SPRAWOZDANIE

Zajęcia: Zbiory Big Data i Eksploracja Danych

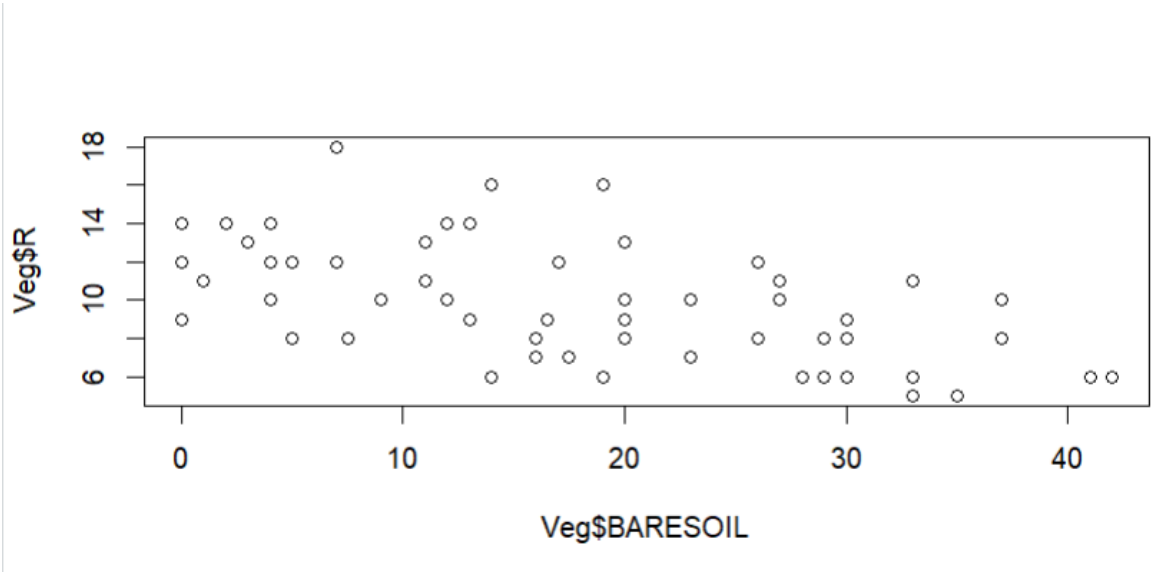
Prowadząca: dr inż. Ruslana Ziubina

Laboratorium nr 3 Data rozpoczęcia: 17.11.2023 Temat: Podstawowe wykresy i wizualizacja danych	Rafał Klinowski Informatyka II stopień, stacjonarne, Semestr 2, gr. a

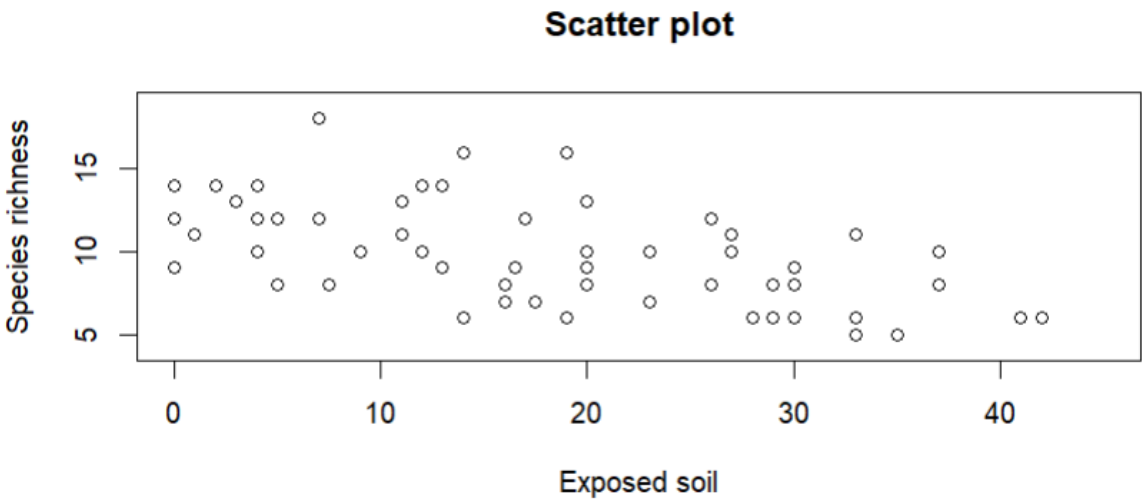
Poszczególne ćwiczenia będą wykonywane w pliku źródłowym edytowanym przy pomocy środowiska RStudio, opisanego w poprzedniej części laboratorium.

Ćw. 1.

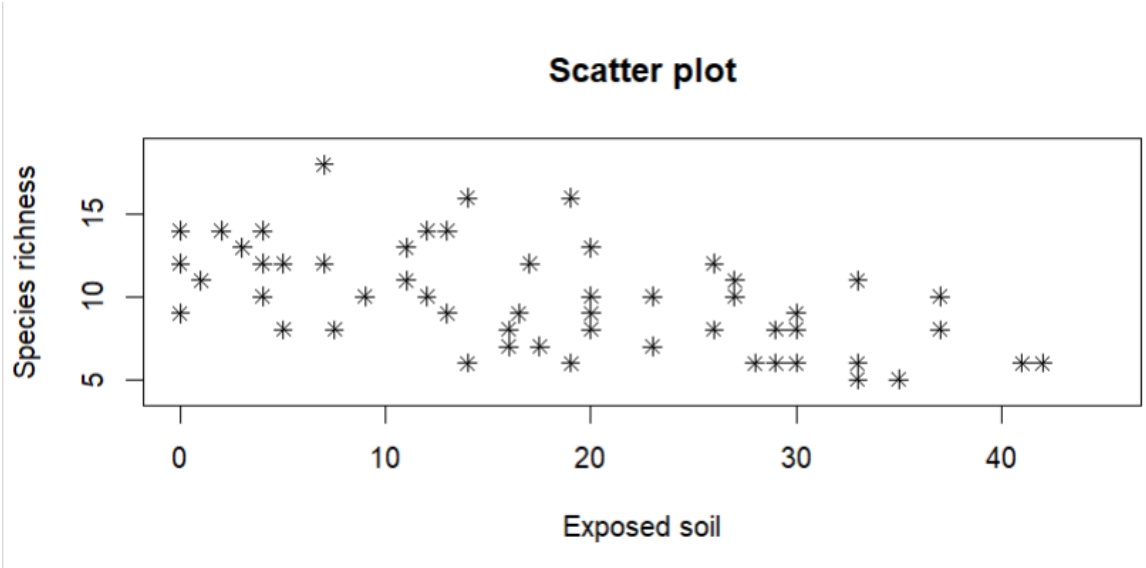
Na początku zapoznano się ze sposobami wyświetlania danych poprzez tworzenie prostych wykresów i wizualizacji.



