie przycisku myszy, co spowoduje zaznaczenie wybranego punktu i wyświetlenie informacji o nim; nieco bardziej zaawansowana technika, zwana pędzlowaniem (ang. *brushing*), polega na interaktywnym zaznaczaniu wybranego obszaru wykresu, co skutkuje np. podświetleniem punktów znajdujących się w tym obszarze na innych wykresach, co pozwala efektownie interpretować dane wielowymiarowe.

Wbrew pozorom interaktywna wizualizacja danych nie jest wcale młodym działem wizualizacji. John Tukey wraz z zespołem z Computation Research Group of the Stanford Linear Accelerator Center pracował nad pierwszym programem do interaktywnej wizualizacji już w początkowych latach 70., co zaowocowało pierwszym takim programem — PRIM-9 — w 1972 r. (Friedman i Stuetzle, 2002). Od tego czasu interaktywna wizualizacja staje się coraz bardziej popularna wśród specjalistów od wizualizacji, zwłaszcza teraz, w dobie powszechności Internetu.

Tym bardziej zadziwiające jest, że w praktyce analizy danych interaktywna wizualizacja jest rzadko stosowana. Niestety nauki rolnicze nie są wyjątkiem. Celem niniejszej pracy jest zapoznanie czytelnika z najprostszymi technikami interaktywnej wizualizacji danych ilościowych w środowisku R (R Development Core Team 2010). Środowisko R jest darmowym oprogramowaniem do analizy i wizualizacji danych. Oprogramowanie instalujące można ściągnąć ze strony http://www.r-project.org/. Dodatkowo stworzono tysiące dodatkowych pakietów, oferujących funkcje służące do analizy specyficznych zagadnień. Doskonałym polskojęzycznym wprowadzeniem do środowiska R jest książka Biecka (2008).

Naszym zdaniem nawet najprostsze techniki interaktywnej wizualizacji mogą być pomocne w analizie danych. Przede wszystkim jednak celem tej pracy jest zwrócenie uwagi czytelnika na interaktywną wizualizację i zachęcenie go do zapoznania się z jej bogatymi możliwościami. Środowisko R dostarcza różnorodnych możliwości interaktywnej wizualizacji, co nie oznacza, że nie warto sięgnąć po inne oprogramowanie. Nie można jednak zapominać, że R jest równocześnie bardzo efektywnym narzędziem do statystycznej analizy danych, co skłania nas do zarekomendowania go użytkownikom, którzy szukają narzędzia zarówno statystycznego, jak i graficznego.

PRZYKŁAD 1

Dodanie etykiet do interaktywnie zaznaczonych punktów na wykresie

Jest to najprostszy sposób interakcji użytkownika i wykresu. Zaprezentujmy go dla zbioru danych `soil` z pakietu `agricolae` (de Mendiburu, 2010), przedstawiających właściwości gleby dla próbek zebranych w 13 lokalizacjach. Aby zainstalować ten pakiet, wystarczy do konsoli R wpisać komendę:

```
> install.package("agricolae")
```

Wczytajmy dane i zróbmy wykres zależności magnezu (Mg) od wapnia (Ca):

```
> data(soil, package = "agricolae")
> plot(soil$Ca, soil$Mg, las = 1); axis(3, lab = F); axis(4, lab= F)
```

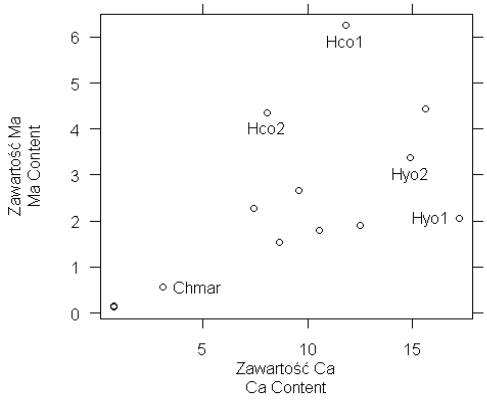
Argument `las = 1` powoduje, że etykiety znaczników osi y są zorientowane poziomo, co ułatwia ich odczyt; z kolei dwukrotne wywołanie funkcji `axis` dodaje znaczniki osi u góry i z prawej strony wykresu. Możemy sprawić, by wykres stał się interaktywny poprzez wywołanie funkcji `identify()`, której podstawowym celem jest identyfikacja punktów na wykresie:

```
> identify(soil$Ca, soil$Mg)
```

