ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №1
«КЛАСТЕРИЗАЦІЯ З ВІДОМОЮ КІЛЬКІСТЮ КЛАСТЕРІВ»

частина 1

Ломако О., 1 к. маг, «статистика», варіант 9

В першій частині першої лабораторної роботи першого семестру другого курсу магістратури проведемо кластерний аналіз даних методами центроїдів і медоїдів.

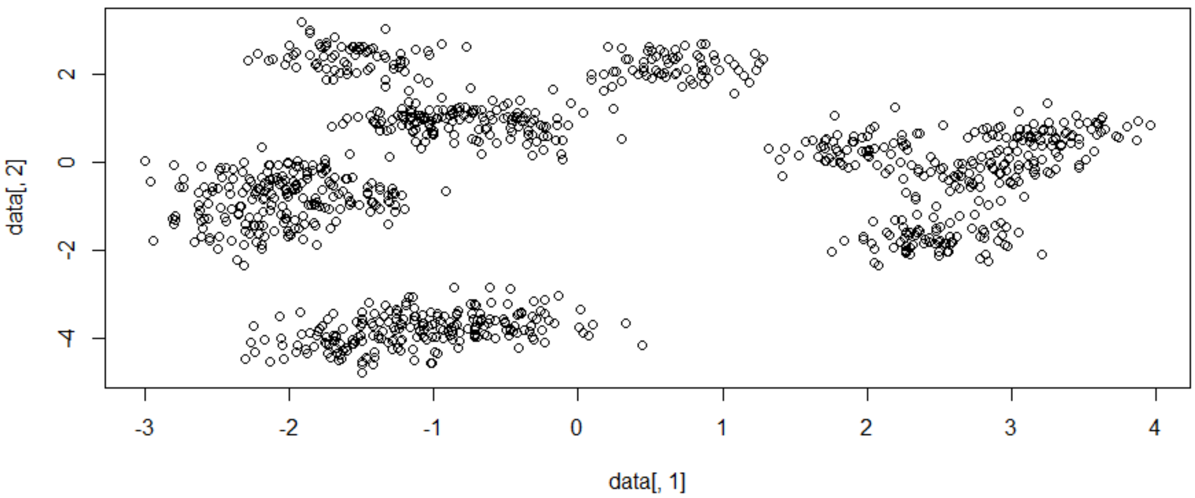
Спершу, як завжди, почнемо зі зчитування даних.

```
> library(factoextra) # підключаємо бібліотеки
>
> # встановлюємо робочу директорію
> data <- read.table('C:\\Users\\Razor\\Desktop\\дистанційне навчання\\статис
тичний аналіз багатовимірних даних\\lab1\\mult6.txt')
```

Оскільки в майбутньому нам доведеться відображати діаграми розсіювання із розмалюванням відповідних кластерів (в роботі пропонується розглянути до 20), задамо, на майбутнє, палітру 20 кольорів (про всяк випадок).

```
> # задаємо палітру кольорів
> col = c('black', 'red', 'green', 'blue', 'orange',
+         'purple', 'yellow', 'brown', 'burlywood',
+         'deepskyblue', 'darkseagreen', 'deeppink',
+         'salmon', 'turquoise1', 'darkblue', 'darkred',
+         'aquamarine', 'grey', 'chocolate', 'magenta')
```

Виведемо спершу діаграму розсіювання, наприклад, перших двох змінних (без розфарбування).

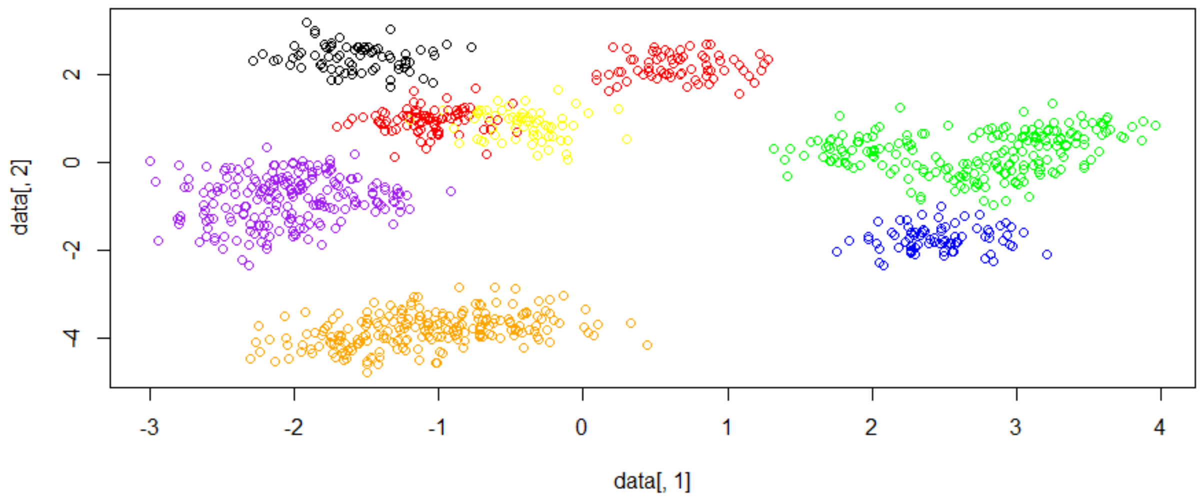


Візуально важко сказати скільки конкретно кластерів тут можна виділити. Спробуємо застосувати метод центроїдів. Спробуємо припустити що їх тут, нехай, 7.

```
> # метод центроїдів
```

```
> km.res <- kmeans(data, 7, nstart = 25)
> km.res$betweenss/km.res$tot.withinss # відношення міжкластерної суми квадра
тів до внутрішньокластерної
[1] 15.50091
> plot(data[,1], data[,2], col = col[km.res$cluster]) # діаграма розсіювання
перших двох змінних з кластеризацією
```

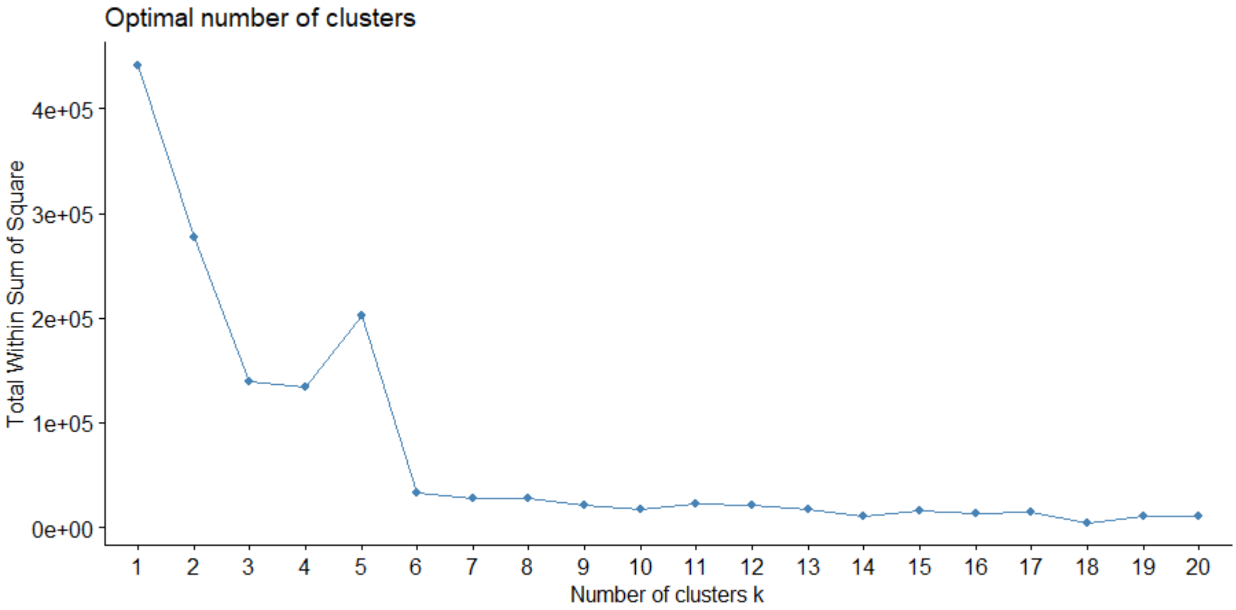
Тут внутрішньокластерна сума квадратів є в 15.5 разів меншою за міжкластерну, а отже можемо з певною впевненістю стверджувати, що тут виділяється якась структура даних. Відповідна діаграма розсіювання з розмалюванням:



Здавалось би, непогано, але чомусь посеред кластеру виділеним червоним присутній кластер розмальований жовтим. Не зовсім, на мою думку, логічно.

Тому побудуємо діаграму, яка в залежності від кількості кластерів показуватиме внутрішньогрупову суму квадратів.