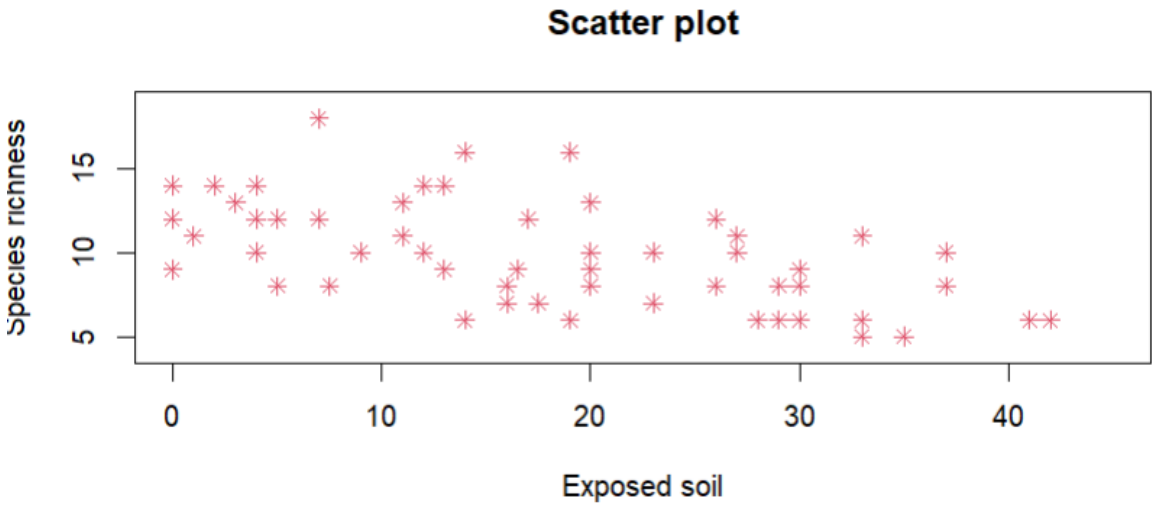
Rysunek 1. Prosty wykres utworzony za pomocą funkcji plot().



Rysunek 2. Przerobiony wykres po dodaniu tytułu, opisów osi oraz ich skalowania.

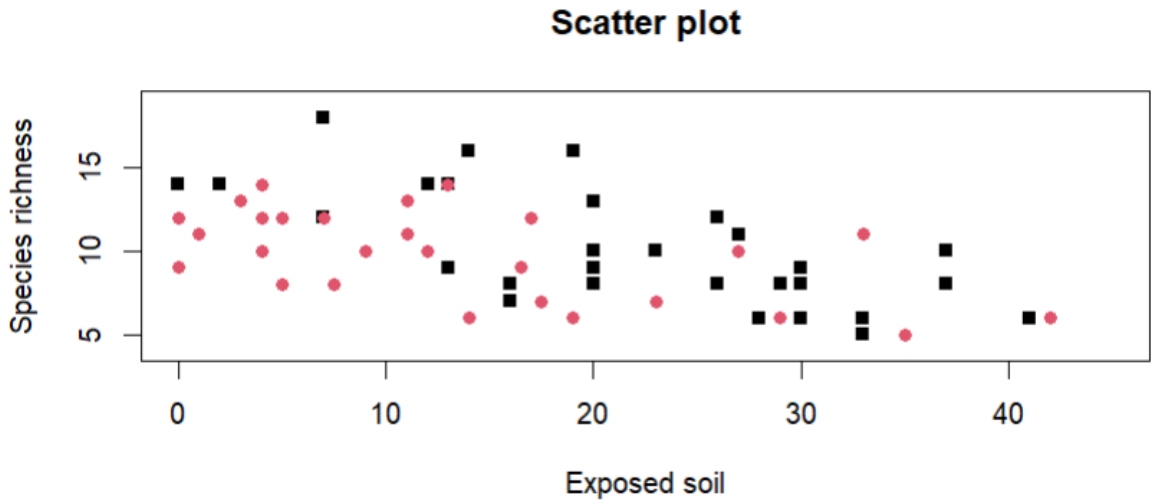


Rysunek 3. Powyższy wykres po zmianie symbolu punktów (atrybut pch).



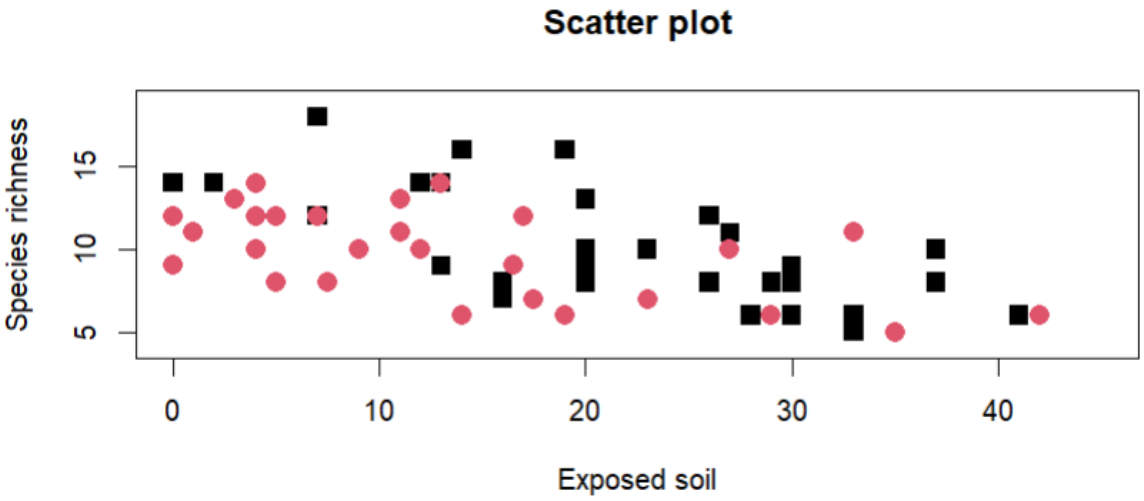
Rysunek 4. Zmiana koloru symboli (atrybut col).