Od tej chwili kliknięcie w bliskim (tę odległość można modyfikować poprzez argument `tolerance`) otoczeniu wybranego punktu sprawi, że obok niego pojawi się etykieta. Domyślną etykietą są nazwy wierszy; w przypadku zbioru `soil` nazwy wierszy są równoważne ich numerom, dlatego warto zastanowić się nad lepszym wyborem etykiet. Jako że dany punkt na wykresie reprezentuje zawartość Ca i Mg w określonej lokalizacji, właśnie lokalizację warto wykorzystać jako etykietę. Możemy to uzyskać dzięki argumentowi `labels` w następujący sposób:

```
> identify(soil$Ca, soil$Mg, labels = soil$place)
```

albo krócej: `> with(soil, identify(Ca, Mg, labels = place))`



Rys. 1. Przykład zastosowania funkcji identify() przez dodanie etykiet dla zaznaczonych punktów. Przykład 1

Fig. 1. An example of application of the identify() function by adding the labels to selected points. Exemple 1

Rysunek 1 przedstawia wykres po zastosowaniu powyższego kodu i kliknięciu w otoczeniu pięciu punktów. Warto zauważyć, że to, gdzie znajduje się kursor w otoczeniu punktu, ma znaczenie na umiejscowienie etykiety – np. jeżeli klikniemy po prawej stronie punktu odpowiadającemu największej zawartości Ca, to etykieta zostanie dodana właśnie po prawej stronie tego punktu, a tym samym zostanie ucięta; dlatego należy kliknąć z lewej strony tego punktu, by uzyskać efekt przedstawiony na Rysunku 1.

Z trybu interaktywnego można wyjść na klika sposobów: wciskając klawisz Esc, wybierając z menu "Stop" opcję "Stop locator", czy też wciskając prawy przycisk myszy na obszarze wykresu i zaznaczając opcję „Stop”.

PRZYKŁAD 2

Wyświetlenie informacji o interaktywnie zaznaczonym punkcie

Tym razem chcielibyśmy uzyskać więcej informacji o interaktywnie zaznaczonym punkcie niż tylko etykietę. Zamiast dodawania etykiety, zaznaczenie punktu będzie powodowało wyświetlenie osobnego okienka, w którym pojawi się interesująca nas informacja. Skorzystajmy z tego samego wykresu co poprzednio:

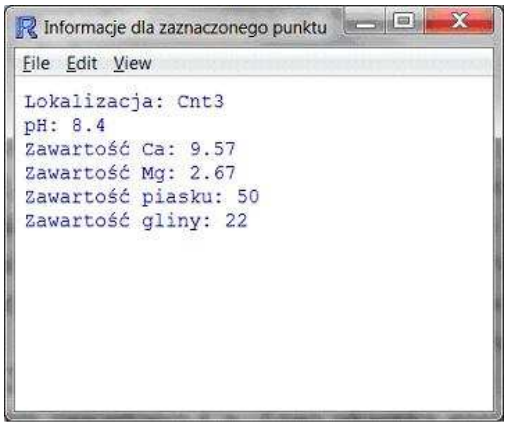
```
> plot(soil$Ca, soil$Mg, las = 1); axis(3, lab = F); axis(4, lab= F)
```