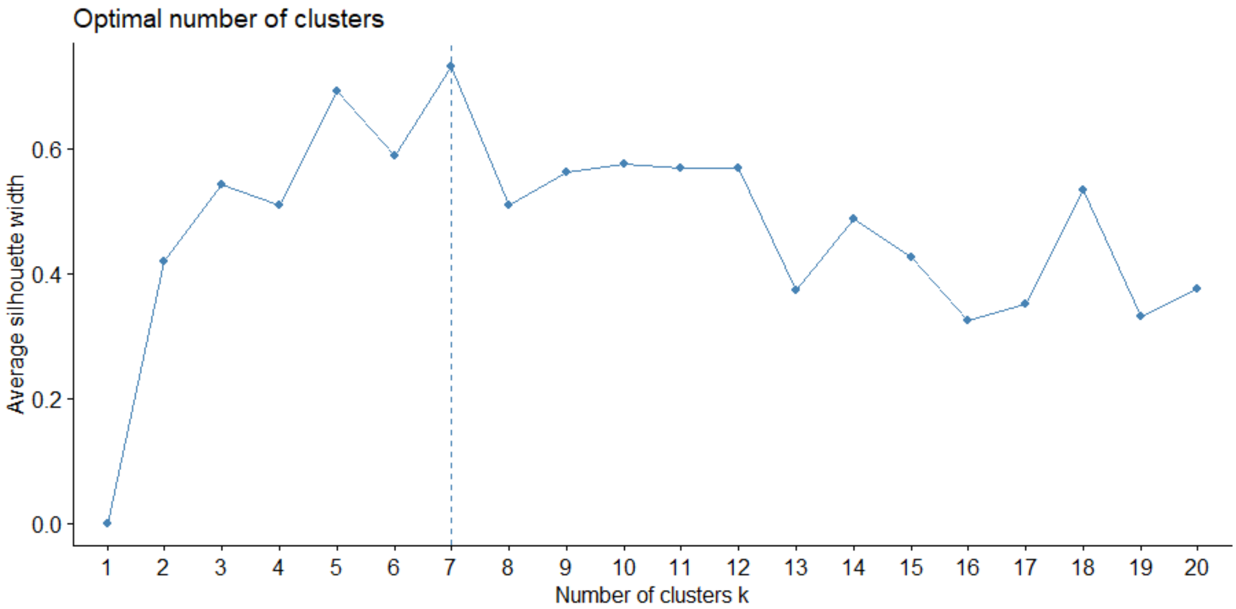
```
> # внутрішньогрупові суми квадратів
> fviz_nbclust(data, kmeans, method = "wss",k.max = 20)
```



Помітний злам при $k = 3$ і $k = 6$, причому після 6 помітна більш-менш стала поведінка. Дивно, до речі, що при збільшенні k з 4 до 5 внутрішньогрупова сума квадратів навпаки зростає…

Разом з тим побудуємо діаграму середніх силуетів.

```
> # діаграма середніх силуетів
> fviz_nbclust(data, kmeans, method = "silhouette",k.max = 20)
```



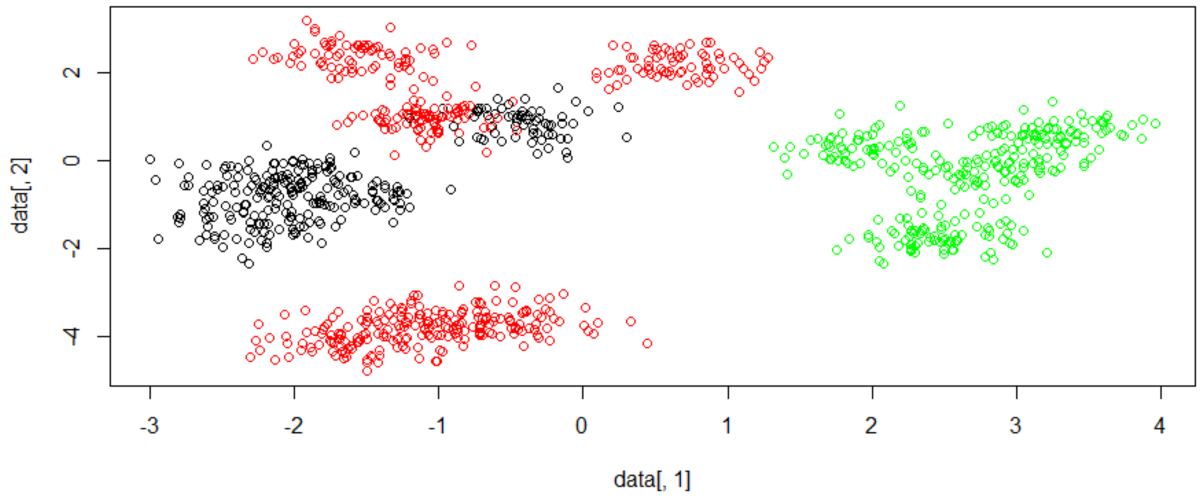
На цій діаграмі помітні «максимуми» при $k = 5, k = 7, k = 14$ і $k = 18$.

Отже, маємо розглянути шість варіантів $k - 3, 5, 6, 7, 14, 18$.

$k = 3$. Проведемо кластеризацію.

```
> #
> # k = 3
> #
> km.res3 <- kmeans(data, 3, nstart = 25)
```

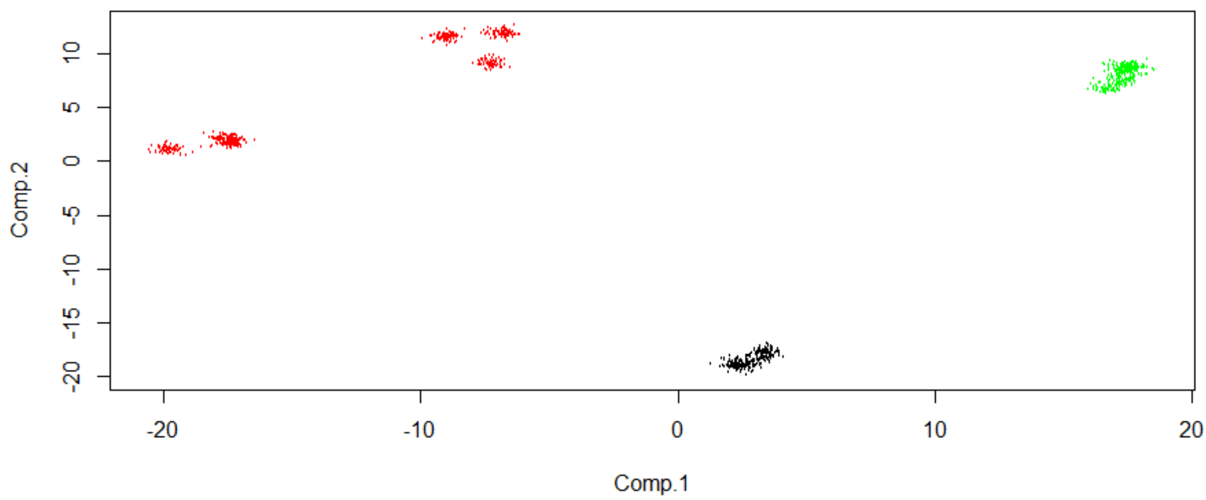
```
> km.res3$betweenss/km.res3$tot.withinss # відношення міжкластерної суми квад
ратів до внутрішньокластерної
[1] 2.168475
>
> plot(data[,1], data[,2], col = col[km.res3$cluster]) # діаграма розсіювання
перших двох змінних з кластеризацією
```



Виходячи лише з такого зображення, кластеризація не виглядає надто вдалою… Внутрішньогрупова сума квадратів є лише в 2 рази меншою за міжкластерну.

Поглянемо на діаграму розсіювання даних у просторі перших двох головних компонент.

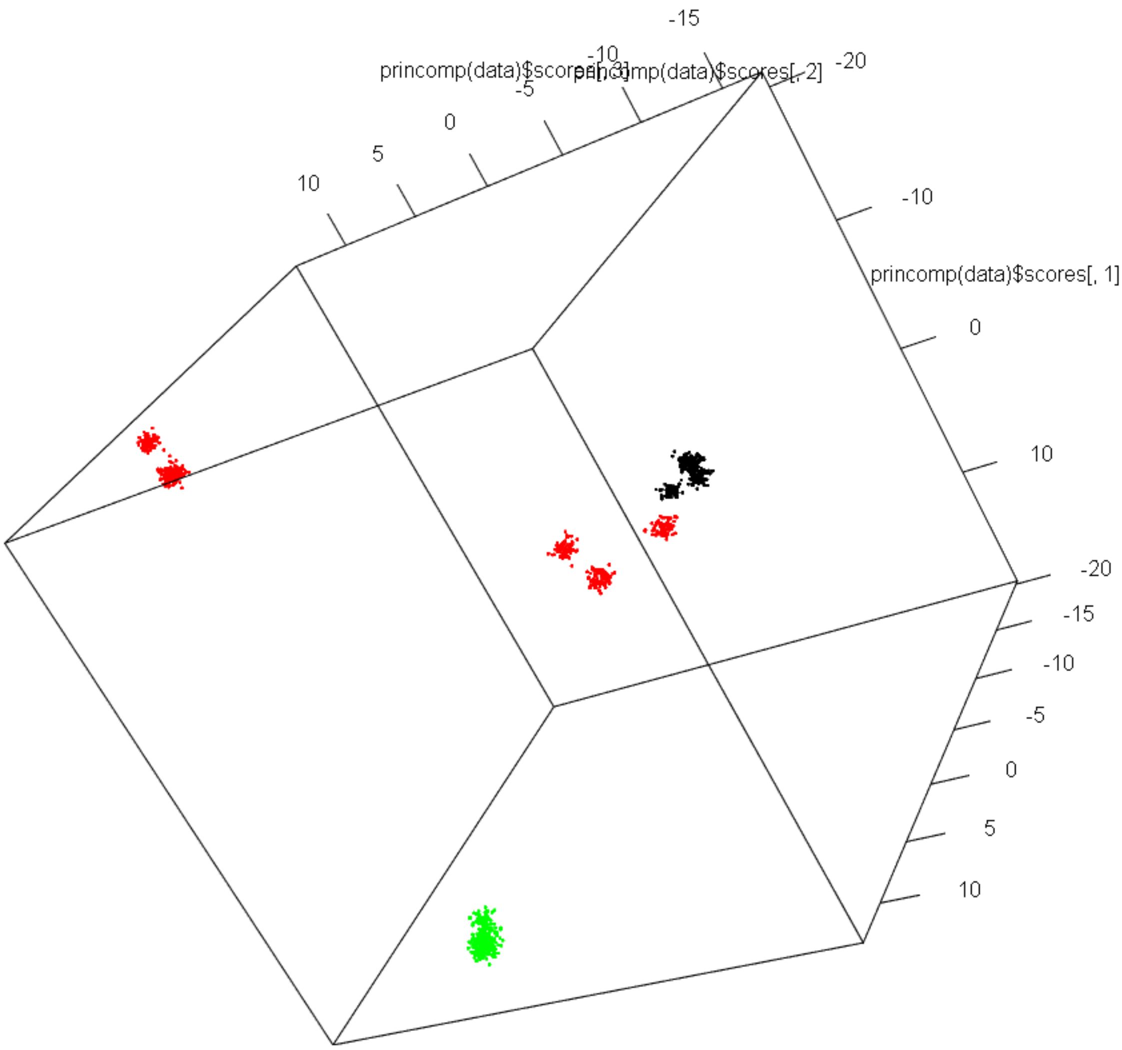
```
> # діаграма розсіювання даних у просторі перших двох головних компонент
> plot(princomp(data)$scores[,1:2],col=col[km.res3$cluster],cex=0.2)
```



На діаграмі, здається, можна виділити більшу кількість кластерів, в районі 7-9 (можливо, чорну і зелену області можна розбити на дві), що зібрані в 4 групи.

