

# Programowanie w języku R. Dane wielowymiarowe

Aleksander Denisiuk  
Uniwersytet Warmińsko-Mazurski  
Olsztyn, ul. Słoneczna 54  
[denisjuk@matman.uwm.edu.pl](mailto:denisjuk@matman.uwm.edu.pl)

6 maja 2020

# ***Dane wielowymiarowe***

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

Najnowsza wersja tego dokumentu dostępna jest pod adresem

<http://wmii.uwm.edu.pl/~denisjuk/uwm>

## Wykresy

- ❖ Wykresy 3D
- ❖ Wykresy złożone

## Piktogramy

## Składowe główne

## Klasteryzacja

## Analiza dyskryminacyjna

## Wybór metody

# Wykresy

# Wykresy danych wielowymiarowych

Wykresy

❖ Wykresy 3D

❖ Wykresy złożone

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

- Obniżyć wymiar danych do dwóch
- Wykresy trójwymiarowe

◆ czwarta cecha — kolorem

- Wykres `scatterplot3d`

```
library(scatterplot3d)
scatterplot3d(iris$Sepal.Length,
               iris$Sepal.Width, iris$Petal.Length,
               color=as.numeric(iris$Species))
```

- Wykres `rgl` (interaktywny)

```
library(rgl)
plot3d(iris$Sepal.Length, iris$Sepal.Width,
        iris$Petal.Length,
        col=as.numeric(iris$Species), size=3)
```

# Wykresy złożone

Wykresy

❖ Wykresy 3D

❖ Wykresy złożone

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

- Biblioteka `lattice`

```
library(lattice)
xyplot(Sepal.Length ~ Petal.Length +
       Petal.Width | Species,
       data=iris, auto.key=TRUE)
```

- Wykres `coplot()`

```
coplot(Petal.Length ~ Sepal.Length |
       Species, data=iris)
```

- Wykres `pairs()`

```
pairs(iris[1:4], pch=21, bg =
      c("red", "green3", "blue")
      [unclass(iris$Species)])
```

◆ `unclass()` — zamienia atrybut na liczbę

Wykresy

**Piktogramy**

- ❖ Gwiazdy
- ❖ Twarze Chernoffa
- ❖ Wykres współrzędnych równoległych

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

# Piktogramy

# Gwiazdy

Wykresy

Piktogramy

❖ Gwiazdy

❖ Twarze Chernoffa

❖ Wykres  
współrzędnych  
równoległych

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

```
stars (mtcars[1:8, 1:7], cex=1.2,  
      key.loc = c(7.5, 2.25))
```

- Każdy promień — to cecha obiektu
  - ✦ długość odpowiada wielkości

# Twarze Chernoffa

Wykresy

Piktogramy

❖ Gwiazdy

❖ Twarze Chernoffa

❖ Wykres  
współrzędnych  
równoległych

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

```
library(TeachingDemos)  
faces(mtcars[1:9, 1:7])
```

- Poszczególne zmienne odzwierciedlane są przez charakterystyki twarzy
  - ✦ wielkość oczu, wielkość źrenic, pozycja źrenic, skośność oczu, etc
    - do 18 cech



# Wykres współrzędnych równoległych

Wykresy

Piktogramy

❖ Gwiazdy

❖ Twarze Chernoffa

❖ Wykres  
współrzędnych  
równoległych

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

```
measurements <- read.table("data/eq-s.txt",  
  h=T, sep=";")  
library(MASS)  
parcoord(measurements[, -1],  
  col=rep(rainbow(54), table(measurements[, 1])))
```

- Pomiarzy cech roślin
- Pierwsza kolumna — miejsce pomiaru (54)
  - ◆ `rainbow()` generuje ciągłą paletę kolorów
    - co robi `table()`?