Załóżmy, że chcielibyśmy uzyskać informację o lokalizacji próbki, pH gleby oraz zawartości w niej Ca, Mg, piasku i gliny. Możemy to zrobić następująco:

```
> setwd("d:/")
> n.max <- nrow(soil)
> kliknięć <- 0
> while (kliknięć < n.max) {
  plot(soil$Ca, soil$Mg, las = 1); axis(3, lab = F); axis(4, lab= F)
    a <- identify(soil$Ca, soil$Mg, plot = F, n = 1)
    inf <- soil[a,]
    with(inf, write.table(paste("Lokalizacja:", place, "\npH: ", pH,
          "\nZawartość Ca: ", Ca, "\nZawartość Mg: ", Mg,
          "\nZawartość piasku: ", sand, "\nZawartość gliny: ", clay,
          sep = ""), file = "plik.txt", quote = F, row.names = F,
          col.names = F))
file.show("plik.txt", delete.file = TRUE, title = "Informacje dla
          zaznaczonego punktu", encoding = "CP1250")
          kliknięć <- kliknięć + 1
}
```

Parametr `n.max` ustala, ile maksymalnie punktów można zaznaczyć: wybraliśmy tyle, ile jest wierszy w zbiorze, ale oczywiście wartość tego parametru można zmienić. Tego typu interaktywność można w R uzyskać na wiele różnych sposobów; my wybraliśmy właśnie taki, ponieważ nie wymaga on instalowania dodatkowych pakietów: tworzymy po prostu na dysku D (należy to zmienić, jeżeli dysk ma inny symbol) tymczasowy plik tekstowy, który jest następnie wyświetlany i od razu usuwany z dysku (odpowiada za to

argument `delete.file = T)`. Rysunek 2 przedstawia pomniejszone okienko, jakie wyświetli się po naciśnięciu punktu o największej zawartości Ca.



Rys. 2. Przykładowe okienko wyświetlające informacje o interaktywnie zaznaczonym punkcie (lokalizacja: zawartość) Przykład 2
Fig. 2. An example of a window representing information about interactively selected point (localization: content) Exemple 2

PRZYKŁAD 3

Badanie wpływu obserwacji na linię regresji

Przy pomocy interaktywnej wizualizacji możemy zbadać, jak poszczególne obserwacje wpływają na linię regresji. Rozpatrzymy zarówno standardową analizę liniowej regresji prostej, jak i nieparametryczną regresję lokalnie ważoną (Cleveland 1994). Wywołanie niniejszej funkcji pozwala na usunięcie istniejących punktów ze zbioru (tymczasowego, na podstawie którego wykonywana jest analiza i wykres — faktyczny zbiór danych nie ulega modyfikacji), co sprawia, że zmienia się linia regresji.

```
wpływ.odstających <- function(y, x, typ = "regresja", ...) {
      dane <- data.frame(x = x, y = y)
      dd <- dane
      i <- 1
      while(i < (nrow(dd) - 3)) {
      if (typ == "regresja") {
      plot(dd$x, dd$y, xlim = range(dane$x), ylim = range(dane$y), xlab =
      "X", ylab = "Y", las = 1)
      axis(3, lab = F); axis(4, lab= F)
            abline(lm(y ~ x, dd)$coefficients)
      } else if (typ == "loess") {
      scatter.smooth(dd$x, dd$y, xlim = range(dane$x), ylim = range(dane$y),
      xlab = "X", ylab = "Y", las = 1, ...)
```

```
        axis(3, lab= F); axis(4, lab = F) } else
        stop('Błędna nazwa typu dopasowania funkcji.\nDopuszczalne typy to
        "regresja" i "loess"')
        a <- identify(dd, plot = F, n = 1)
        dd <- dd[-a,]
        i <- i + 1
        }
}
```