Для наглядності покрутимо цю діаграму у тривимірному просторі.

```
> plot3d(princomp(data)$scores[,1], princomp(data)$scores[,2], princomp(data)
$scores[,3], col = col[km.res3$cluster])
```

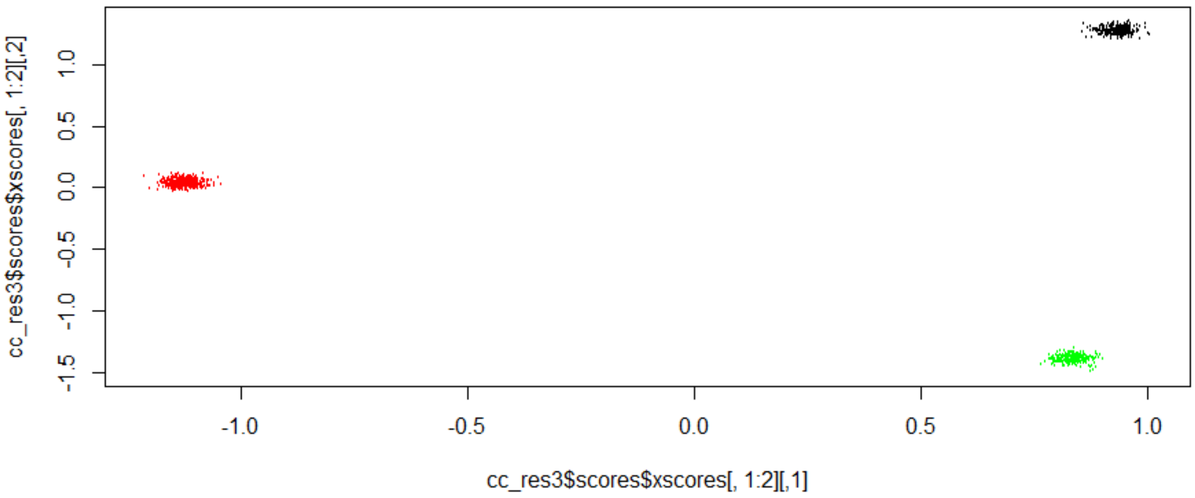


І все таки, здається тут виділяється біля 8 кластерів, які розділені в 3-4 групи.

Далі поглянемо на діаграму розсіювання у просторі канонічних компонент.

```
> require(CCA)
> cl3 <- km.res3$cluster
> k <- length(levels(as.factor(cl3)))
> n <- nrow(data)
> C <- matrix(data = as.numeric(rep(cl3, k) == rep(1:k, each = n)), ncol = k,
nrow = n)
> cc_res3 <- rcc(data,C,0.1,0.1)
```

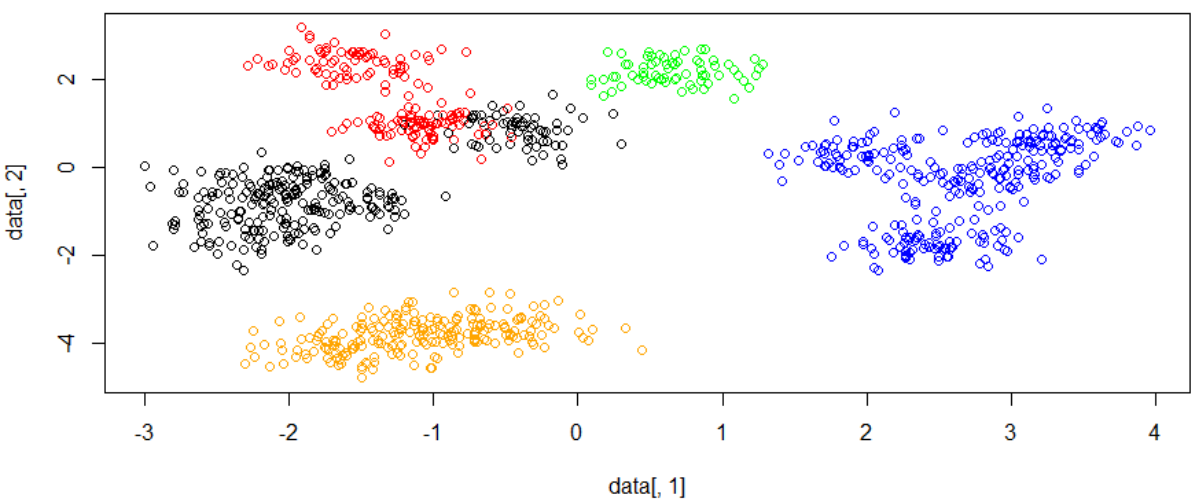
```
> # діаграма розсіювання перших двох канонічних компонент
> plot(cc_res3$scores$xscores[,1:2], col = col[cl3], cex=0.2)
```



А отже, перша канонічна компонента повністю розділяє кластери.

$k = 5$

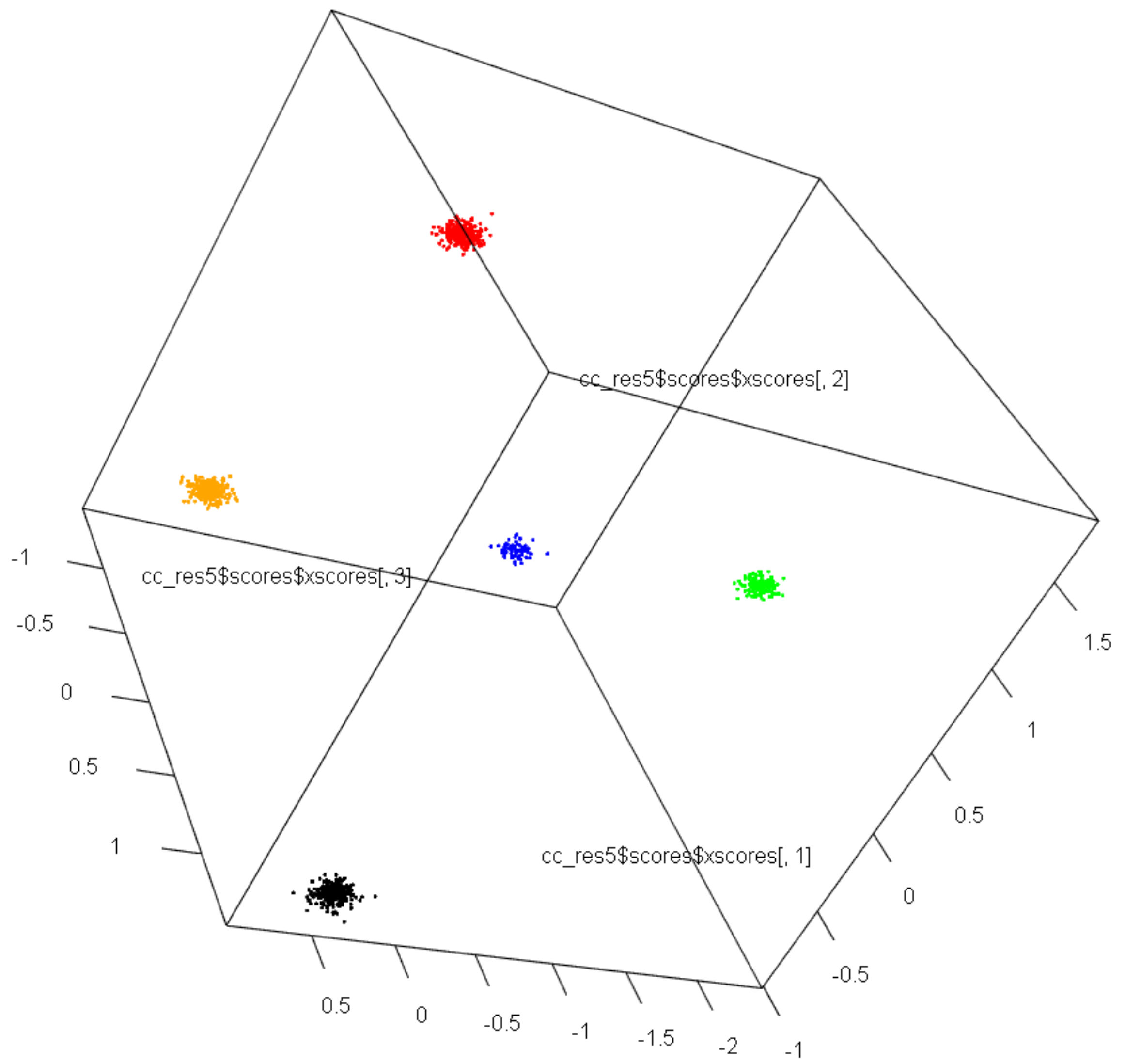
```
> #
> # k = 5
> #
> km.res5 <- kmeans(data, 5, nstart = 25)
> km.res5$betweenss/km.res5$tot.withinss # відношення міжкластерної суми квад
ратів до внутрішньокластерної
[1] 11.31561
>
> plot(data[,1], data[,2], col = col[km.res5$cluster]) # діаграма розсіювання
перших двох змінних з кластеризацією
```



Такий розділ даних, на око, виглядає дещо більш вдалим. Зобразимо діаграми розсіювання даних у просторі канонічних компонент.

```
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
```