Jako atrybuty pch oraz col można również przekazać wektory liczb, na przykład osobne kolumny zbioru danych.



Rysunek 5. Bardziej zaawansowany wykres z podziałem na dwa przedziały czasowe, zaznaczone innymi symbolami oraz kolorami.

Rozmiar symboli można zmienić przy pomocy parametru cex.



Rysunek 6. Powyższy wykres po zwiększeniu rozmiaru symboli.

Ćw. 2.

Graphics Devices in R. W ramach tego podpunktu zapoznano się z dokumentem opisującym podstawy tworzenia różnego typu wykresów, zmiany ich parametrów oraz przekierowywania wykresu do pliku wyjściowego zamiast bezpośrednio na ekran.

Ćw. 3.

Analiza zbioru danych dotyczącego zachowań piskląt sowy płomykówki. Korzystamy ze zbioru danych „Owls.txt”.

```
> Owls <- read.table(file="Owls.txt", header=TRUE)
> names(Owls)
[1] "Nest"               "FoodTreatment"      "SexParent"          "ArrivalTime"
[5] "SiblingNegotiation" "BroodSize"          "NegPerChick"
> str(Owls)
'data.frame':   599 obs. of  7 variables:
 $ Nest              : chr  "AutavauxTV" "AutavauxTV" "AutavauxTV" "AutavauxTV" ...
 $ FoodTreatment     : chr  "Deprived" "Satiated" "Deprived" "Deprived" ...
 $ SexParent         : chr  "Male" "Male" "Male" "Male" ...
 $ ArrivalTime       : num  22.2 22.4 22.5 22.6 22.6 ...
 $ SiblingNegotiation: int  4 0 2 2 2 2 18 4 18 0 ...
 $ BroodSize         : int  5 5 5 5 5 5 5 5 5 5 ...
 $ NegPerChick       : num  0.8 0 0.4 0.4 0.4 0.4 3.6 0.8 3.6 0 ...
```

Rysunek 7. Wczytanie zbioru danych i zapoznanie się z podstawowymi informacjami na jego temat.

Wykorzystanie polecenia unique() do wyodrębnienia nazw gniazd.

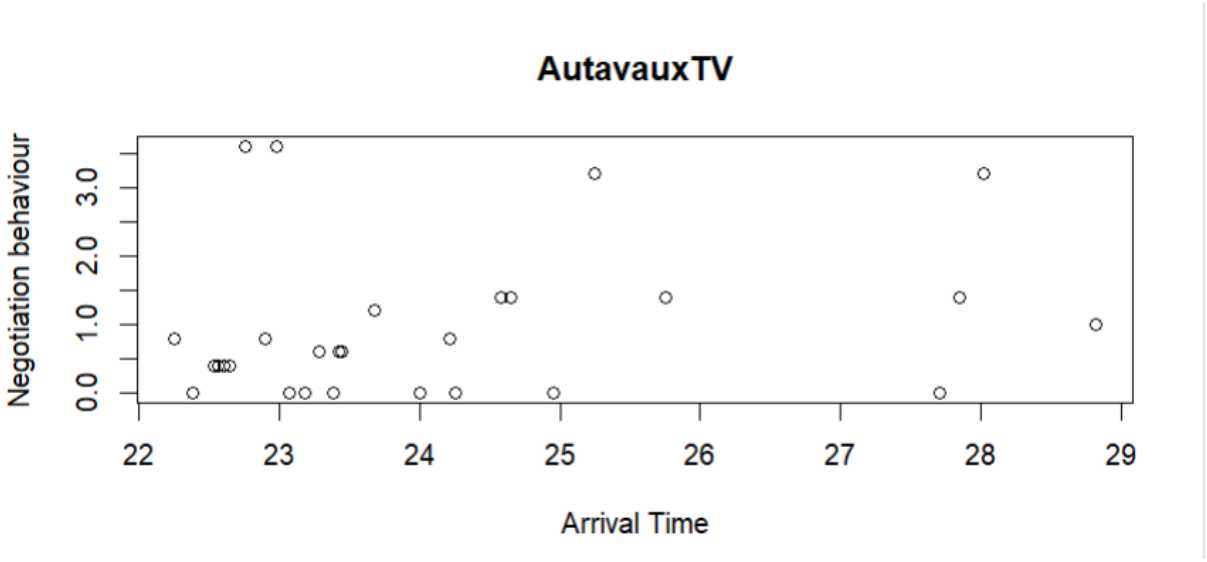
```
> unique(Owls$Nest)
 [1] "AutavauxTV"      "Bochet"          "Champmartin"     "ChEsard"         "Chevroux"
 [6] "CorcellesFavres" "Etrabloz"        "Forel"           "Franex"          "GDLV"
[11] "Gletterens"      "Henniez"         "Jeuss"           "LesPlanches"     "Lucens"
[16] "Lully"           "Marnand"         "Moutet"          "Murist"          "Oleyes"
[21] "Payerne"         "Rueyes"          "Seiry"           "SEvaz"           "StAubin"
[26] "Trey"            "Yvonnand"
```

Rysunek 8. Nazwy gniazd ze zbioru danych bez powtórzeń.

