

Contents

| | | |
|----------|---|-----------|
| 1 | Wstęp | 4 |
| 1.1 | Testowane algorytmy | 4 |
| 1.2 | Ocena jakości algorytmów | 4 |
| 1.3 | Zawartość poszczególnych folderów | 4 |
| 1.4 | Uwagi | 4 |
| 2 | Funkcja testująca | 5 |
| 2.1 | Potrzebne pakiety | 5 |
| 2.2 | Opis funkcji | 8 |
| 2.3 | Przykładowe wywołania funkcji | 8 |
| 3 | Algorytm spektralny | 10 |
| 3.1 | Implementacja algorytmu | 10 |
| 3.1.1 | Funkcja <i>Mnn</i> | 10 |
| 3.1.2 | Funkcja <i>Mnn_graph</i> | 10 |
| 3.1.3 | Funkcja <i>Laplacian_eigen</i> | 11 |
| 3.1.4 | Funkcja <i>spectral_clustering</i> | 12 |
| 3.2 | Badanie algorytmu na wszystkich zbiorach benchmarkowych | 12 |
| 3.2.1 | Bez standaryzacji zmiennych | 12 |
| 3.2.2 | Ze standaryzacją zmiennych | 16 |
| 4 | Algorytm <i>hclust</i> | 19 |
| 4.1 | Badanie algorytmu na wszystkich zbiorach benchmarkowych | 19 |
| 4.1.1 | Bez standaryzacji zmiennych | 19 |
| 4.1.2 | Ze standaryzacją zmiennych | 22 |
| 5 | Algorytm <i>Genie</i> z pakietu <i>genie</i> | 25 |
| 5.1 | Badanie algorytmu na wszystkich zbiorach benchmarkowych | 25 |
| 5.1.1 | Bez standaryzacji zmiennych | 25 |
| 5.1.2 | Ze standaryzacją zmiennych | 28 |
| 6 | Algorytm <i>HCPC</i> | 31 |
| 6.1 | Badanie algorytmu na wszystkich zbiorach benchmarkowych | 31 |
| 6.1.1 | Bez standaryzacji zmiennych | 31 |
| 6.1.2 | Ze standaryzacją zmiennych | 33 |
| 7 | Algorytm <i>cmeans</i> z pakietu <i>e1071</i> | 35 |
| 7.1 | Badanie algorytmu na wszystkich zbiorach benchmarkowych | 35 |
| 7.1.1 | Bez standaryzacji zmiennych | 35 |
| 7.1.2 | Ze standaryzacją zmiennych | 38 |

| | | |
|----------|-----------------------------|-----------|
| 8 | Podsumowanie | 41 |
| 8.1 | <i>genie</i> | 41 |
| 8.1.1 | Bez standaryzacji zmiennych | 41 |
| 8.1.2 | Ze standaryzacją zmiennych | 42 |
| 8.2 | <i>hclust</i> | 42 |
| 8.2.1 | Bez standaryzacji zmiennych | 42 |
| 8.2.2 | Ze standaryzacją zmiennych | 43 |
| 8.3 | <i>spectral_clustering</i> | 43 |
| 8.3.1 | Bez standaryzacją zmiennych | 43 |
| 8.3.2 | Ze standaryzacją zmiennych | 44 |
| 8.4 | <i>HCPC</i> | 44 |
| 8.4.1 | Bez standaryzacji zmiennych | 44 |
| 8.4.2 | Ze standaryzacją zmiennych | 44 |
| 8.5 | <i>cmeans</i> | 45 |
| 8.5.1 | Bez standaryzacji zmiennych | 45 |
| 8.5.2 | Ze standaryzacją zmiennych | 46 |

1 Wstęp

Poniższy dokument przedstawia analizę danych benchmarkowych przy wykorzystaniu pięciu algorytmów analizy skupień (*spectral clustering*).