Wykresy

Piktogramy

**Składowe główne**

❖ Algorytm

❖ R

❖ Wkład

❖ ade4

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

# Analiza głównych składowych

# Algorytm

Wykresy

Piktogramy

Składowe główne

❖ Algorytm

❖ R

❖ Wkład

❖ ade4

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

- Dane są punktami w przestrzeni  $\mathbb{R}^n$
- Zmienić współrzędne:
  - ✦ pierwsza współrzędna zawiera najwięcej informacji
    - druga mniej
    - trzecia jeszcze mniej
    - etc...
- Dla analizy można uwzględnić tylko kilka pierwszych współrzędnych
  - ✦ dla wizualizacji tylko dwie

# Algorytm

[Wykresy](#)

[Piktogramy](#)

[Składowe główne](#)

❖ **Algorytm**

❖ R

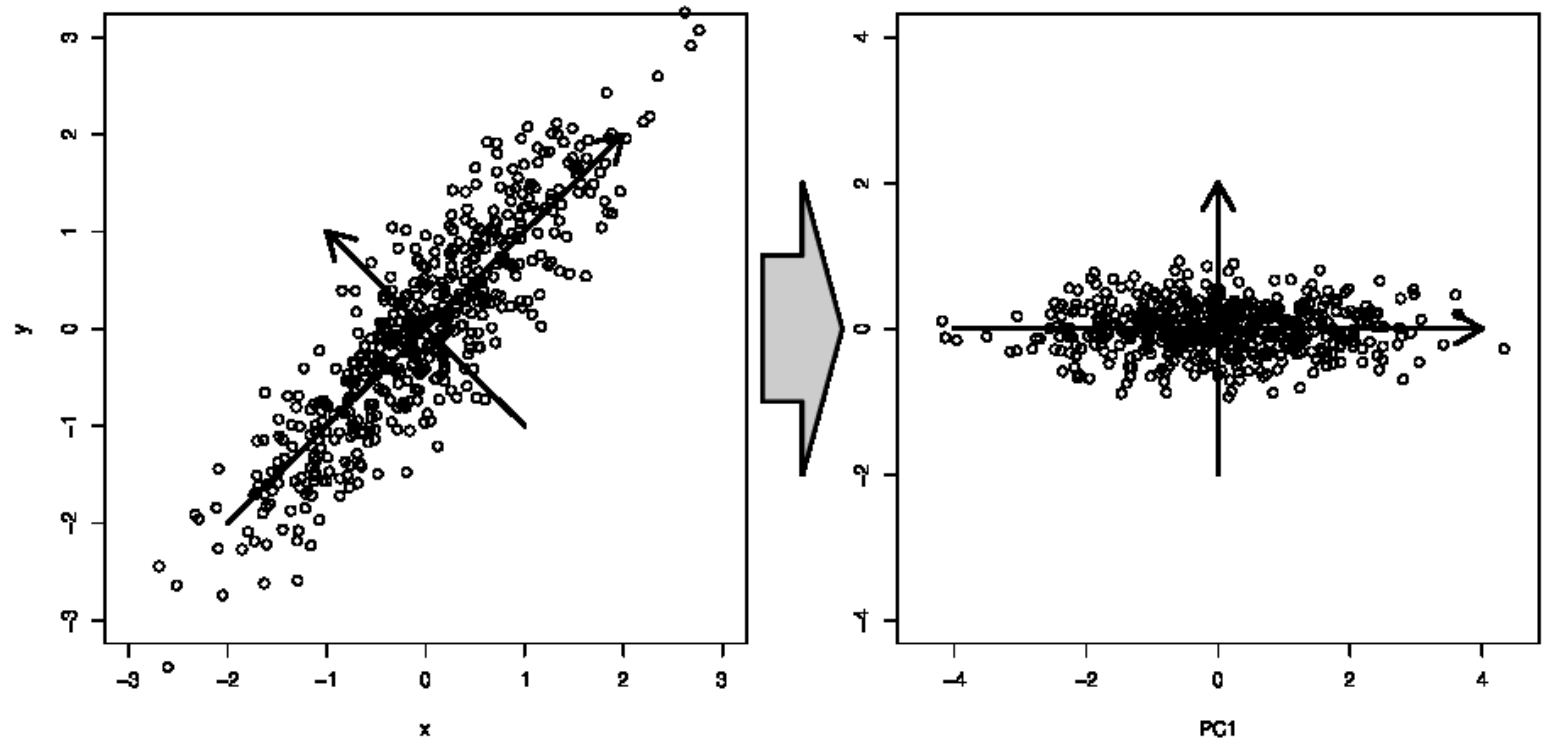
❖ Wkład

❖ ade4

[Klasteryzacja](#)