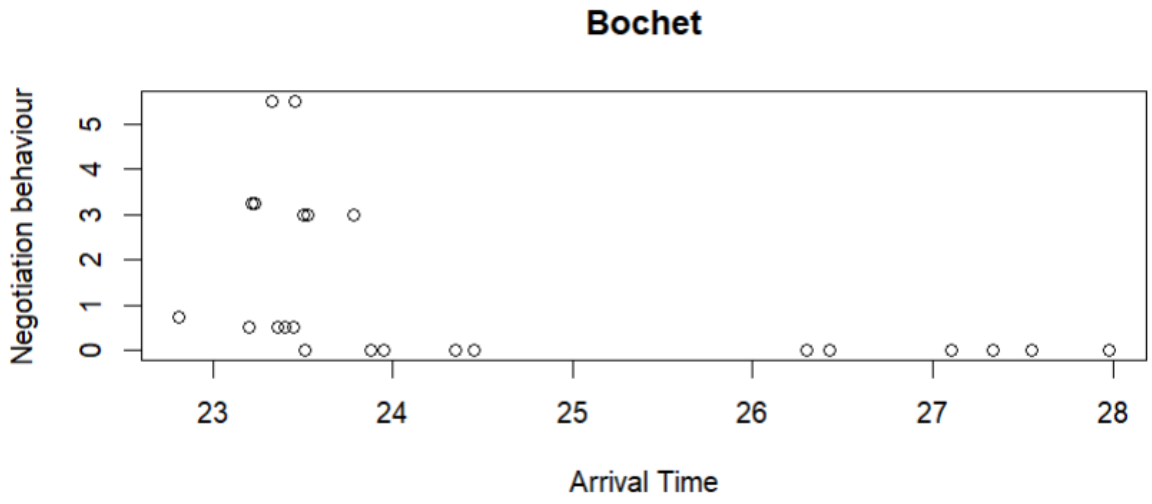
```
> Owls.ATV <- Owls[Owls$Nest == "AutavauxTV",]
> Owls.ATV
         Nest FoodTreatment SexParent ArrivalTime SiblingNegotiation BroodSize NegPerChick
1  AutavauxTV      Deprived      Male       22.25                 4         5         0.8
2  AutavauxTV      Satiated      Male       22.38                 0         5         0.0
3  AutavauxTV      Deprived      Male       22.53                 2         5         0.4
4  AutavauxTV      Deprived      Male       22.56                 2         5         0.4
5  AutavauxTV      Deprived      Male       22.61                 2         5         0.4
```

Rysunek 9. Pobranie danych dla jednego z gniazd.

Teraz utworzono wykres dla danych z tego gniazda.



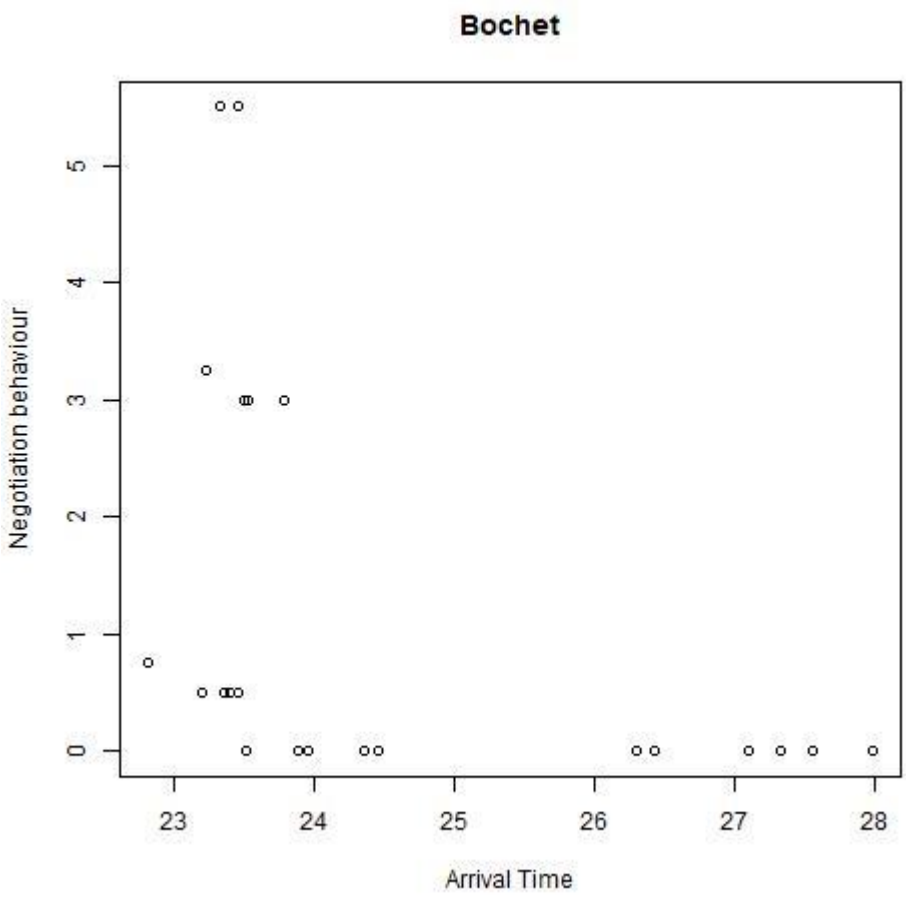
Rysunek 10. Wykres dla danych dotyczących gniazda „ATV”.



Rysunek 11. Analogiczny wykres dla gniazda „Bot”.

```
> plik <- paste(Nest.i, ".jpg", sep="")
> jpeg(file = plik)
> plot(x = Owls.i$ArrivalTime, y = Owls.i$NegPerChick,
+      xlab = "Arrival Time", main = Nest.i,
+      ylab = "Negotiation behaviour")
> dev.off()
```

Rysunek 12. Kod odpowiedzialny za przekierowanie wykresu do pliku zamiast bezpośrednio na ekran.