1.1 Testowane algorytmy

Przetestowane zostały następujące algorytmy analizy skupień:

- własna implementacja algorytmu spektralnego
- algorytmy hierarchiczne z funkcji *hclust()*
- algorytm *Genie* z pakietu **genie**
- *HCPC* - Hierarchical Clustering on Principal Components
- *cmeans*

1.2 Ocena jakości algorytmów

Do oceny podobieństwa dwóch k -podziałów zostały użyte następujące indeksy:

- skorygowany indeks Randa (**AR**) - indeks ten pozwala ocenić zgodność dwóch podziałów na zbioru na rozłączne podzbiory.
- indeks Fowlkesa-Mallowsa (**MS**)

1.3 Zawartość poszczególnych folderów

- *benchmarkPlots* - wykresy danych zbiorów benchmarkowych
- *benchmarkResults* - pliki *.csv* zawierające wyniki poszczególnych algorytmów
 - pliki **STAND.csv* - zawierają wyniki poszczególnych algorytmów przy uwzględnieniu standaryzacji zmiennych (kolumn w **X**)
 - pozostałe pliki - zawierają wyniki poszczególnych algorytmów bez uwzględnienia standaryzacji zmiennych (kolumn w **X**)
- *myBenchmark* - pliki *.data* oraz *.labels0* stworzonych przeze mnie zbiorów benchmarkowych
- *pd2-zbiory-benchmarkowe* - ściąganie ze strony zbiory benchmarkowe
- *scripts* - skrypty wykorzystywane do tworzenia wyników
- *tasks* - treść pracy domowej

1.4 Uwagi

Uwaga 1: W sprawozdaniu nie dodawałem wykresów przedstawiających poszczególne zbiory danych (uznałem, iż nie jest to konieczne). Natomiast w folderze *benchmarkPlots* są przedstawiane dane zbiory o nazwach w kolejności wystąpienia.

2 Funkcja testująca

Poniższy kod testujący powstał w celu zautomatyzowania badania poszczególnych algorytmów oraz ich zapis do plików *.csv*. Przedstawiony poniżej kod można też znaleźć w pliku *raport.R* (jest on zamieszczony wraz z przykładowymi wywołaniami). Opis funkcji znajduje się pod listingiem kodu.

2.1 Potrzebne pakiety

```
library("mclust")
library("genie")
library("FactoMineR")
library("e1071")
source("spectral.R")

menu <- function(choseAlgorithm="spectral_clustering", stand=FALSE, params=NULL){

  path <- getwd()
  path <- normalizePath("..")
  dirPath <- file.path(path, "pd2-zbiory-benchmarkowe")
  data <- list.files(dirPath, "data\\*.gz$", recursive=TRUE)
  dataLabels <- list.files(dirPath, "labels0\\*.gz$", recursive=TRUE)

  # deciding how many digits after period
  decimalPlaces <- 3

  if (choseAlgorithm == "spectral_clustering"){
    # check if all params all natural
    stopifnot(all(is.natural.number(params)))

    inFM <- data.frame(matrix(ncol = length(params), nrow = length(data)))
    inAM <- data.frame(matrix(ncol = length(params), nrow = length(data)))
  }

  if (choseAlgorithm == "hclust"){
    methodList = list("ward.D", "ward.D2", "single", "complete",
                      "average", "mcquitty", "median", "centroid")

    inFM <- data.frame(matrix(ncol = length(methodList), nrow = length(data)))
    inAM <- data.frame(matrix(ncol = length(methodList), nrow = length(data)))
  }

  if (choseAlgorithm == "genie"){
    stopifnot(all( (params > 0) & (params < 1)))

    inFM <- data.frame(matrix(ncol = length(params), nrow = length(data)))
    inAM <- data.frame(matrix(ncol = length(params), nrow = length(data)))
  }

  if (choseAlgorithm == "HCPC"){
    inFM <- data.frame(matrix(ncol = length(params), nrow = length(data)))
    inAM <- data.frame(matrix(ncol = length(params), nrow = length(data)))
  }
}
```

```

}

if (choseAlgorithm == "cmeans"){
  inFM <- data.frame(matrix(ncol = length(params), nrow = length(data)))
  inAM <- data.frame(matrix(ncol = length(params), nrow = length(data)))
}

for (iter in 1:length(data)){

  X <- read.table(file.path(dirPath, data[iter]), header = FALSE,
                  sep = "", dec = ".")
  labels <- read.table(file.path(dirPath, dataLabels[iter]), header = FALSE,
                      sep = "", dec = ".")

  if(stand == TRUE){
    X <- as.data.frame(scale(X))
  }

  if (choseAlgorithm == "spectral_clustering"){
    for (iterParams in 1:length(params)){

      result <- spectral_clustering(X, length(unique(labels$V1)), params[iterParams])

      calculatedLabel <- result

      inFM[iter, iterParams] <- round(as.numeric(FM_index(labels$V1, calculatedLabel)),
                                    decimalPlaces)
      inAM[iter, iterParams] <- round(as.numeric(mclust::adjustedRandIndex(labels$V1,
                                    calculatedLabel)), decimalPlaces)

      # naming columns
      colnames(inFM)[iterParams] <- paste("FM M", params[iterParams], sep = " ")
      colnames(inAM)[iterParams] <- paste("AM M", params[iterParams], sep = " ")
    }
  }

  if (choseAlgorithm == "hclust"){
    for (iterMethod in 1:length(methodList)){

      result <- hclust(dist(X, method = "euclidean"), method = methodList[iterMethod])
      calculatedLabel <- cutree(result, length(unique(labels$V1)))

      # we round our result to two decimal places
      inFM[iter, iterMethod] <- round(as.numeric(FM_index(labels$V1, calculatedLabel)),
                                    decimalPlaces)
      inAM[iter, iterMethod] <- round(as.numeric(mclust::adjustedRandIndex(labels$V1,
                                    calculatedLabel)), decimalPlaces)

      # naming columns
      colnames(inFM)[iterMethod] <- paste("FM", methodList[iterMethod], sep = " ")
      colnames(inAM)[iterMethod] <- paste("AM", methodList[iterMethod], sep = " ")
    }
  }
}