[Analiza  
dyskryminacyjna](#)

[Wybór metody](#)



# Analiza w R

Wykresy

Piktogramy

Składowe główne

❖ Algorytm

❖ R

❖ Wkład

❖ ade4

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

- Przeskalować dane: `scale()`

- Funkcja `princomp()`

```
iris.pca <- princomp(scale(iris[,1:4]))
```

- Zobaczyć wyniki

```
summary(iris.pca)
```

```
plot(iris.pa)
```

# Wizualizacja

Wykresy

Piktogramy

Składowe główne

❖ Algorytm

❖ R

❖ Wkład

❖ ade4

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

- Przeliczamy dane na główne składowe `predict()`

- ❖ funkcja *generyczna*

```
iris.p <- predict(iris.pca)
plot(iris.p[,1:2], type="n",
      xlab = "PC1", ylab = "PC2")
text(iris.p[,1:2], labels =
      abbreviate(iris[,5],1, method="both.sides"))
```

- Który gatunek się różni?

# Wkład cech do komponent głównych

Wykresy

Piktogramy

Składowe główne

❖ Algorytm

❖ R

❖ Wkład

❖ ade4

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

- Graficznie: `biplot()`
- Tekstowo: `loadings()`

```
biplot(iris.pca)
```

```
loadings(iris.pca)
```

# Pakiet ade4

Wykresy

Piktogramy

Składowe główne

❖ Algorytm

❖ R

❖ Wkład

❖ ade4

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

- `dudi.pca()` — przeprowadza analizę składowych głównych (Duelity Diagram)

- ◆ opcja `scannf=FALSE` zostawia dwie główne składowe

- Wykres rozproszenia: `s.class()`

```
library(ade4)
iris.dudi <- dudi.pca(iris[,1:4],
                     scannf=FALSE)
s.class(iris.dudi$li, iris[,5])
```



# Jakość rozróżniania klas

Wykresy

Piktogramy

Składowe główne

❖ Algorytm

❖ R

❖ Wkład

❖ ade4

Klasteryzacja

Analiza  
dyskryminacyjna

Wybór metody

- `bca()` — *factor* jest jedyną zmienną objaśnialną (Between-Class Analysis)

- ◆ opcja `scannf=FALSE` zostawia dwie główne składowe

- Testowanie klasyfikacji: `randtest()`

- ◆ jakość: parametr `Observations`

- ◆ funkcja *generyczna*

```
iris.between <- bca(iris.dudi, iris[,5],  
  scannf=FALSE)  
randtest(iris.between)
```

Wykresy

Piktogramy

Składowe główne

**Klasteryzacja**

- ❖ Klasyfikacja bezwzorcową
- ❖ MDS
- ❖ hclust
- ❖ kmeans
- ❖ fuzzy clustering
- ❖ Correspondence analysis

Analiza dyskryminacyjna

Wybór metody

# Analiza skupień

# Klasyfikacja bezwzorcowa

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

❖ Klasyfikacja  
bezwzorcowa

❖ MDS

❖ hclust

❖ kmeans

❖ fuzzy clustering

❖ Correspondence  
analysis

Analiza  
dyskryminacyjna

Wybór metody

- Klasyfikacja bez nadzoru (unsupervised learning)
- Grupowanie elementów we względnie jednorodne klasy (klastry)
- Podobieństwo pomiędzy elementami
- Funkcja podobieństwa (metryka)
  - ◆ euklidesowa
  - ◆ Manhattan
- Macierz niepodobieństwa
  - ◆ Pakiet `cluster`

```
library(cluster)
iris.dist <- daisy(iris[,1:4],
                  metric="manhattan")
```

    - dla danych nominalnych zostanie wykorzystana metryka Gowera

# Skalowanie wielowymiarowe

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

❖ Klasyfikacja  
bezwzorcowa

❖ MDS

❖ hclust

❖ kmeans

❖ fuzzy clustering

❖ Correspondence  
analysis

Analiza  
dyskryminacyjna

Wybór metody

- Dana jest macierz niepodobieństwa (macierz odległości)
- Ustawić obiekty jako punkty w przestrzeni  $n$ -wymiarowej, aby wzajemne odległości pokrywały się z daną macierzą

```
example(cmdscale)
```