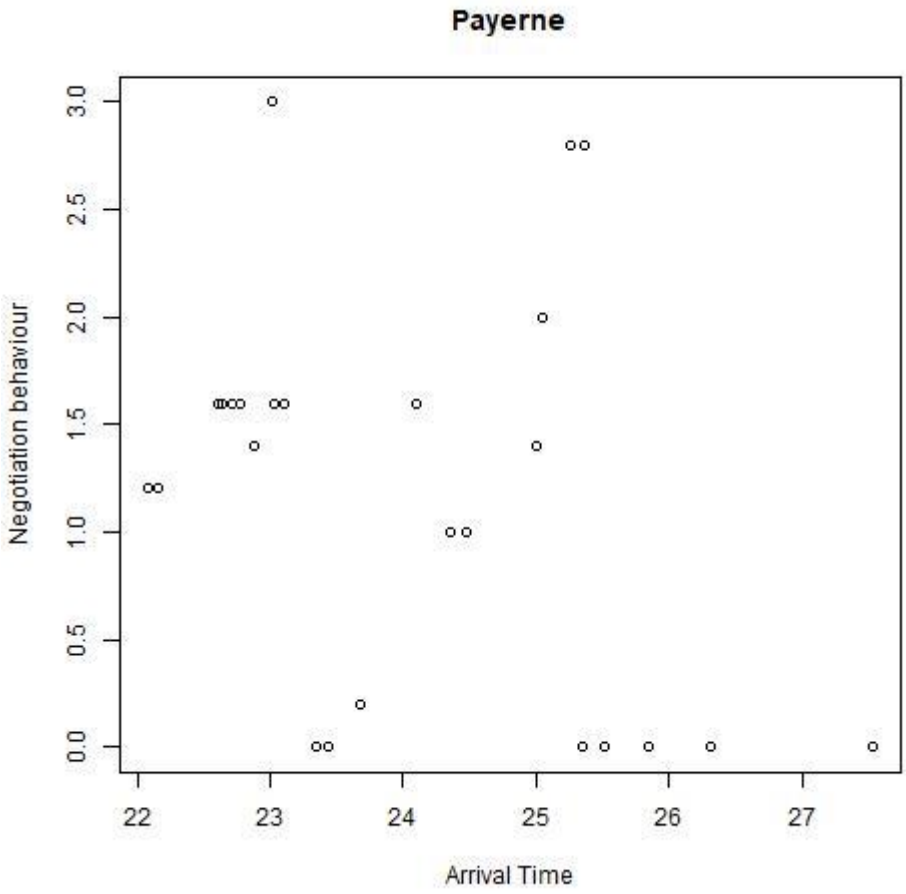


Rysunek 13. Uzyskany plik JPG.

Powyższy proces powtórzono w pętli dla wszystkich typów gniazd. Przeniesiono również wykresy do odpowiedniego podfolderu.

```
> for (i in 1:27){
+   Nest.i <- AllNests[i]
+   Owls.i <- Owls[Owls$Nest == Nest.i, ]
+   plik <- paste("Wykresy/", Nest.i, ".jpg", sep = "")
+   jpeg(file = plik)
+   plot(x = Owls.i$ArrivalTime, y = Owls.i$NegPerChick,
+        xlab = "Arrival Time",
+        ylab = "Negotiation behaviour", main = Nest.i)
+   dev.off()
+ }
```

Rysunek 14. Kod źródłowy realizujący powyższe polecenie.



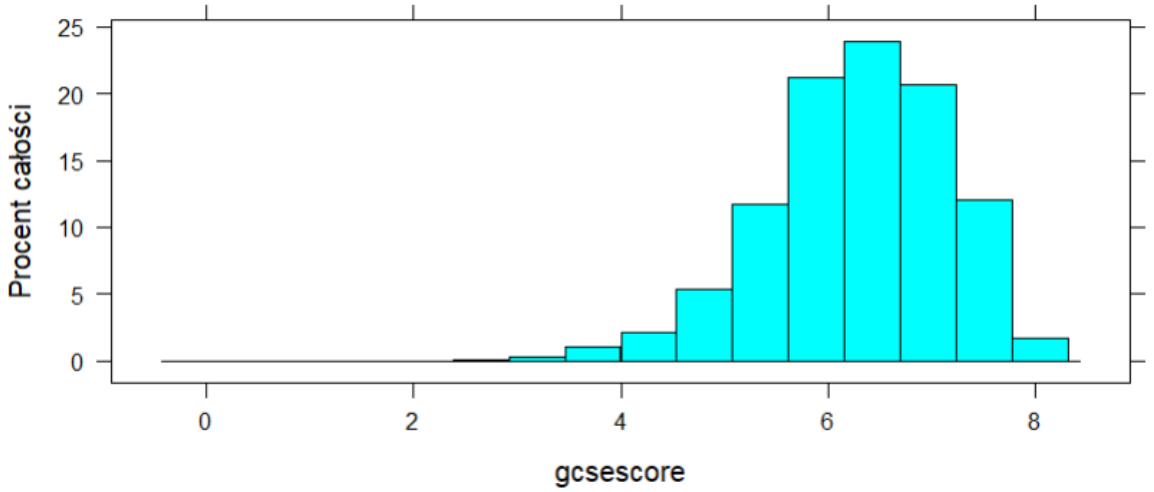
Rysunek 15. Przykładowy uzyskany plik JPG.

Ćw. 4.

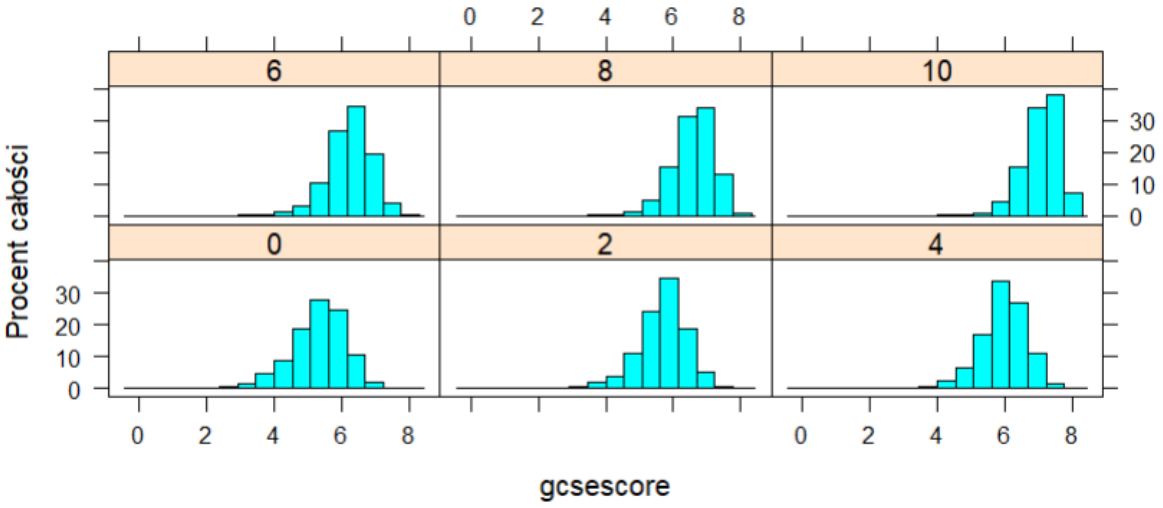
Po załadowaniu pakietu Lattice należało wykonać kilka przykładowych ćwiczeń związanych z jego obsługą.

```
> data(Chem97, package="mlmRev")
> head(Chem97)
  lea school student score gender age gcsescore   gcsecnt
1   1      1       1     4      F   3     6.625 0.3393157
2   1      1       2    10      F  -3     7.625 1.3393157
3   1      1       3    10      F  -4     7.250 0.9643157
4   1      1       4    10      F  -2     7.500 1.2143157
5   1      1       5     8      F  -1     6.444 0.1583157
6   1      1       6    10      F   4     7.750 1.4643157
```