Oto krótka charakterystyka argumentów funkcji: `y` to wektor wartości zmiennej zależnej, `x` to wektor wartości zmiennej niezależnej, zaś `typ` to typ regresji: `"regresja"`, która odpowiada standardowej liniowej regresji prostej, a `"loess"` nieparametrycznej regresji lokalnie ważonej (Cleveland, 1994). Jako że *loess* może być szacowana na podstawie różnych ustawień algorytmu, odpowiednie argumenty można modyfikować w wywołaniu funkcji `wpływ.odstających()` (odpowiada za to argument `"..."`). Aby dowiedzieć się, jakie to argumenty, należy zapoznać się z pomocą dotyczącą funkcji `scatter.smooth`:

```
> ?scatter.smooth
```

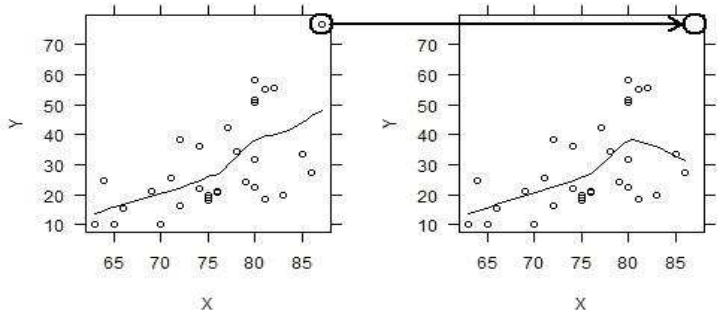
Bez braku ingerencji w te ustawienia, użyte zostaną domyślne ustawienia (z którymi również można się zapoznać na stronie pomocy dla funkcji `scatter.smooth`).

Funkcję należy wywołać następująco. Dla regresji:

`> wpływ.odstających(y = trees$Volume, x = trees$Height)` dla *loess*:

`> wpływ.odstających(y = trees$Volume, x = trees$Height)` dla *loess*, ale ze zmianą algorytmu estymacji:

```
> wpływ.odstających(y = trees$Volume, x = trees$Height, typ =
"loess", family = "gaussian")
```



Rys. 3. Przykład badania wpływu obserwacji na linię regresji (Przykład 3). Zaznaczenie punktu w prawym górnym roku wykresu z lewej spowodowało jego usunięcie, co wyraźnie zmieniło linię nieparametrycznej regresji lokalnie ważonej, szacowanej przy pomocy metody najmniejszych kwadratów

Fig. 3. An example of studying how an observation affects a regression line. Selecting a point in top right corner of the left graph caused its removal, which noticeably changed the nonparametric locally weighted regression line, estimated with the least square method

Rysunek 3 przedstawia wpływ odstającego punktu na postać linii regresji nieparametrycznej, uzyskanej przy pomocy estymacji metodą najmniejszych kwadratów. Wykres po lewej przedstawia wyjściową sytuację, a ten po prawej – po usunięciu punktu w prawym górnym rogu. Łatwo zauważyć, że estymacja przy pomocy najmniejszych kwadratów (`family = "gaussian"`) jest o wiele bardziej wrażliwa na obserwacje odstające niż domyślny M-estymator.

PRZYKŁAD 4

Interaktywne zaznaczanie skupień na dendrogramie

Jednym z nadrzędnych celów analizy skupień jest grupowanie podobnych obiektów w przestrzeni wielowymiarowej. Spośród wielu podejść do analizy skupień grupowanie hierarchiczne jest prawdopodobnie najczęściej wykorzystywane. Dendrogram jest popularnym narzędziem wizualizacji grupowania obiektów właśnie w hierarchicznej analizie skupień. Niestety nie jest on prosty w odczycie, nawet przy niewielkiej liczbie grupowanych obiektów: już powyżej 30 dendrogram staje się duży, etykiety obiektów robią się małe, a odczyt skupień – trudny. Dlatego w tym wypadku szczególnie warto skorzystać z interaktywnej wizualizacji.

