Projekt ze Sztucznej inteligencji, 2018/2019:

Grupowanie i klasyfikacja obiektów

Adrian Marcinczyk, Jakub Woroniecki

# Spis Treści

[**Spis Treści**](#_3brm7nozlbh7) **1**

[**1. Specyfikacja sprzętu i oprogramowania użytego do wykonania zadań.**](#_pc8g8jcjcl8i) **2**

[**2. Opis zbioru danych i wykorzystanych metod**](#_clqzqnjla6e8) **2**

[**3. Grupowanie obiektów**](#_c87bpkmjhv7w) **3**

[**4. Klasyfikacja obiektów**](#_dee2rahe7w23) **8**

[**5. Wnioski**](#_dxiixnhsuo4a) **10**

# 1. Specyfikacja sprzętu i oprogramowania użytego do wykonania zadań.

**Tabela parametrów poszczególnych podzespołów:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Procesor | Pamięć RAM | Karta graficzna | Dysk Twardy | Architektura Systemu |
| Intel CORE i5-4460  taktowanie 3.2GHz  4 rdzenie | Zainstalowana pamięć RAM 8 GB | NVIDIA GeForce GTX 745 | ATA ST1000DX001-SSHD  1 TB pojemności  64 MB pamięci podręcznej  prędkość obrotowa 7200 obr./min. | Komputer oparty na architekturze x64 obsługujący interfejs ACPI |

Zadania zostały wykonane za pomocą języka **R** w wersji **3.6.** Kod został skompilowany przy pomocy oprogramowania **RStudio**.

# 2. Opis zbioru danych i wykorzystanych metod

Zbiorem danych użytym do klasyfikacji jest zbiór opisujący parametry związane z chorobą serca (zbiór “heart.csv”) pobrany ze strony Kaggle (<https://www.kaggle.com/ronitf/heart-disease-uci>). Zbiór nie posiada brakujących wartości, został już wcześniej przygotowany do dalszej analizy. Zbiór danych klasyfikuje posiadanie przez pacjenta choroby serca, zależnie od określonych cech. Tymi cechami są:

* age - wiek w latach
* sex - (1 = mężczyzna ; 0 = kobieta)
* cp - typ bólu klatki piersiowej
* trestbps - ciśnienie krwi trakcie spoczynku
* chol - cholesterol w surowicy w mg/dl
* fbs - (ilość cukru we krwi na czczo > 120 mg/dl) (1 = true; 0 = false)
* restecg - wyniki elektrokardiograficzne w trakcie spoczynku
* thalach - maksymalne tętno
* exang - duszności w trakcie wysiłku (1 = tak; 0 = nie)
* oldpeak - zagłębienie ST spowodowane wysiłkiem fizycznym w stosunku do odpoczynku
* slope - nachylenie ST w trakcie szczytowego wysiłku fizycznego
* ca - liczba głównych naczyń (0-3) zabarwionych przez fluoroskopię
* thal - 3 = normalny; 6 = stała wada; 7 = wada odwracalna
* target - ma chorobę lub nie (1=tak, 0=nie)

Klasa decyzyjna nie została określona, dlatego jako klasę decyzyjną wybieramy parametr **target posiadający dwie wartości (**1 dla posiadania choroby i 0 dla nieposiadania choroby)**.**

Zbiór zawiera 76 atrybutów, ale wszystkie opublikowane eksperymenty odnoszą się do podzbioru 14 z nich.

Zbiorem danych użytym do przeprowadzenia grupowania obiektów, jest zbiór dotyczący współczynników przestępstw w poszczególnych Stanach w USA. Zbiór został pobrany ze strony Kaggle (<https://www.kaggle.com/deepakg/usarrests>**).** Zbiór nie posiada brakujących wartości, został już wcześniej przygotowany do dalszej analizy.

Ten zestaw danych zawiera statystyki, w aresztach na 100 000 mieszkańców za napaść, morderstwo i gwałt w każdym z 50 stanów USA w 1973 r. Zbiór zawiera łącznie 5 atrybutów i 50 obserwacji.