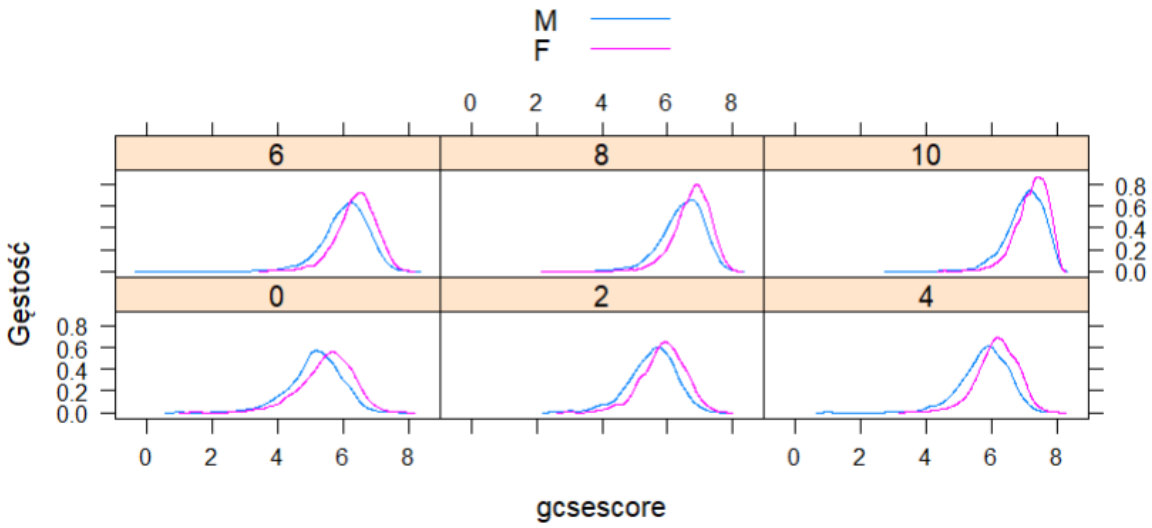
Rysunek 16. Załadowanie danych Chem97.



Rysunek 17. Utworzenie histogramu dla powyższych danych.



Rysunek 18. Utworzenie kilku histogramów dzielących dane na grupy w zależności od wyniku („score”).

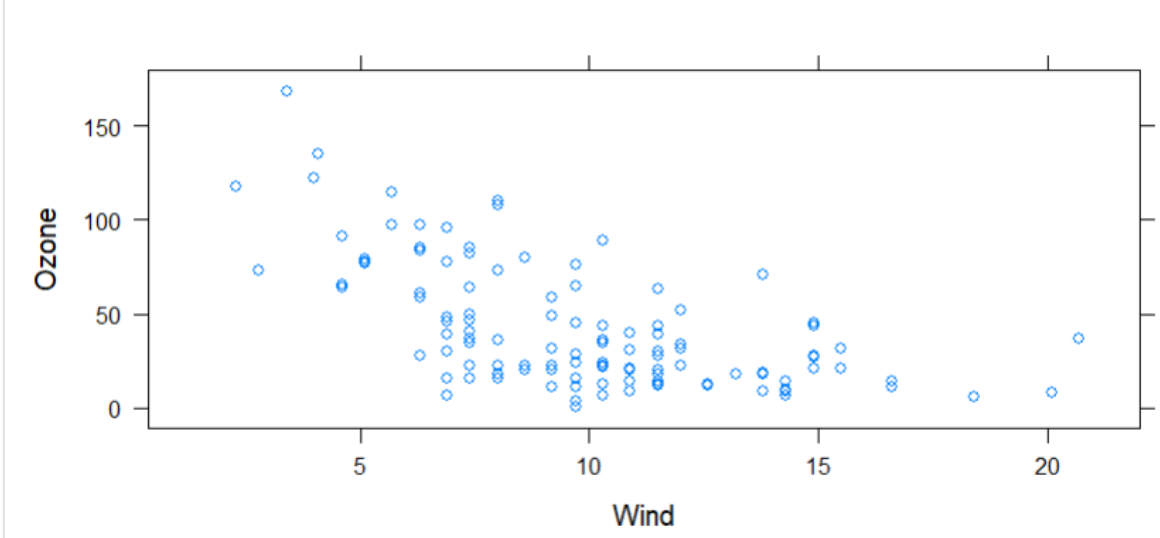


Rysunek 19. Utworzenie wykresu gęstości z podziałem na płcie.

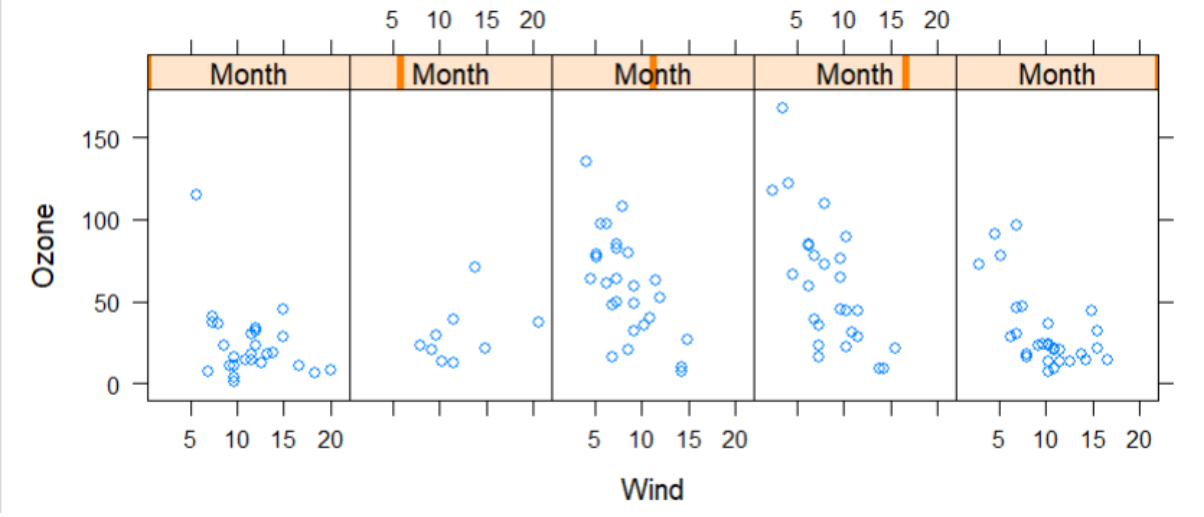
```
> histogram(~ gcsescore, data = Chem97)
> histogram(~ gcsescore | factor(score), data = Chem97)
> densityplot(~ gcsescore | factor(score), Chem97, groups = gender,
+             plot.points = FALSE, auto.key = TRUE)
```

Rysunek 20. Polecenia wykorzystanie do utworzenia powyższych wykresów.