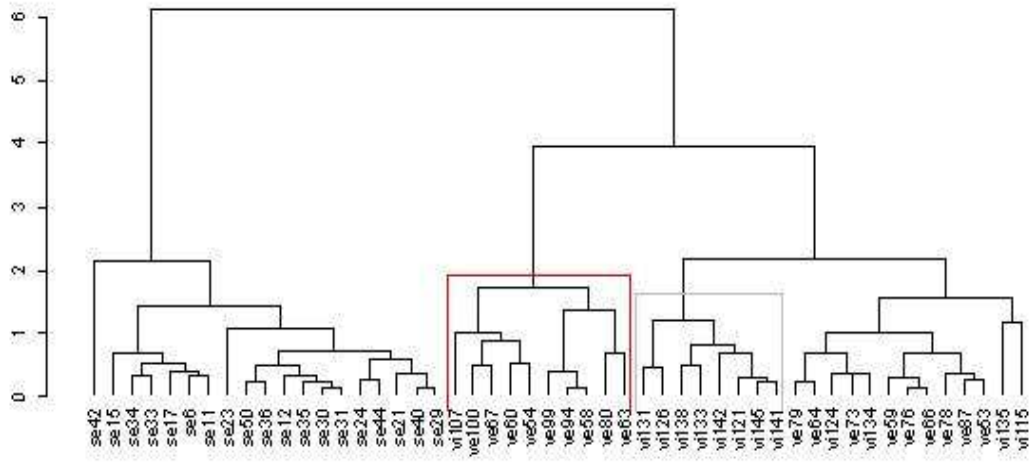
Pogrupujmy dla przykładu słynny zbiór danych dotyczących trzech gatunków irysa (Anderson, 1935; Fisher, 1936). Ponieważ w zbiorze jest 150 roślin – za dużo, by przedstawiać na dendrogramie – w drugim wierszu poniższego kodu losujemy 50 roślin spośród 150.

```
> data(iris)
> iiris <- iris[sample(1:150, 50), ]
> grupowanie <- hclust(dist(iiris[, 1:4]))
> rownames(iiris) <- paste(substr(iiris$Species, 1, 2),
rownames(iiris), sep = "")
> windows(width = 12); plot(as.dendrogram(grupowanie))
> identify(grupowanie, function(k) print(table(iiris[k,5])))
```

Rysunek 4 przedstawia dendrogram otrzymany po wywołaniu powyższego kodu. Aby uzyskać informację o danym skupieniu, należy kliknąć w jego bliskim otoczeniu. Spowoduje to zaznaczenie wybranej gałęzi dendrogramu czerwonym prostokątem. Na Rysunku 4 zaznaczono dwa skupienia (najpierw ten z prawej, następnie z lewej). Po zamknięciu okna wykresu w konsoli pojawiła się informacja o liczebności obserwacji dla trzech gatunków irysa w zaznaczonych skupieniach:

```
setosa versicolor  virginica
     0          0          8
setosa versicolor  virginica
     0          9          1
```

Z powyższego wynika, że w pierwszym zaznaczonym skupieniu znajduje się osiem obserwacji, wszystkie dla *I. virginica*, zaś w drugim skupieniu 10 obserwacji, z czego 9 dla *I. versicolor* i jedna dla *I. virginica*.



Rys. 4. Przykład interaktywnego zaznaczenia skupień na dendrogramie (Przykład 4)
Fig. 4. An example of interactive selection of clusters on a dendrogram