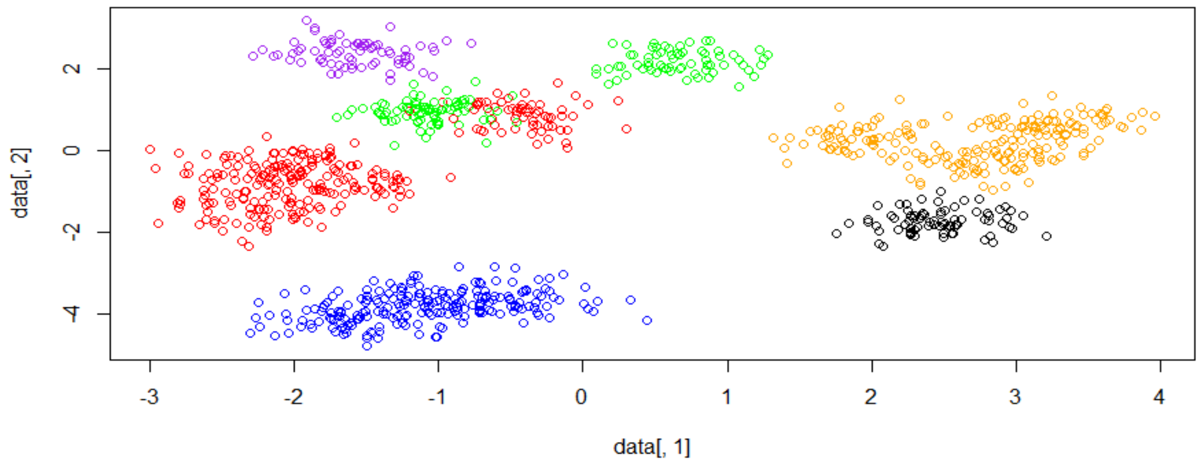
```
> cl5 <- km.res5$cluster
> k <- length(levels(as.factor(cl5)))
> n <- nrow(data)
> C <- matrix(data = as.numeric(rep(cl5, k) == rep(1:k, each = n)), ncol = k,
nrow = n)
> cc_res5 <- rcc(data,C,0.1,0.1)
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
> plot3d(cc_res5$scores$xscores[,1], cc_res5$scores$xscores[,2], cc_res5$scor
es$xscores[,3], col = col[cl5])
```



Тут чітко виділяються 5 кластерів.

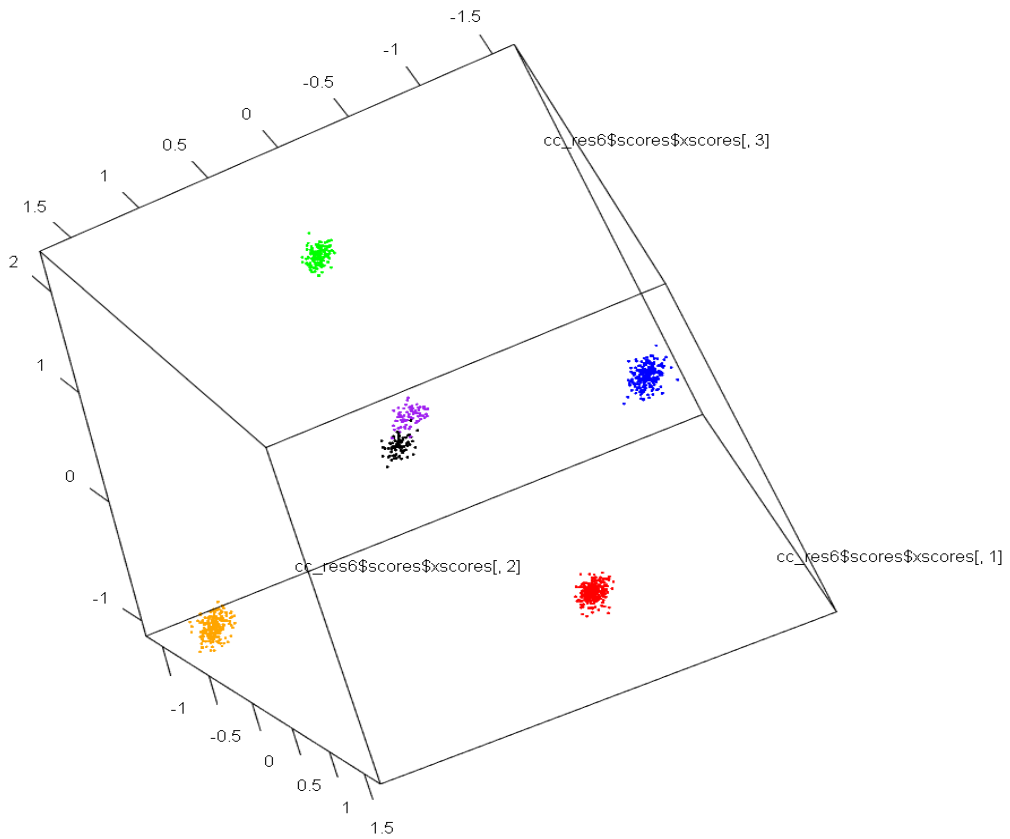
$k = 6$

```
> #
> # k = 6
> #
> km.res6 <- kmeans(data, 6, nstart = 25)
> km.res6$betweenss/km.res6$tot.withinss # відношення міжкластерної суми квад
ратів до внутрішньокластерної
[1] 13.43926
> plot(data[,1], data[,2], col = col[km.res6$cluster]) # діаграма розсіювання
перших двох змінних з кластеризацією
```



На мою думку, саме для перших двох змінних збільшення кількості клатерів до 6 не дало гарного результату: все одно є ціла купа червоних точок поміж зелених.

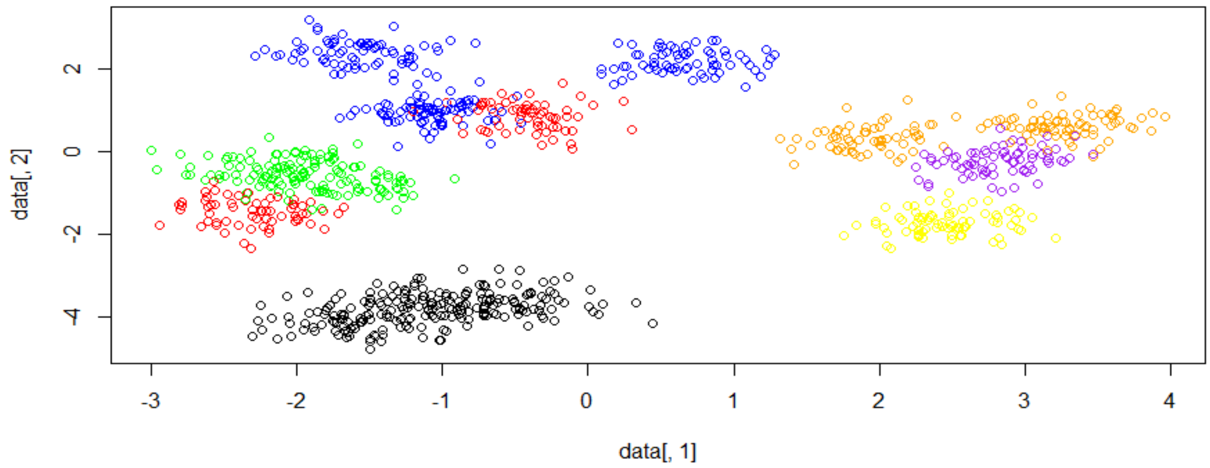
```
> cl6 <- km.res6$cluster
> k <- length(levels(as.factor(cl6)))
> n <- nrow(data)
> C <- matrix(data = as.numeric(rep(cl6, k) == rep(1:k, each = n)), ncol = k,
nrow = n)
> cc_res6 <- rcc(data,C,0.1,0.1)
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
> plot3d(cc_res6$scores$xscores[,1], cc_res6$scores$xscores[,2], cc_res6$scor
es$xscores[,3], col = col[cl6])
```



Тут можемо бачити, що два кластери (відмічені чорним і фіолетовим) розташовані заблизько один до одного, а тому шостий кластер, здається, виявився зайвим.

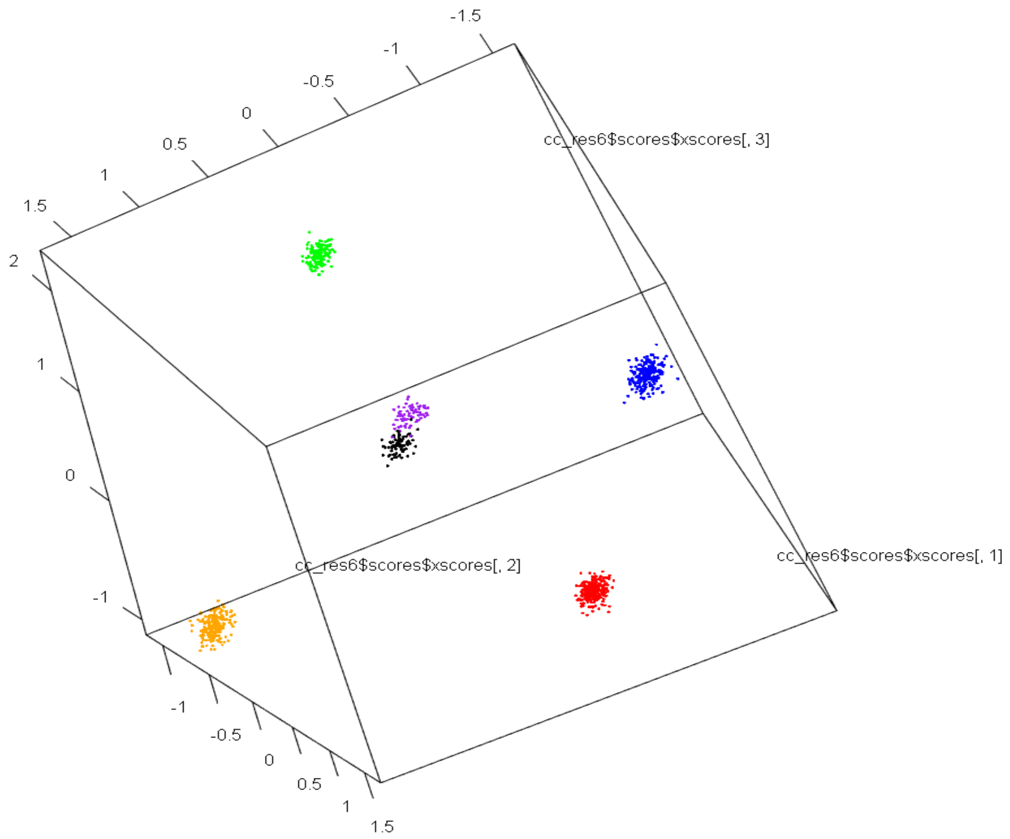
$k = 7$

```
> #
> # k = 7
> #
> km.res7 <- kmeans(data, 7, nstart = 25)
> km.res7$betweenss/km.res7$tot.withinss # відношення міжкластерної суми квад
ратів до внутрішньокластерної
[1] 14.31005
>
> plot(data[,1], data[,2], col = col[km.res7$cluster]) # діаграма розсіювання
перших двох змінних з кластеризацією
```