- dla danych Iris

```
iris.c <- cmdscale(iris.dist)
plot(iris.c[,1:2], type="n",
      xlab="Dim. 1", ylab="Dim. 2")
text(iris.c[,1:2],
      labels=abbreviate(iris[,5], 1,
                        method="both.sides"))
```

- Dane nieparametryczne: `isoMDS()`

# Wizualizacja

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

❖ Klasyfikacja  
bezwzorcowo

❖ MDS

❖ hclust

❖ kmeans

❖ fuzzy clustering

❖ Correspondence  
analysis

Analiza  
dyskryminacyjna

Wybór metody

```
library(KernSmooth)
est <- bkde2D(iris.c[,1:2], bandwidth=c(.7,1.5))
plot(iris.c[,1:2], type="n", xlab="Dim. 1",
      ylab="Dim. 2") text(iris.c[,1:2],
labels=abbreviate(iris[,5],1,
method="both.sides"))
contour(est$x1, est$x2, est$fhat, add=TRUE,
drawlabels=FALSE, lty=3)
```

- `bkde2D()` szacuje gęstość danych
- `contour()` rysuje „mapę”

# Klasteryzacja hierarchiczna

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

❖ Klasyfikacja  
bezwzorcowo

❖ MDS

❖ **hclust**

❖ kmeans

❖ fuzzy clustering

❖ Correspondence  
analysis

Analiza  
dyskryminacyjna

Wybór metody

```
iriss <- iris[seq(1,nrow(iris),5),]  
iriss.dist <- daisy(iriss[,1:4])  
iriss.h <- hclust(iriss.dist, method="ward.D")  
plot(iriss.h, labels=abbreviate(iriss[,5],1,  
                                method="both.sides"), main="")
```

- Metoda grupowania Warda
- `iris[seq(1,nrow(iris),5),]??`

# Stabilność skupień (klastrów)

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

❖ Klasyfikacja  
bezwzorcowo

❖ MDS

❖ hclust

❖ kmeans

❖ fuzzy clustering

❖ Correspondence  
analysis

Analiza  
dyskryminacyjna

Wybór metody

```
library(pvclust)
irisst <- t(iris[,1:4])
colnames(irisst) <- paste(abbreviate(iris[,5], 3),
  colnames(irisst))
irisst.pv <- pvclust(irisst,
  method.dist="manhattan",
  method.hclust="ward", nboot=1000)
plot(irisst.pv, col.pv=c(1,0,0), main=" ")
```

- Metoda Bootstrap
- Przy każdym węźle ocena stabilności
  - ◆ dobra stabilność  $\approx 100$

# Metoda k-średnich

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

❖ Klasyfikacja  
bezwzorcowa

❖ MDS

❖ hclust

❖ **kmeans**

❖ fuzzy clustering

❖ Correspondence  
analysis

Analiza  
dyskryminacyjna

Wybór metody

```
iris.k <- kmeans(iris[,1:4], 3)  
table(iris.k$cluster, iris$Species)
```

- Dana jest ilość klastrów
- Na wejściu są dane, a nie macierz niepodobieństwa



# Metody rozmytej analizy skupień

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

❖ Klasyfikacja  
bezwzorcowo

❖ MDS

❖ hclust

❖ kmeans

❖ fuzzy clustering

❖ Correspondence  
analysis

Analiza  
dyskryminacyjna

Wybór metody

```
iris.f <- fanny(iris[,1:4], 3)
plot(iris.f, which=1, main=" ")
head(data.frame(sp=iris[,5], iris.f$membership))
```

- Dana jest ilość klastrów
- Na wejściu są dane, a nie macierz niepodobieństwa

# Analiza odpowiedniości

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

❖ Klasyfikacja  
bezwzorcowo

❖ MDS

❖ hclust

❖ kmeans

❖ fuzzy clustering

❖ Correspondence  
analysis

Analiza  
dyskryminacyjna

Wybór metody

```
library(MASS)
```

```
corresp(caith)
```

```
biplot(corresp(caith, nf = 2))
```