Kolejne wykresy przygotowano na podstawie dokumentu „Plotting Lattice”.



Rysunek 21. Wykres dla danych dotyczących jakości powietrza – ozon w zależności od wiatru.



Rysunek 22. Powyższy wykres z podziałem na miesiące z zakresu 5-9.

W powyższym przykładzie wykorzystano „panele”, czyli sposób na wyświetlenie równolegle kilku wykresów (parametr „layout”). Panele tworzą się automatycznie gdy dane są dzielone na grupy za pomocą operatora |.

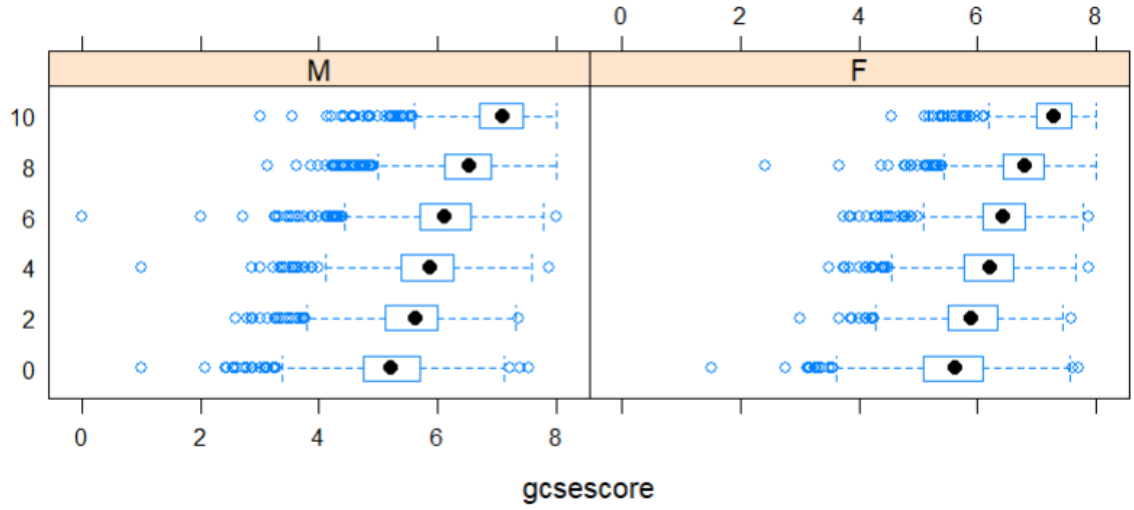
Główną różnicą w zachowaniu pakietu Lattice, w porównaniu do funkcji wbudowanych, jest to, że zwracają one obiekty klasy „trellis”, które mogą być przechowane, zachowanie czy zapisane. W przypadku gdy nie są one zapisane, obiekty są automatycznie wyświetlane na urządzeniu graficznym.

```
> p <- xyplot(Ozone ~ Wind, data=airquality)
> print(p)
```

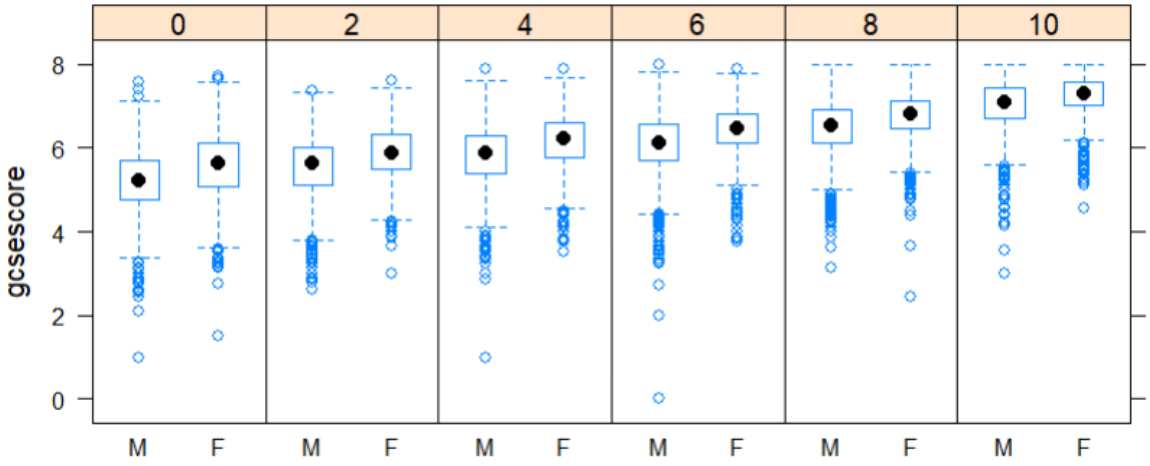
Rysunek 23. Sposób zachowania utworzonego wykresu jako zmiennej oraz jego wyświetlania.

Ćw. 6.

1. Wykonaj wykres z pliku „An Introduction to Lattice”, strona 8. Poznaj zbiór wykorzystany do tego wykresu.



Rysunek 24. Pierwszy z utworzonych wykresów.



Rysunek 25. Drugi z utworzonych wykresów.

```
> bwplot(factor(score) ~ gcsescore | gender, Chem97)
> bwplot(gcsescore ~ gender | factor(score), Chem97, layout = c(6, 1))
```

Rysunek 26. Kod napisany do utworzenia powyższych wykresów.