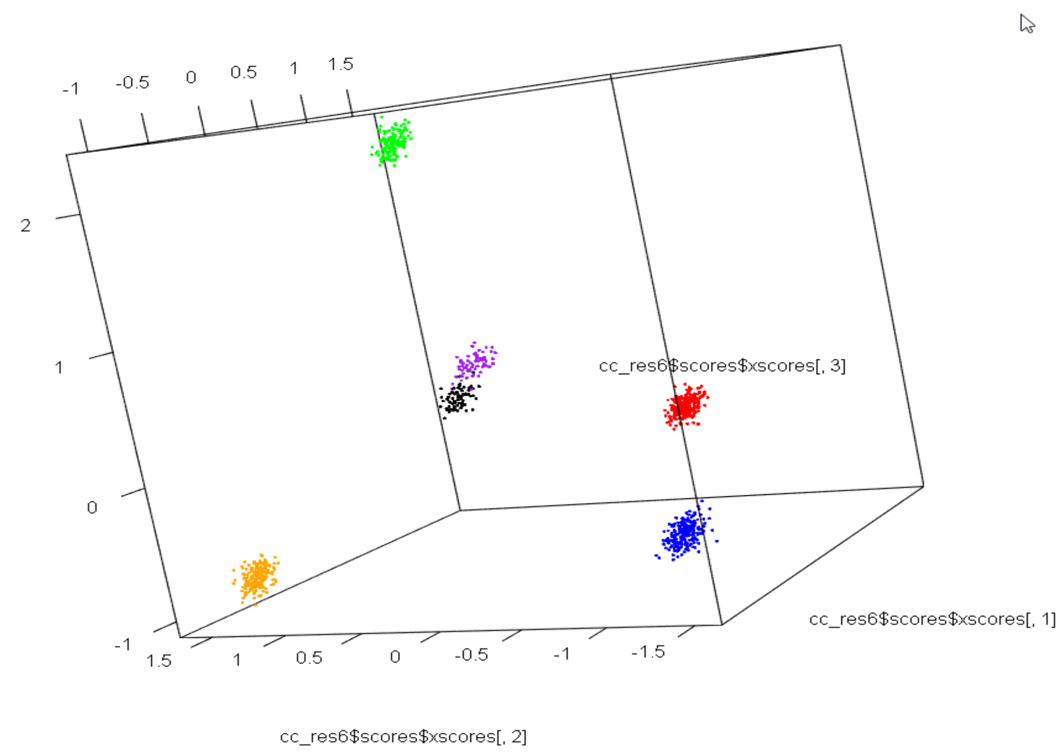


Можемо спостерігати, що кластер, точки якого позначені червоним кольором, розташовані на відстані, хоча стверджується що це один кластер.

```
> cl7 <- km.res7$cluster
> k <- length(levels(as.factor(cl7)))
> n <- nrow(data)
> C <- matrix(data = as.numeric(rep(cl7, k) == rep(1:k, each = n)), ncol = k,
nrow = n)
> cc_res7 <- rcc(data,C,0.1,0.1)
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
> plot3d(cc_res6$scores$xscores[,1], cc_res6$scores$xscores[,2], cc_res6$scor
es$xscores[,3], col = col[cl6])
```



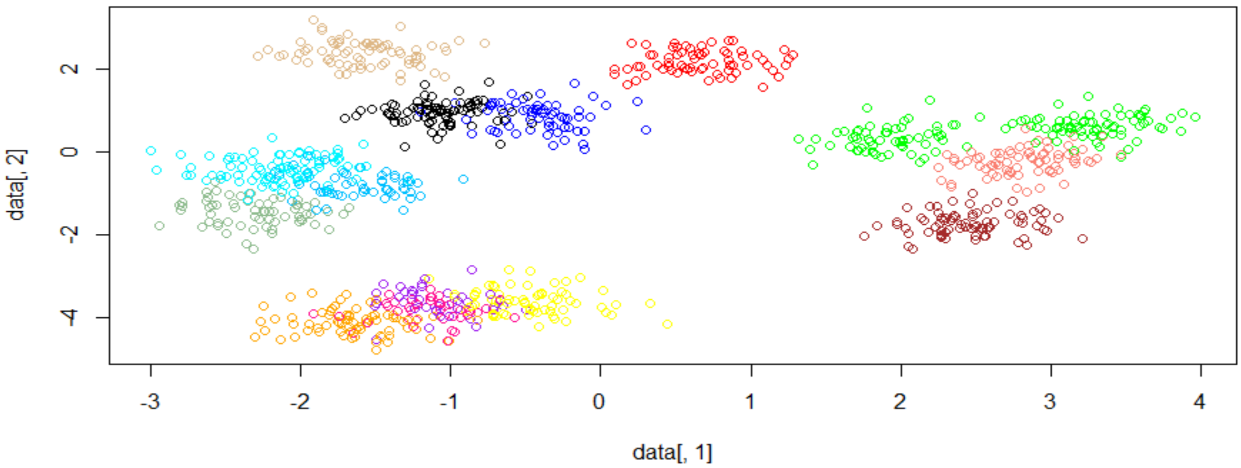
Тут можемо бачити, що два кластери (відмічені чорним і фіолетовим) розташовані заблизько один до одного, а тому шостий кластер, здається, виявився зайвим.



Докорінно тут ситуація не змінилася: все одно маємо купку, в якій знаходяться два кластери.

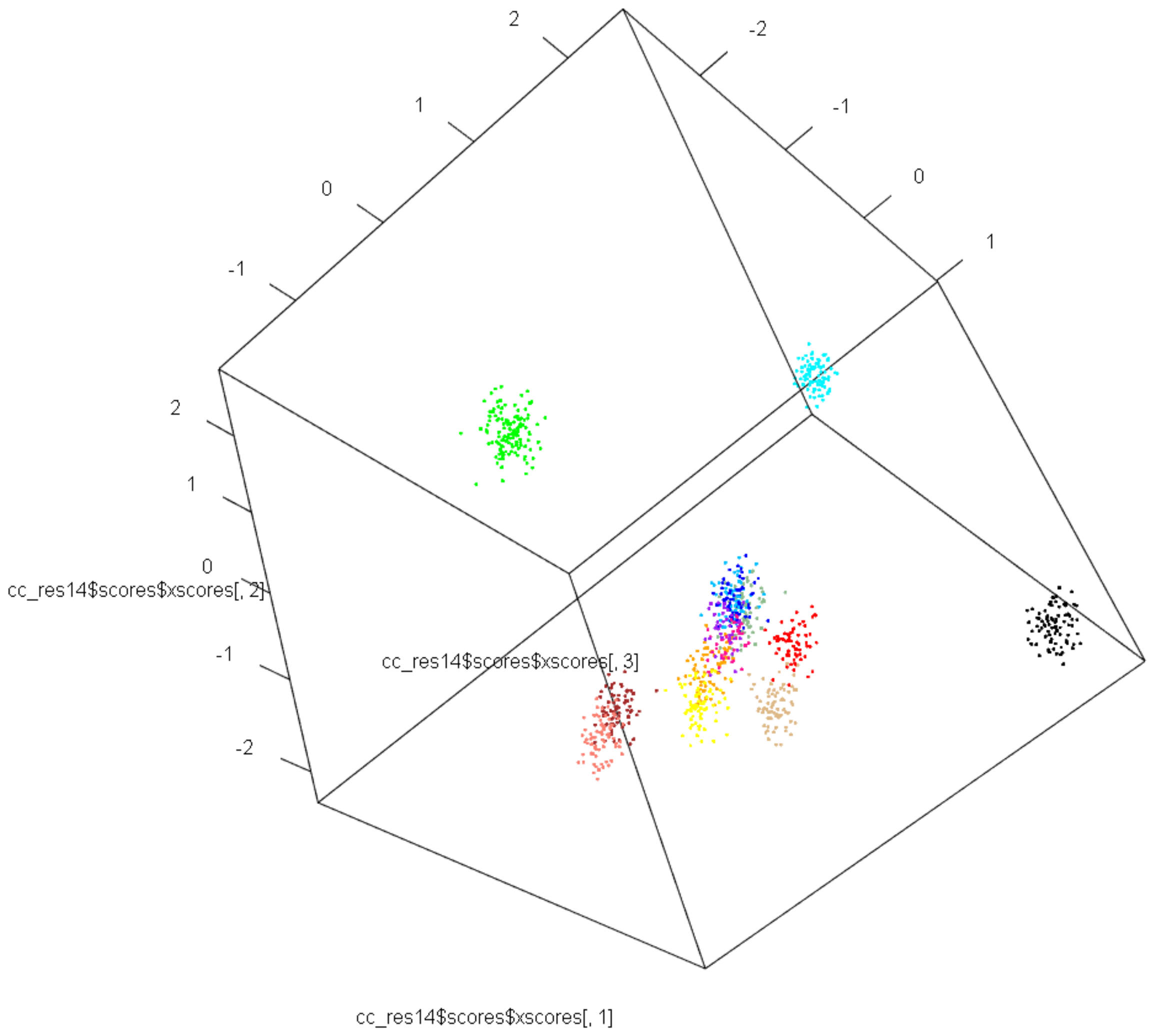
$k = 14$

```
> #
> # k = 14
> #
> km.res14 <- kmeans(data, 14, nstart = 25)
> km.res14$betweenss/km.res14$tot.withinss # відношення міжкластерної суми кв
адратів до внутрішньокластерної
[1] 57.9986
> plot(data[,1], data[,2], col = col[km.res14$cluster]) # діаграма розсіюванн
я перших двох змінних з кластеризацією
```



Здається, непогана кластеризація, проте, приглядівшись, можемо спостерігати, що знизу поміж фіолетових, жовтих та коричневих точок є червоні, які в той же самий час присутні і зверху.