# 3. Grupowanie obiektów

Do przeprowadzania klasyfikacji obiektów użyto metody **c-średnich** (fuzzy clustering), oraz metody **k-średnich** (k-means).

**Wczytanie niezbędnych bibliotek, zbioru danych, a także standaryzacja zbioru.**

library(tibble)

library(ggplot2)

library(class)

library(cluster)

library(factoextra)

library("NbClust")

data <- read.csv("USArrests.csv")

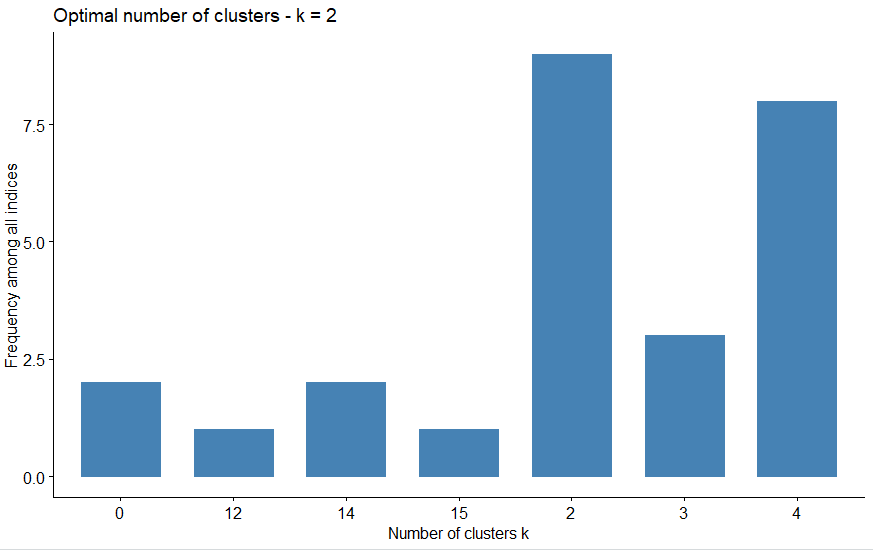
df <- scale(USArrests) # Standaryzacja zbioru

**Wyznaczenie optymalnej liczby skupień; metoda ward.D2**

res.nbclust <- NbClust(data = df, diss = NULL, distance = "euclidean", min.nc = 2, max.nc = 15, method = "ward.D2", index = "all", alphaBeale = 0.1)

fviz\_nbclust(res.nbclust, ggtheme = theme\_minimal())

**Wynik działania funkcji fviz\_nbclust**



**Metoda grupowania fuzzy clustering**

res.fanny <- fanny(df, 2)

res.fanny2 <- fanny(df, 3)

head(res.fanny$clustering) # Grupy obserwacji

fviz\_cluster(res.fanny, ellipse.type = "norm", repel = TRUE,

palette = "jco", ggtheme = theme\_minimal(),

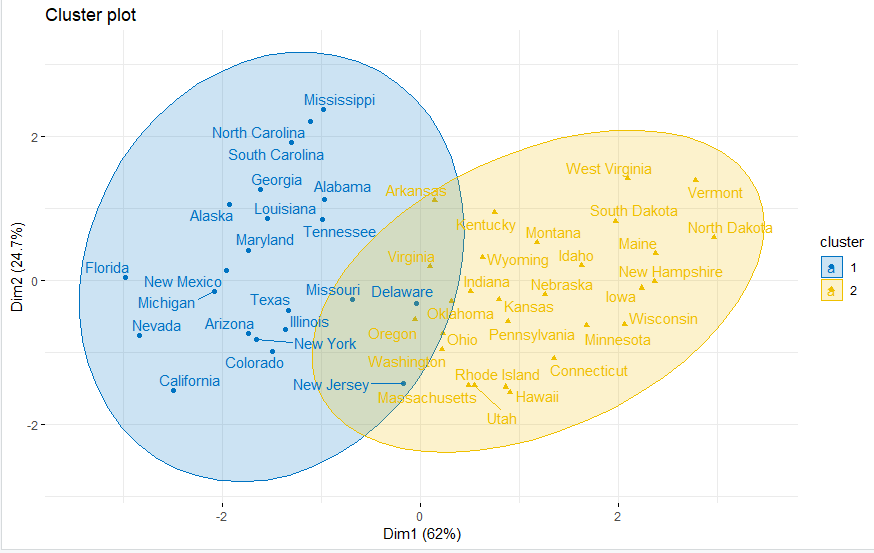
legend = "right")