- Dane: kolor oczu i kolor włosów ludzi na Caithness
- `corresp()` — na podstawie macierzy kontyngencji oblicza nowe parametry (zmiana układu współrzędnych), tak, aby korelacja Pearsona była maksymalną
- `nf = 2` — uwzględnić pierwsze dwa parametry (koniecznie dla `biplot()`)

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

**Analiza  
dyskryminacyjna**

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

# Analiza dyskryminacyjna

# Klasyfikacja wzorcowa

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

- Znane są klasy danych
- Wypracować wskaźniki, pozwalające na klasyfikacje
  - ◆ klasyfikacja nowych danych
  - ◆ określenie ważności cech do klasyfikacji
    - część danych do nauczania
    - druga część — do kontroli

# *Liniowa analiza dyskryminacyjna*

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

## ● linowe kombinację cech

```
library(MASS)
iris.train <- iris[seq(1,nrow(iris),5),]
iris.unknown <- iris[-seq(1,nrow(iris),5),]
iris.lda <- lda(iris.train[,1:4],
               iris.train[,5])
iris.ldap <- predict(iris.lda,
                    iris.unknown[,1:4])$class
table(iris.ldap, iris.unknown[,5])
```

# Oszacowanie klasyfikacji

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

```
misclass <- function(pred, obs) {  
  tbl <- table(pred, obs)  
  sum <- colSums(tbl)  
  dia <- diag(tbl)  
  msc <- (sum - dia)/sum * 100  
  cat("Classification table:\n")  
  print(tbl)  
  cat("Misclassification errors:\n")  
  print(round(msc, 1))  
}
```

- skrypt functions/misclass.r

```
misclass(iris.ldap, iris.unknown[,5])
```

# Statystyczne oszacowanie klasyfikacji

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

- Wilowymiarowa analiza wariancji (MANOVA) `manova()`

```
ldam <- manova(  
  as.matrix(iris.unknown[,1:4]) ~ iris.ldap )  
summary(ldam)
```

- dla klasy `manova` funkcja `summary()` ma dodatkowy parametr `test='Wilks'`

- ◆ współczynnik podobieństwa grup

- im bliżej zera, tym bardziej grupy się różnią

```
summary(ldam, test="Wilks")
```

# Wizualizacja

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

```
iris.lda2 <- lda(iris[,1:4], iris[,5])
iris.ldap2 <- predict(iris.lda2, dimen=2)$x
plot(iris.ldap2, type="n",
      xlab="LD1", ylab="LD2")
text(iris.ldap2,
      labels=abbreviate(iris[,5], 1,
                        method="both.sides"))
```

- Do klasyfikacji użyto wszystkich danych



# Drzewa decyzyjne

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

```
library(tree)
iris.tree <- tree(iris[,5] ~ ., iris[, -5])
plot(iris.tree)
text(iris.tree)
```

- Do klasyfikacji użyto wszystkich danych

# Lasy losowe

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

```
library(randomForest)
iris.rf <-
  randomForest(iris.train[,5] ~ .,
               data=iris.train[,1:4])
iris.rfp <- predict(iris.rf, iris.unknown[,1:4])
table(iris.rfp, iris.unknown[,5])
```

- Losuje dużą ilość drzew decyzyjnych

# Wizualizacja

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

```
iris.urf <- randomForest(iris[, -5])  
MDSplot(iris.urf, iris[, 5])
```

## ● Klasyfikacja bezwzorcowa

# Maszyna wektorów nośnych

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

❖ Klasyfikacja  
wzorcowa

❖ LDA

❖ Decision trees

❖ Random forest

❖ SVM

Wybór metody

```
library(e1071)
iris.svm <- svm(Species ~ ., data = iris.train)
iris.svmp <- predict(iris.svm, iris[,1:4])
table(iris.svmp, iris[,5])
```

- Rozdziela się klasy płaszczyzną (hiperpłaszczyzną)

Wykresy

Piktogramy

Składowe główne

Klasteryzacja

Analiza  
dyskryminacyjna

**Wybór metody**

❖ Wybór

# Wybór metody

# Wybór odpowiedniej metody analizy

- Wykresy
- Piktogramy
- Składowe główne
- Klasteryzacja
- Analiza dyskryminacyjna
- Wybór metody
- ❖ Wybór