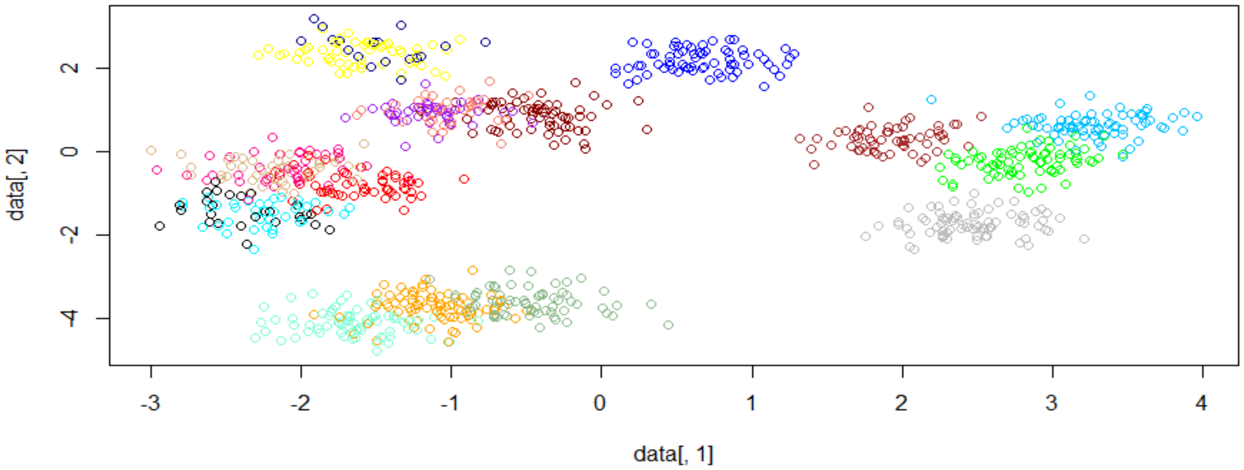
```
> cl14 <- km.res14$cluster
> k <- length(levels(as.factor(cl14)))
> C <- matrix(data = as.numeric(rep(cl14, k) == rep(1:k, each = n)), ncol = k
, nrow = n)
> cc_res14 <- rcc(data,C,0.1,0.1)
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
> plot3d(cc_res14$scores$xscores[,1], cc_res14$scores$xscores[,2], cc_res14$s
cores$xscores[,3], col = col[cl14])
```



Тут чітко «по краям» виділяються три кластери, а от решта – постійно групуються, утворюючи «різнокольорові» плями.

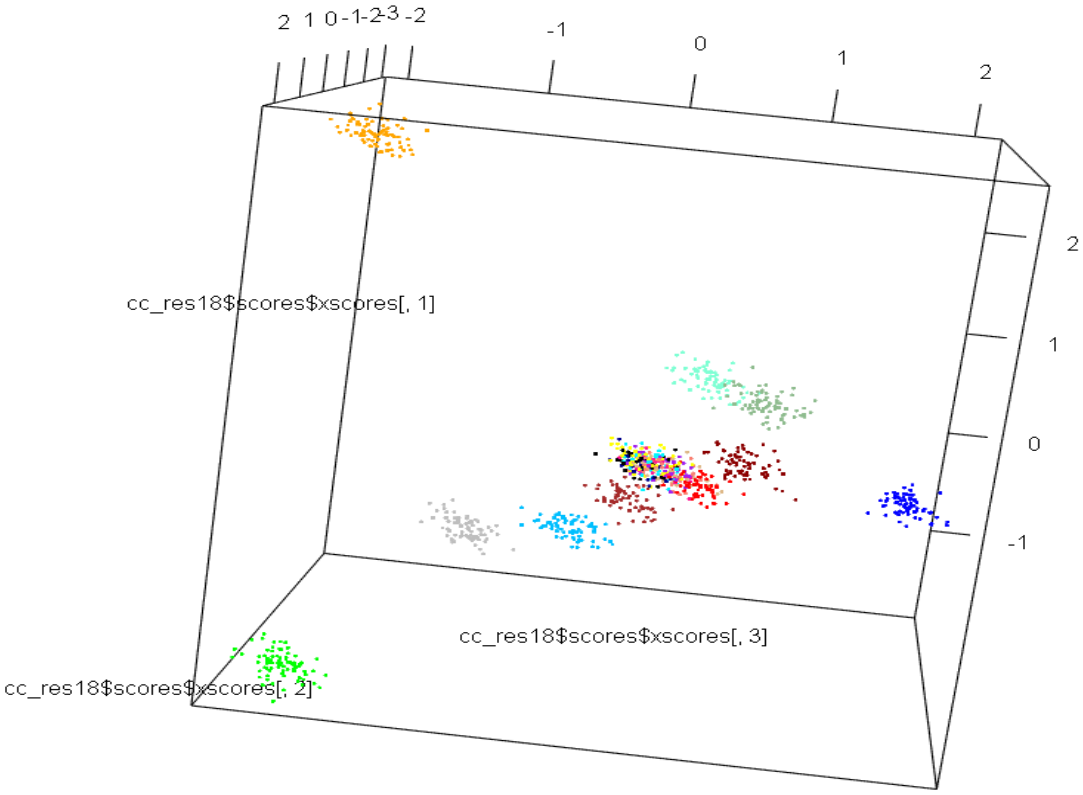
$k = 18$

```
> #
> # k = 18
> #
> km.res18 <- kmeans(data, 18, nstart = 25)
> km.res18$betweenss/km.res18$tot.withinss # відношення міжкластерної суми кв
адратів до внутрішньокластерної
[1] 99.63196
>
> plot(data[,1], data[,2], col = col[km.res18$cluster]) # діаграма розсіюванн
я перших двох змінних з кластеризацією
```



Аналогічно: можемо бачити, як, наприклад, червоні точки розподіляються одразу серед і жовтих, і коричневих, і аквамаринових, і червоних. Здається, не дуже вдало…

```
> cl18 <- km.res18$cluster
> k <- length(levels(as.factor(cl18)))
> C <- matrix(data = as.numeric(rep(cl18, k) == rep(1:k, each = n)), ncol = k
, nrow = n)
> cc_res18 <- rcc(data,C,0.1,0.1)
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
> plot3d(cc_res18$scores$xscores[,1], cc_res18$scores$xscores[,2], cc_res18$s
cores$xscores[,3], col = col[cl18])
```



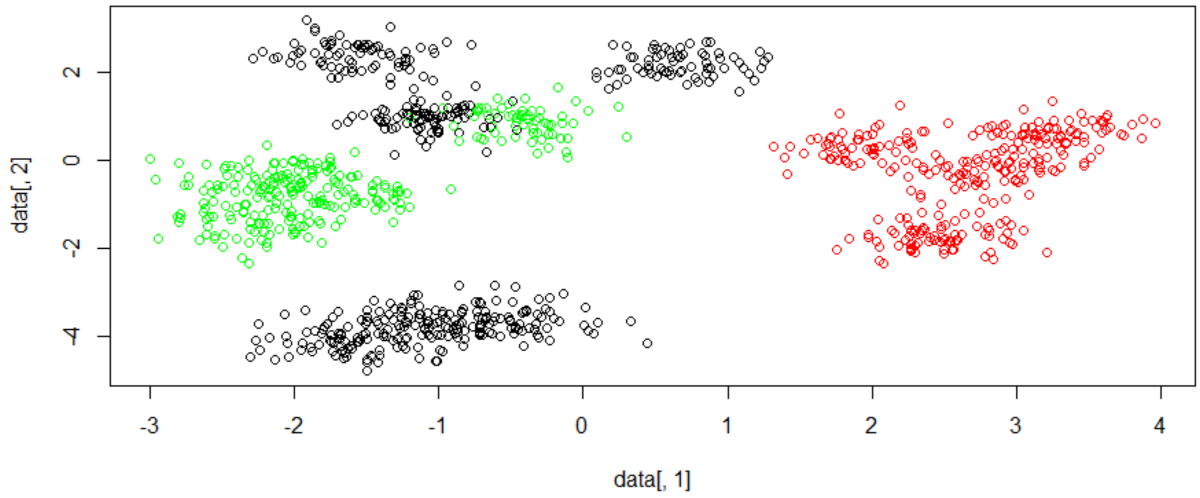
Аналогічно маємо 4 купки по краям – але повний різнокольоровий безлад всередині.

Отож, на методі центроїдів я би зупинився на такій кількості кластерів, як-от 3 або 5.

Далі застосуємо метод медоїдів. Повернувшись до графіку середніх силуетів, приходимо до висновку необхідності розгляду таких k, як $3, 5, 7$.