fviz\_cluster(res.fanny2, ellipse.type = "norm", repel = TRUE,

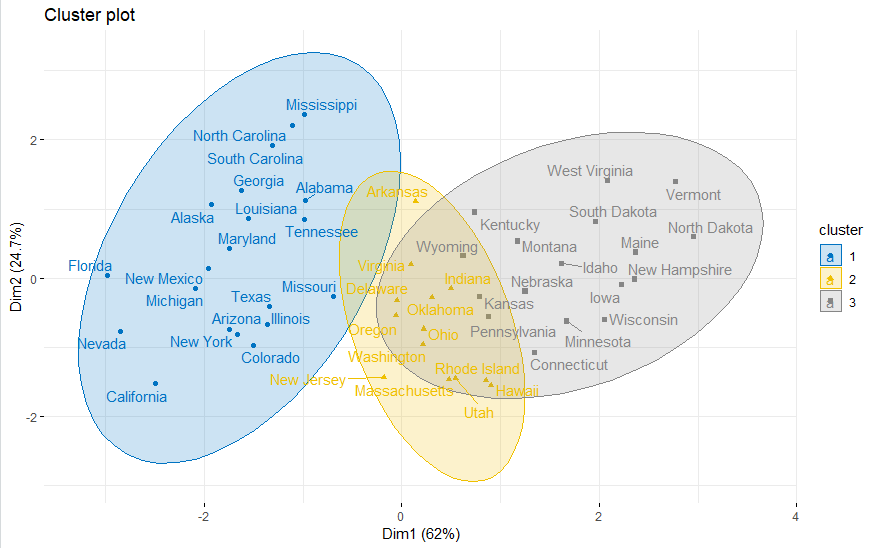
palette = "jco", ggtheme = theme\_minimal(),

legend = "right")

**Wynik działania fviz\_cluster dla 2 skupień**



**Wynik działania fviz\_cluster dla 3 skupień**



**Metoda grupowania k-means**

set.seed(123)

kus\_arrests<-kmeans(df, centers =2, nstart = 50)

kus\_arrests2<-kmeans(df, centers =3, nstart = 50)

kus\_arrests3<-kmeans(df, centers =4, nstart = 50)

plot(x=df[,1], y=df[,2], col=kus\_arrests$cluster)

points(kus\_arrests$centers, pch=3, cex=2)

clusplot(df, kus\_arrests$cluster, color = T, labels = 2, main = 'Cluster Plot')

plot(x=df[,1], y=df[,2], col=kus\_arrests2$cluster)

points(kus\_arrests2$centers, pch=3, cex=2)

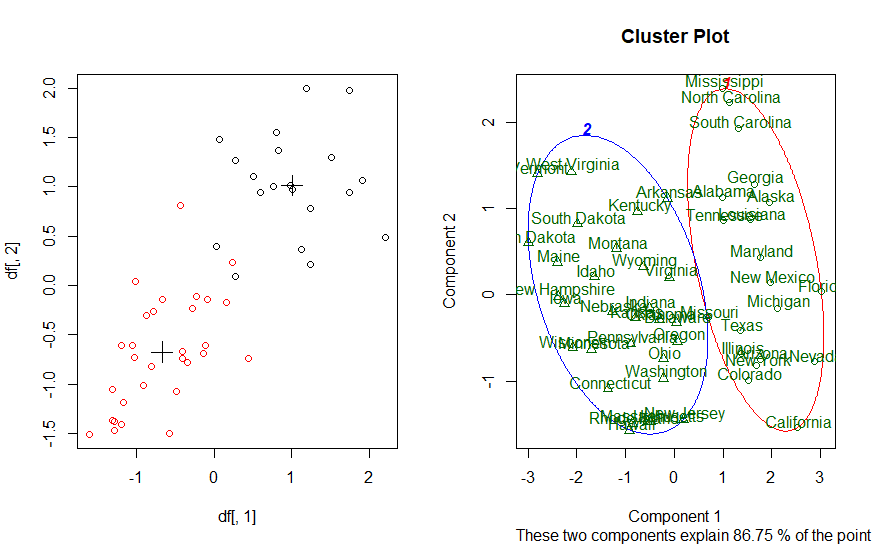
clusplot(df, kus\_arrests2$cluster, color = T, labels = 2, main = 'Cluster Plot')