PRZYKŁAD 5

Pędzlowanie danych na macierzy wykresów rozrzutu

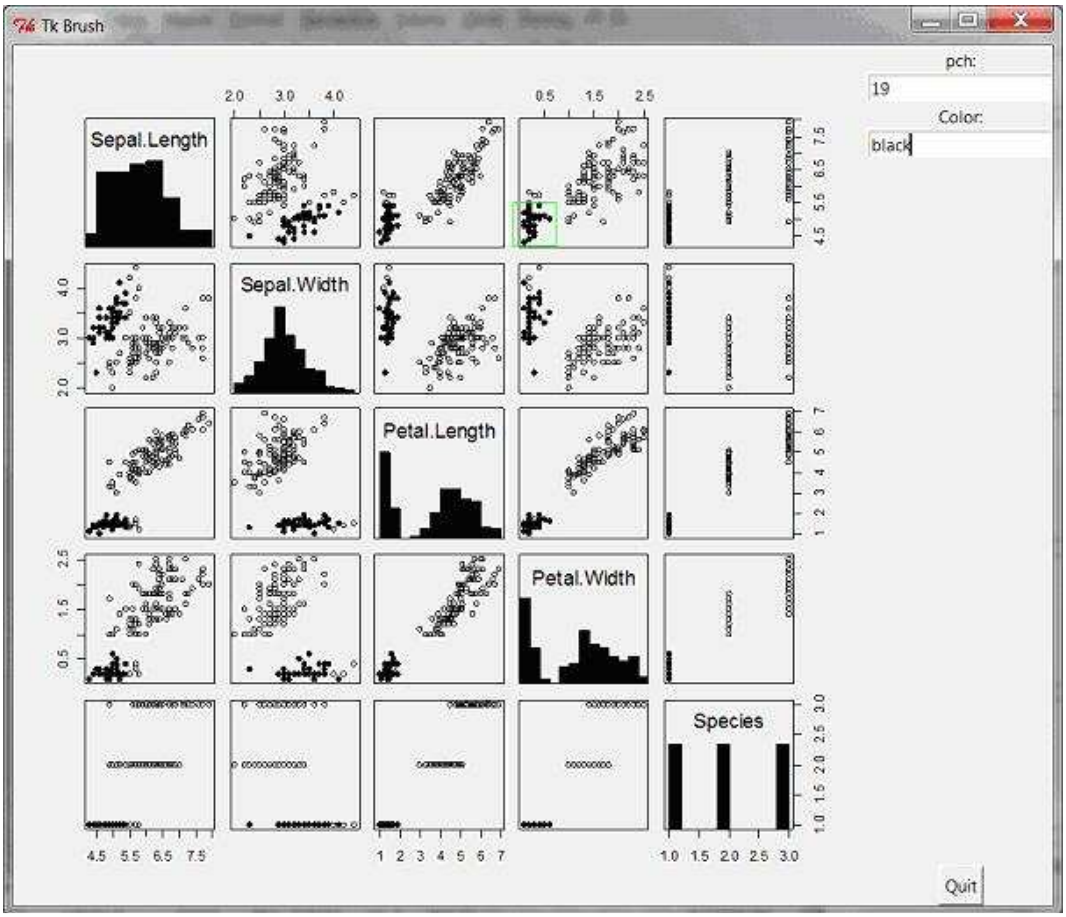
Pędzlowanie (ang. *brushing*) danych jest niezwykle użyteczną techniką interaktywnej eksploracji danych wielowymiarowych. Pierwszym krokiem jest narysowanie kilku wykresów, np. macierzy wykresów rozrzutu (ang. *scatterplot matrix*, Cleveland, 1994) dla wielu zmiennych, w której każda para zmiennych reprezentowana jest przez dwa wykresy rozrzutu: na pierwszym jedna ze zmiennych jest przedstawiona na osi poziomej, a na drugim na osi pionowej. Następnie w obrębie jednego z wykresów zaznaczamy wybrany fragment wykresu, co powoduje, że punkty, które znajdują się w tym obszarze, zostają zaznaczone (np. innym kolorem i/lub kształtem znaku) zarówno na tym, jak i na pozostałych wykresach. W ten sposób można zorientować się, jakie wartości różnych zmiennych przyjmują obserwacje z zaznaczonego obszaru. Najlepiej zorientować się, samemu stosując tę trudną do opisania, ale łatwą do zrozumienia technikę. W tym celu zainstalujmy dwa pakiety: `TeachingDemos` (Snow, 2010) i `tkrplot` (Tierney, 2010):

```
> install.packages("TeachingDemos")
> install.packages ("tkrplot")
```

Teraz wystarczy wczytać pierwszy z nich i uruchomić jedną funkcję:

```
> library(TeachingDemos)
> wykres <- tkBrush(iris)
```

Przykład okienka, w którym wykonywane jest pędzlowanie, przedstawiony jest na rys. 5.