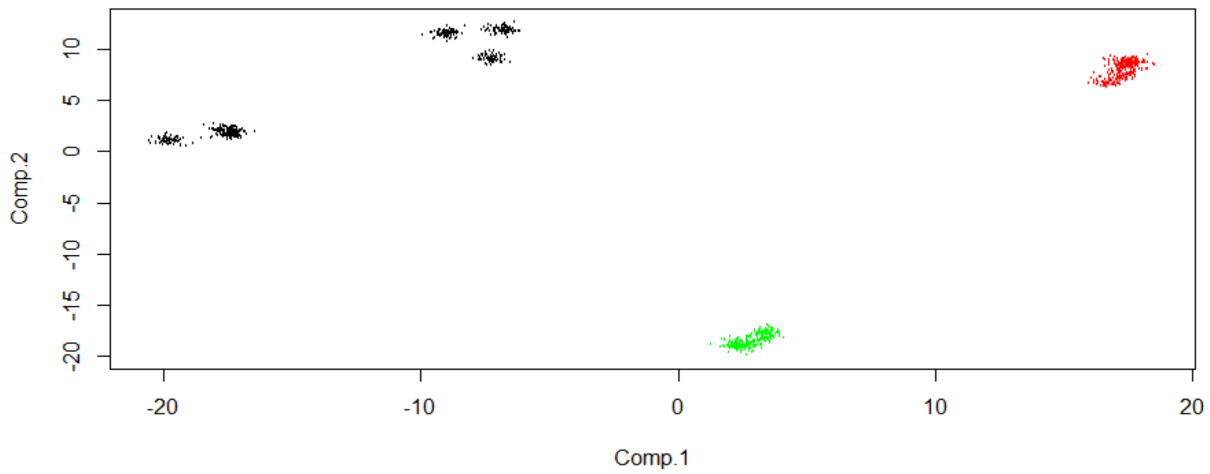
$k = 3$

```
> library(cluster) # підключаємо бібліотеки
> pam.res3 <- pam(data, 3)
> plot(data[,1], data[,2], col = col[pam.res3$cluster]) # діаграма розсіюванн
я перших двох змінних з кластеризацією
```



Далі розглянемо діаграму розсіювання даних у просторі перших двох головних компонент.

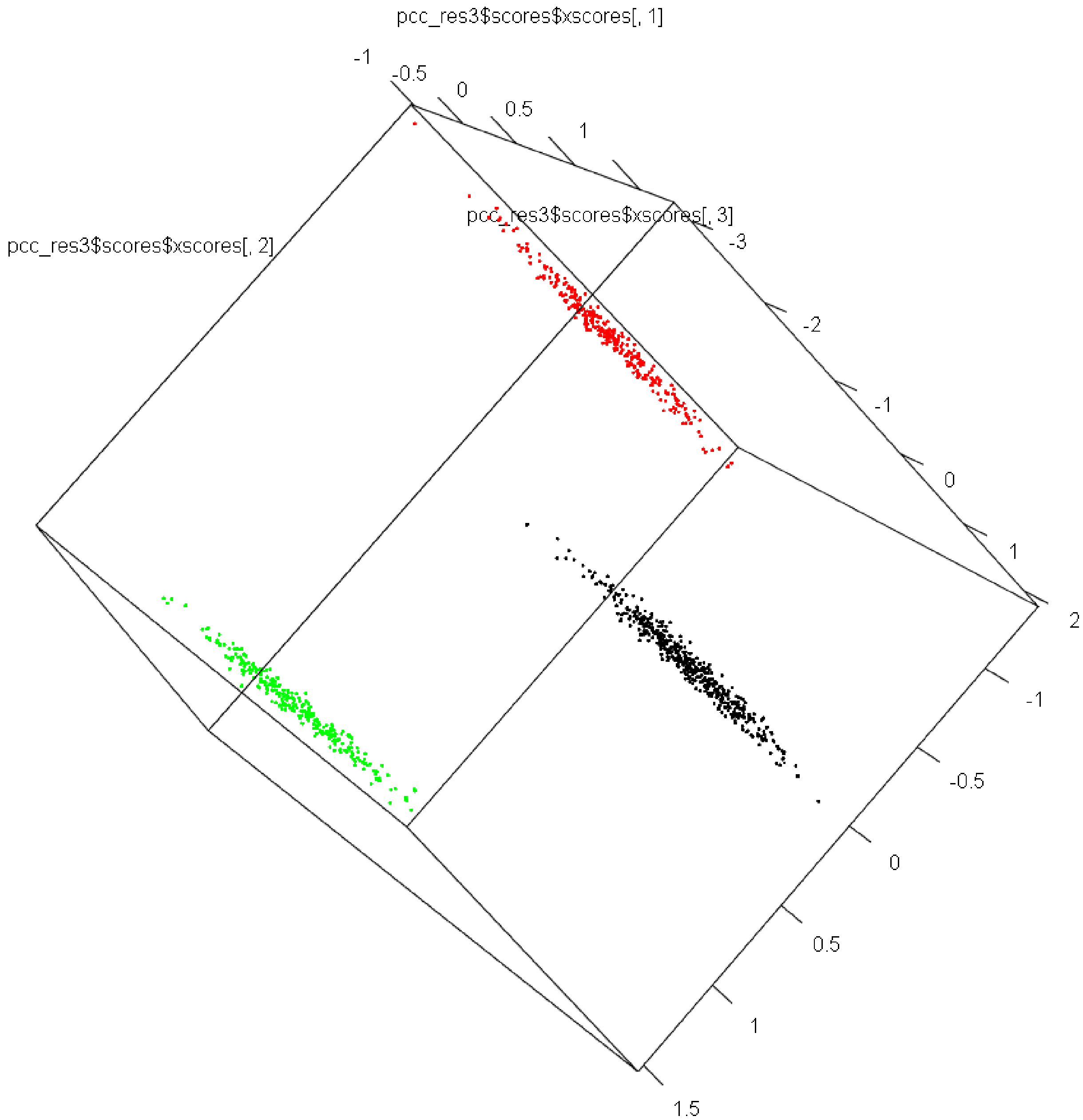
```
> # діаграма розсіювання даних у просторі перших двох головних компонент
> plot(princomp(data)$scores[,1:2],col=col[pam.res3$cluster],cex=0.2)
```



Виглядає цілком логічно, хоча б чорні купки можна було розбити на два кластери.

Але поглянемо на тривимірну діаграму розсіювання даних у просторі перших трьох канонічних компонент.

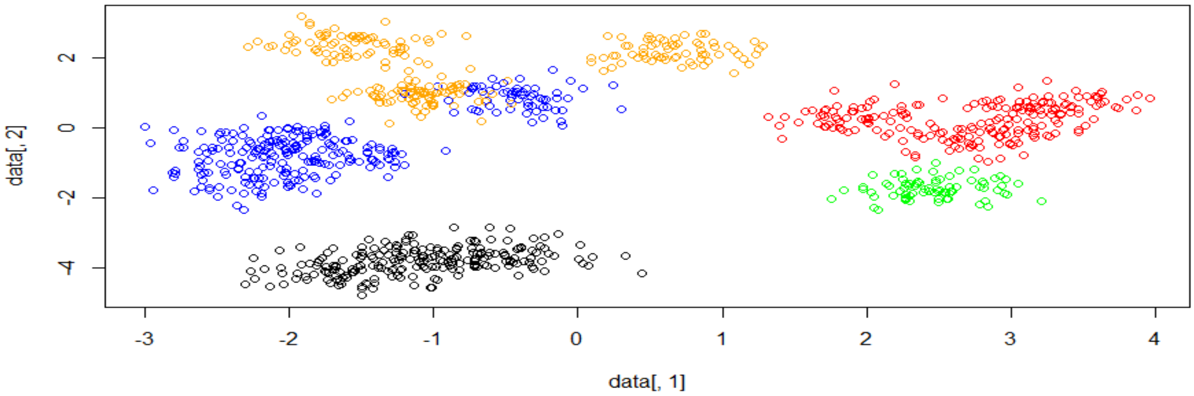
```
> pcl3 <- pam.res3$cluster
> k <- length(levels(as.factor(pcl3)))
> n <- nrow(data)
> C <- matrix(data = as.numeric(rep(pcl3, k) == rep(1:k, each = n)), ncol = k
, nrow = n)
> pcc_res3 <- rcc(data,C,0.1,0.1)
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
> plot3d(pcc_res3$scores$xscores[,1], pcc_res3$scores$xscores[,2], pcc_res3$s
cores$xscores[,3], col = col[pcl3])
```



Чітко виділяються три групи спостережень, а отож кластеризація на три кластери є слушною.

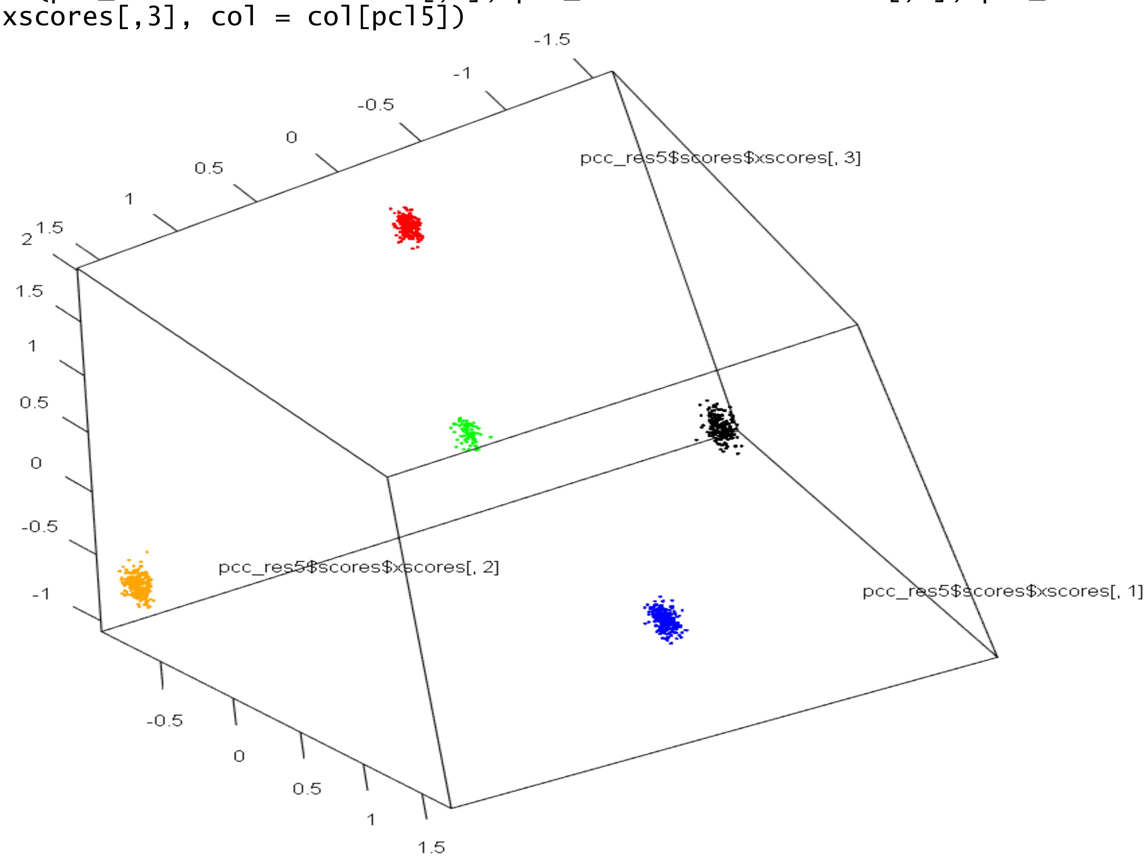
$k = 5$

```
> # k = 5
> pam.res5 <- pam(data, 5)
>
> plot(data[,1], data[,2], col = col[pam.res5$cluster]) # діаграма розсіювання першиx двох змінних з кластеризацією
```



Поглянемо далі на тривимірну діаграму розсіювання даних у просторі перших трьох канонічних компонент.