plot(x=df[,1], y=df[,2], col=kus\_arrests2$cluster)

points(kus\_arrests2$centers, pch=3, cex=2)

clusplot(df, kus\_arrests2$cluster, color = T, labels = 2, main = 'Cluster Plot')

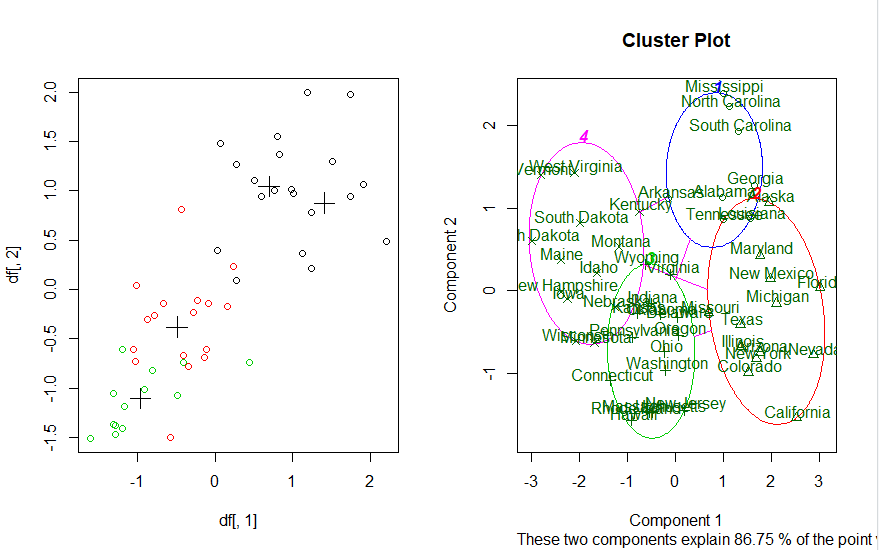
**Wykres grupowania k-means dla k = 2**



**Wykres grupowania k-means dla k = 3**

# 

**Wykres grupowania k-means dla k = 4**



# 4. Klasyfikacja obiektów

Do przeprowadzania klasyfikacji obiektów użyto metody k-NN oraz metody lasu losowego. Jako metody trenowania i testowania użyto prostego podziału na zbiór data.train (90 % danych) i data.test (10 % danych). Jest to odpowiednio 272 i 30 obserwacji.

**Wczytanie niezbędnych bibliotek, zbioru danych, także podział na zbiór testowy i treningowy, normalizacja danych, wskazanie atrybutu decyzyjnego**

*library(rpart)*

*library(caret)*

*library(e1071)*

*library(class)*

*library(randomForest)*

*data2 <- read.csv("heart.csv")*

*ran <- sample(1:nrow(data2), 0.9 \* nrow(data2))*

*nor <-function(x) { (x -min(x))/(max(x)-min(x)) }*

*data2\_norm <- as.data.frame(lapply(data2[,c(1,2,3,4,5,6,7,8,9,10,11,12,13)], nor))*