Rys. 5. Przykład okienka macierzy wykresów rozrzutu podczas pędzlowania (Przykład 5). Na przekątnej przedstawione są histogramy zmiennych, zaś na pozostałych – wykresy zależności między zmienną wierszową i kolumnową. Ostatni wiersz i ostatnia kolumna przedstawiają zmienną jakościową (gatunek irysa), która jest zakodowana następująco: 1 = *I. setosa*, 2 = *I. versicolor*, 3 = *I. virginica*
Fig. 5. An example of a window containing a scatterplot matrix while brushing. The diagonal panels represent histograms of the variables, while the non-diagonal panels represent relationships between the row and column variables. The last row and last column represent a quality variable (iris species), which is coded as follows: 1 = *I. setosa*, 2 = *I. versicolor*, 3 = *I. virginica*

PODSUMOWANIE

W naszym przekonaniu wizualizacja danych powinna być nieodłącznym elementem każdej analizy statystycznej, ponieważ ułatwia i wzbogaca analizę oraz wnioskowanie. Interaktywna wizualizacja danych pozwala na jeszcze więcej, zwłaszcza w przypadku

danych wielowymiarowych, z jakimi mamy zwykle do czynienia w badaniach rolniczych. Dzięki niej możemy zrozumieć dane, wyszukać obserwacje nietypowe, zauważyć struktury w danych czy zależności między zmiennymi, które w innym wypadku było by trudno dostrzec i skutecznie przeanalizować.

Przykłady przedstawione powyżej są tylko niewielką próbą możliwości, jakie daje nam interaktywna wizualizacja. Naszym celem było wprowadzenie czytelnika w środowisko interaktywnej wizualizacji i zachęcenie go do wykorzystania jej technik zależnie od potrzeb. Tych technik jest o wiele więcej, ale nawet te, które zaprezentowaliśmy w pracy, mogą się okazać bardzo przydatne w codziennej analizie danych z badań rolniczych. Bardzo ciekawe możliwości w środowisku R oferuje na przykład pakiet iplots (Urbanek i Wichtrey, 2010).

LITERATURA

Anderson E. 1935. The irises of the Gaspe Peninsula, Bulletin of the American Iris Society, 59: 2 — 5.

Biecek P. 2008. Przewodnik po pakiecie R. Oficyna Wydawnicza GIS.

Cleveland W. S. 1993. Visualizing data. Hobart Press, Summit, NJ, USA.

Cleveland W. S. 1994. The elements of graphing data. 2ed. Hobart Press, Summit, N J, USA.

Fisher R. A. 1936. The use of multiple measurements in taxonomic problems. Annals of Eugenics, 7, Part II,: 179 — 188.

Friedman J. H., Stuetzle W. 2002. John W. Tukey's work on interactive graphics. The Annals of Statistics 30 (6): 1629 — 1639.

Kozak, M. 2010. Basic principles of graphing data. Scientia Agricola 67: 483 — 494.

Kozak, M., Bocianowski, J., Sawkojć, S., Wnuk, A. 2010. Call for more graphical elements in statistical teaching and consultancy. Biometrical Letters 47 (1): 57 — 68.

Mendiburu F. de. 2010. agricolae: Statistical procedures for agricultural research. R package version 1.0-9. http://CRAN.R-project.org/package=agricolae.

Snow G. 2010. TeachingDemos: Demonstrations for teaching and learning. R package version 2.7. http://CRAN.R-project.org/package=TeachingDemos.

Tierney L. 2010. tkrplot: TK Rplot. R package version 0.0-19. http://CRAN.R-project.org/package=tkrplot

Urbanek S., Wichtrey T. 2010. iplots: iPlots – interactive graphics for R. R package version 1.1-3. http://www.rosuda.org/iplots/.