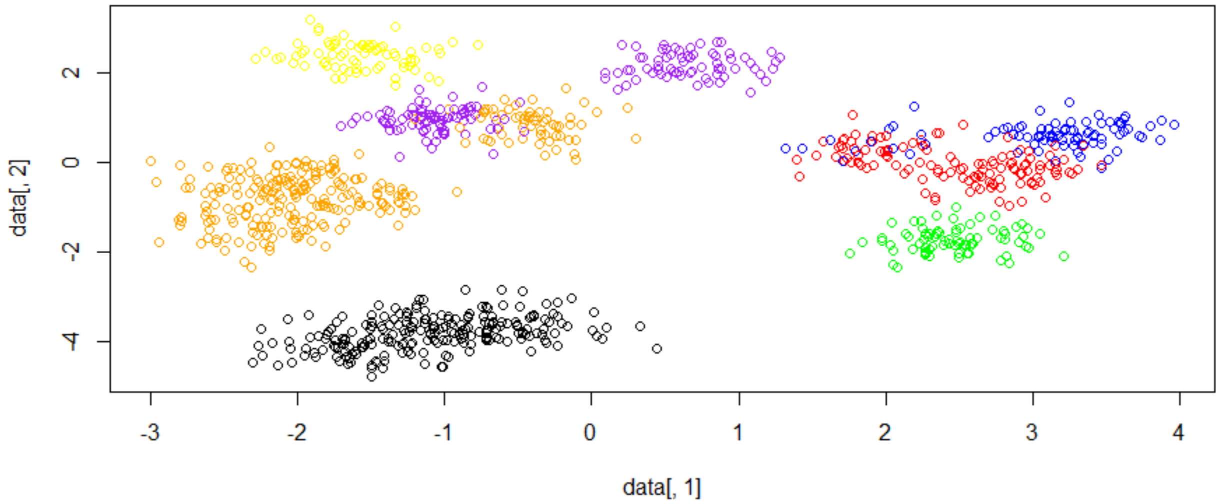
```
> pcl5 <- pam.res5$cluster
> k <- length(levels(as.factor(pcl5)))
> C <- matrix(data = as.numeric(rep(pcl5, k) == rep(1:k, each = n)), ncol = k
, nrow = n)
> pcc_res5 <- rcc(data,C,0.1,0.1)
>
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
> plot3d(pcc_res5$scores$xscores[,1], pcc_res5$scores$xscores[,2], pcc_res5$s
cores$xscores[,3], col = col[pcl5])
```



Чітко виділяються п'ять віддалених купок, а отже поділ на 5 кластерів також є цілком допустимим.

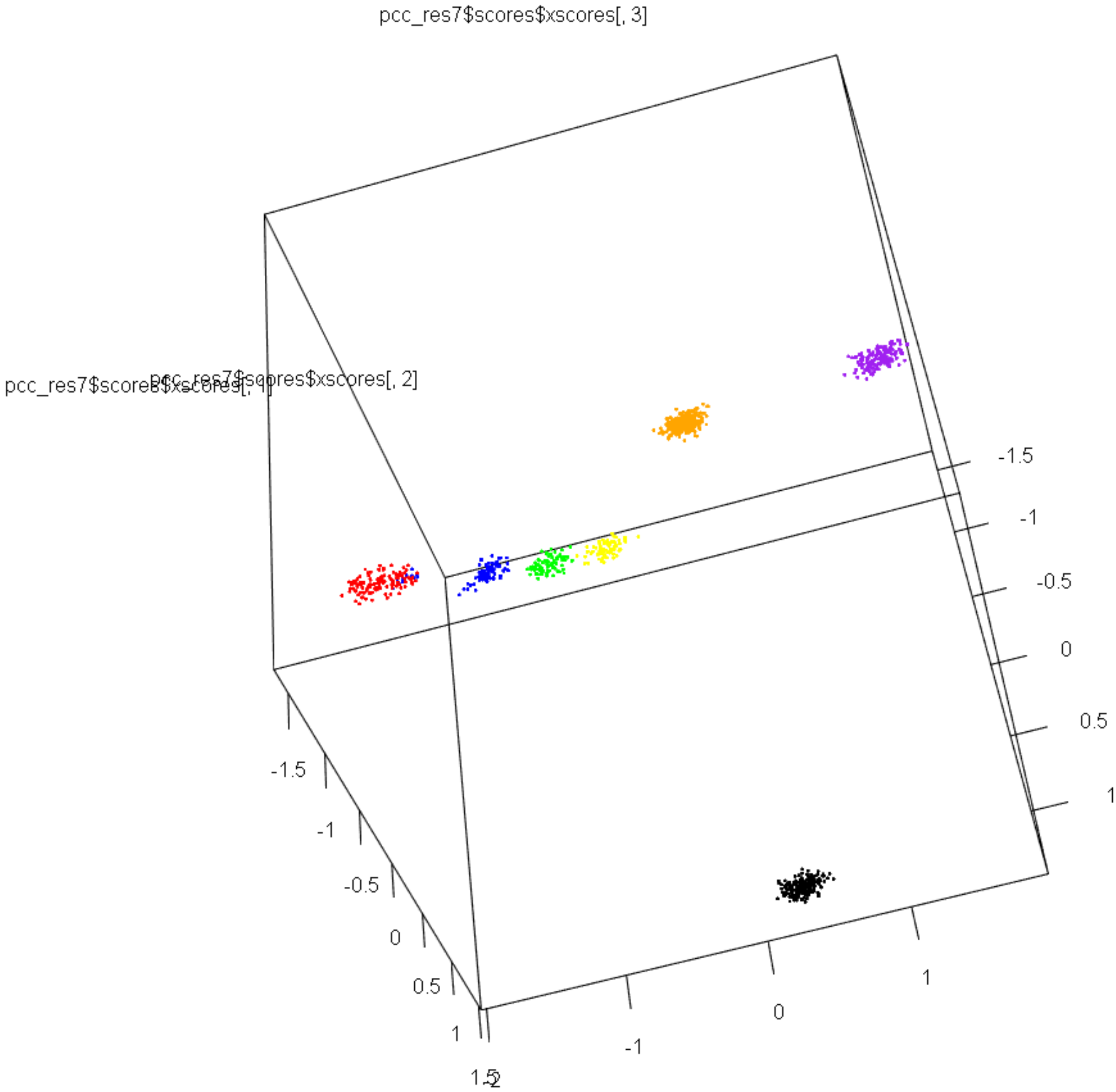
$k = 7$

```
> # k = 7
>
> pam.res7 <- pam(data, 7)
> plot(data[,1], data[,2], col = col[pam.res7$cluster]) # діаграма розсіюванн
я перших двох змінних з кластеризацією
```



Поглянемо далі на тривимірну діаграму розсіювання даних у просторі перших трьох канонічних компонент.

```
> pcl7 <- pam.res7$cluster
> k <- length(levels(as.factor(pcl7)))
> C <- matrix(data = as.numeric(rep(pcl7, k) == rep(1:k, each = n)), ncol = k
, nrow = n)
> pcc_res7 <- rcc(data,C,0.1,0.1)
> # тривимірна діаграма розсіювання перших трьох канонічних компонент
> plot3d(pcc_res7$scores$xscores[,1], pcc_res7$scores$xscores[,2], pcc_res7$s
cores$xscores[,3], col = col[pcl7])
```



Тут картина є дещо гіршою: незважаючи на чітко віддалені групи спостережень, в той самий час маємо ситуацію, коли частина синіх точок знаходяться всередині червоної групки. До того ж сильно щільно розташовуються решта групок (зелена, жовта, синя і червона).

Отож, в якості фінальних претендентів на кількість кластерів, обидва методи мені дали такі числа, як 3 і 5.

Порівняємо, наскільки різними виявились кластеризації в цих випадках для методів центроїдів і медоїдів. Для цього застосуємо індекс Ренда.

```
> rand.index(km.res3$cluster, pam.res3$clustering)
[1] 1
```

Отже, обидва методи кластеризації для $k = 3$ дали абсолютно однаковий результат.

```
> rand.index(km.res5$cluster, pam.res5$clustering)
[1] 0.9475616
```

Що ж до $k = 5$, маємо певні відмінності. Розглянемо таблицю спряженості:

```
> library(MASS)
> table(pam.res5$clustering,km.res5$cluster)

      1   2   3   4   5
  1   0   0   0   0 220
  2   0   0   0 208   0
  3   0   0   0  76   0
  4 274   0   0   0   0
  5   0 155  67   0   0
```

Отже, в цілому обидва методи більшість спостережень закидали по однаковим групам, проте частину спостережень різні методи віднесли до різних груп.