*summary(data2\_norm)*

*data2.train <- data2\_norm[ran,]*

*data2.test <- data2\_norm[-ran,]*

*data2\_target\_category <- data2[ran,14]*

*data2.test\_category <- data2[-ran,14]*

**Zbudowanie modelu klasyfikacji k-nn wraz z predykcją dla 3 modeli**

*library(class)*

*pr <- knn(data2.train,data2.test,cl=data2\_target\_category,k=5)*

*pr2 <- knn(data2.train,data2.test,cl=data2\_target\_category,k=14)*

*pr3 <- knn(data2.train,data2.test,cl=data2\_target\_category,k=30)*

*##confusion matrix*

*tab <- table(pr,data2.test\_category)*

*tab2 <- table(pr2,data2.test\_category)*

*tab3 <- table(pr3,data2.test\_category)*

*accuracy <- function(x){sum(diag(x)/(sum(rowSums(x)))) \* 100}*

*accuracy(tab)*

*accuracy(tab2)*

*accuracy(tab3)*

**Tabela dokładności predykcji modelu k-nn przy określeniu klasy atrybutu decyzyjnego jako “target” (wartości 0 lub 1)**

|  |  |  |
| --- | --- | --- |
| **Model użyty do eksperymentu:** | **Wartość parametru k** | **Dokładność predykcji (w %)** |
| pr | 5 | 93.54839 |
| pr2 | 14 | 80.64516 |
| pr3 | 30 | 74.19355 |

Zostały przeprowadzone badania dla wartości k < 5, lecz dokładność predykcji była największa w przypadku k = 5. W innych przypadkach działanie modelu k-NN nie przynosi porównywalnych rezultatów.

**Zbudowanie modelu klasyfikacji lasu losowego wraz z oceną OOB dla 3 modeli.**

data2\_target\_category <- as.factor(data2\_target\_category)

set.seed(415)

model1 <- randomForest(as.factor(data2\_target\_category) ~ ., data=data2.train, importance=TRUE, ntree=2000)

model2 <- randomForest(as.factor(data2\_target\_category) ~ ., data=data2.train, importance=TRUE, ntree=2000, mtry=7)

model3 <- randomForest(as.factor(data2\_target\_category) ~ ., data=data2.train, importance=TRUE, ntree=2000, mtry=2)

print(model1)

print(model2)

print(model3)

**Tabela oceny OOB dla 3 modeli**

|  |  |  |
| --- | --- | --- |
| **Model użyty do eksperymentu:** | **Liczba zmiennych użytych do podziału** | **OOB oszacowanie poziomu błędu** |
| model1 | 3 | 16.18% |
| model2 | 7 | 18.38% |
| model3 | 2 | 17.28% |

W każdym modelu liczba drzew była taka sama i wynosiła 2000. W pierwszym modelu funkcja randomForest() automatycznie dobrała optymalną liczbę zmiennych użytych do podziału, wynoszącą 3. W modelu drugim i trzecim liczba zmiennych użytych do podziału została ustawiona ręcznie (odpowiednio 7 i 2). Najmniejszy poziom OOB uzyskano przy modelu pierwszym.

# 

# 5. Wnioski

Obie metody skupienia tj. **k-średnich** i **c-średnich** osiągnęły identyczny poziom podziału na 2 skupienia ( 86,75 %). W przypadku metody k-średnich, dla naszego zbioru możemy przyjąć **k=4** jako maksymalną wartość podziału na skupienia, ponieważ algorytm dla **k > 4** nie dzieli zbioru na odpowiednie skupienia.

Najbardziej optymalną ilością skupień jest **k = 2.**

|  |  |
| --- | --- |
| **Nazwa metody grupowania:** | **Poziom wyjaśnienia punktów:** |
| k-means | 24.7 + 62.05 = 86.75 % |
| fuzzy clustering | 86.75 % |

W przypadku metod klasyfikacji, metoda **lasu losowego** osiągnęła **83.82%** dokładności klasyfikacji, przy **93.54 %** dokładności klasyfikacji w przypadku metody **k-nn**. Metoda k-nn okazała się najdokładniejszą metodą klasyfikacji spośród dwóch jakie brały udział w eksperymentach.

|  |  |
| --- | --- |
| **Nazwa metody klasyfikacji:** | **Maksymalna dokładność klasyfikacji:** |
| random forest | 83.82 % |
| k-nn | 93.54 % |