W ramach tego ćwiczenia utworzono dwa wykresy typu „bwplot”, czyli „box-and-whisker” z pakietu Lattice. Jest to wykres „pudełkowy”, w którym prostokąty są wyznaczone za pomocą kwartyli. Punkt w środku prostokątów oznacza medianę. Linie poza prostokątami oznaczają „odstęp ćwiartkowy”, natomiast pozostałe punkty poza tymi odcinkami to dane, które nie załapały się do tego odstępu, w niektórych przypadkach są to również wartości minimalne i maksymalne.

Wykres został utworzony na dwa sposoby – pierwszym razem z podziałem na płcie, drugim razem z podziałem na uzyskany wynik („score”). W wyniku tego uzyskano dwa

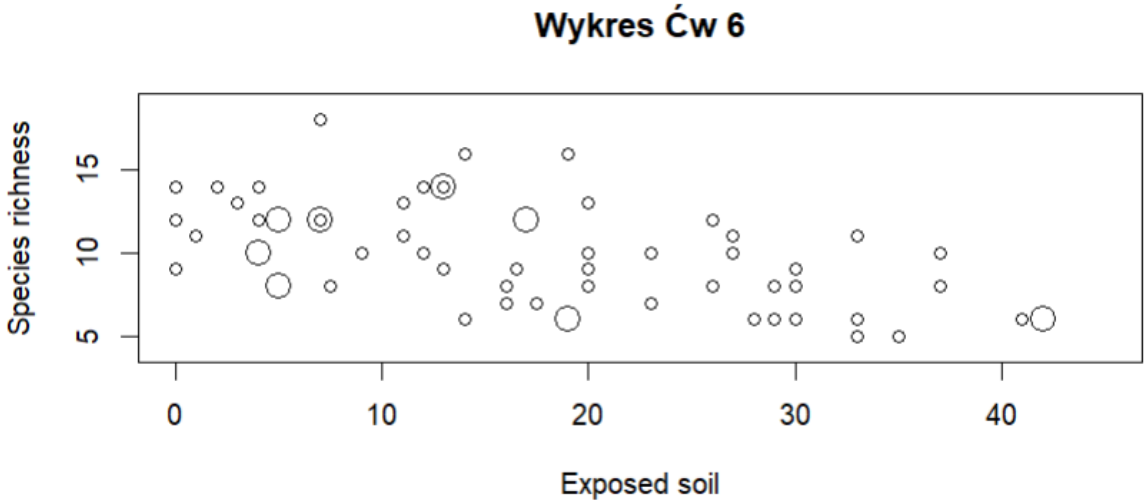
wykresy po 6 prostokątów, lub 6 wykresów po dwa prostokąty. Wyglądem steruje się za pomocą parametru „layout”.

Dane wykorzystane w tym ćwiczeniu to dane „Chem97”, czyli wyniki z egzaminu „A-levels” z chemii w Wielkiej Brytanii w 1997 roku. Zawiera ponad 31 tysięcy wierszy podzielonych na 8 zmiennych.

```
> str(Chem97)
'data.frame':   31022 obs. of  8 variables:
 $ lea      : Factor w/ 131 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ school   : Factor w/ 2410 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ student  : Factor w/ 31022 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ score    : num  4 10 10 10 8 10 6 8 4 10 ...
 $ gender   : Factor w/ 2 levels "M","F": 2 2 2 2 2 2 2 2 2 2 ...
 $ age      : num  3 -3 -4 -2 -1 4 1 4 3 0 ...
 $ gcsescore: num  6.62 7.62 7.25 7.5 6.44 ...
 $ gcsecnt  : num  0.339 1.339 0.964 1.214 0.158 ...
```

Rysunek 27. Zbiór danych Chem97.

2. Napisz skrypt, który do ćw. 1. Wygeneruje duże kółka dla obserwacji z 2002, a mniejsze dla innych lat.



Rysunek 28. Uzyskany wykres – większe kółka reprezentują obserwacje z 2002 roku.

```
Veg <- read.table(file="Vegetation2.txt", header=TRUE)

Veg$Year <- rep(1)
Veg$Year[Veg$Time == 2002] <- 2

plot(x = Veg$BARESOIL, y = Veg$R,
     xlab = "Exposed soil",
     ylab = "Species richness",
     main = "Wykres Ćw 6",
     xlim = c(0,45),
     ylim = c(4,19),
     cex=Veg$Year)
```

Rysunek 29. Kod napisany do utworzenia wykresu. Utworzono dodatkową kolumnę w zbiorze danych, domyślnie o wartościach 1, oraz przypisano wartość 2 do tych wierszy, dla których rok obserwacji wynosił 2002.

Wnioski.

Lattice jest bardzo użytecznym pakietem dającym znacznie więcej możliwości tworzenia wykresów. Funkcje tworzące wykresy mają prostszą składnię – łatwiej zdefiniować, co ma być na wykresie oraz jak ma być pogrupowane – oraz zwracają one obiekt reprezentujący utworzony wykres, który można następnie zapisać lub zachować do dalszej części realizowanego zadania. Ponadto, Lattice posiada spory wybór typów oraz styli wykresów, w tym między innymi wykres pudełkowy, który samoistnie oblicza odpowiednie kwartyle by podzielić dane.

R zawiera wiele wbudowanych zbiorów danych, które w łatwy sposób można podejrzeć czy wykorzystać w ramach ćwiczeń. Zbiory, które zostały wykorzystane podczas tego laboratorium to między innymi Chem97, czyli dość duży zbiór danych dotyczących egzaminu z chemii, oraz zbiór Airquality, czyli niewielki zbiór z pomiarami danych atmosferycznych wraz z datami. Łatwy dostęp do zbiorów danych umożliwia proste ćwiczenie wykorzystywania pewnych funkcji lub tworzenia przykładowych wizualizacji.

Całość laboratorium została przeprowadzona w RStudio, które znacznie ułatwia nie tylko tworzenie poleceń (dzięki kolorowaniu składni oraz podpowiadaniu nazw), ale również ich powtórzenie (w przypadku uruchomienia więcej niż raz) oraz podświetlenie zarówno ich wyników, jak i danych, na podstawie których te wyniki zostały uzyskane. Środowisko posiada również podgląd danych w formie tabeli lub jako wartość, co przydaje się w celu podejrzenia ich wyglądu i rozmiaru, jak również do weryfikacji wpisywanych poleceń.