```

```

    }
  }

  if (choseAlgorithm == "genie"){
    for (iterParams in 1:length(params)){

      result <- genie::hclust2(object = as.matrix(dist(X, method = "euclidean")),
                             thresholdGini = params[iterParams])
      calculatedLabel <- cutree(result, length(unique(labels$V1)))

      inFM[iter, iterParams] <- round(as.numeric(FM_index(labels$V1, calculatedLabel)),
                                     decimalPlaces)
      inAM[iter, iterParams] <- round(as.numeric(mclust::adjustedRandIndex(labels$V1,
                                     calculatedLabel)), decimalPlaces)

      # naming columns
      colnames(inFM)[iterParams] <- paste("FM tG", params[iterParams], sep = " ")
      colnames(inAM)[iterParams] <- paste("AM tG", params[iterParams], sep = " ")
    }
  }

  # Hierarchical Clustering on Principal Components
  if (choseAlgorithm == "HCPC"){
    for (iterParams in 1:length(params)){
      #X.pca <- PCA(X, graph=FALSE)
      result <- FactoMineR::HCPC(res = X, nb.clust=length(unique(labels$V1)),
                                graph=FALSE)

      calculatedLabel <- result$data.clust$clust

      inFM[iter, iterParams] <- round(as.numeric(FM_index(labels$V1, calculatedLabel)),
                                     decimalPlaces)
      inAM[iter, iterParams] <- round(as.numeric(mclust::adjustedRandIndex(labels$V1,
                                     calculatedLabel)), decimalPlaces)

      colnames(inFM)[iterParams] <- paste("FM proba", params[iterParams], sep = " ")
      colnames(inAM)[iterParams] <- paste("AM proba", params[iterParams], sep = " ")
    }
  }

  if (choseAlgorithm == "cmeans"){
    for (iterParams in 1:length(params)){
      #input <- as.SparseSimilarityMatrix(as.matrix(dist(X, method = "euclidean")),
      #lower=-0.2)

      result <- cmeans(x=X, centers = length(unique(labels$V1)), dist='euclidean',
                      m=params[iterParams])

      calculatedLabel = result$cluster

      inFM[iter, iterParams] <- round(as.numeric(FM_index(labels$V1, calculatedLabel)),
                                     decimalPlaces)
    }
  }
}

```

```

inAM[iter, iterParams] <- round(as.numeric(mclust::adjustedRandIndex(labels$V1,
                                     calculatedLabel)), decimalPlaces)

colnames(inFM)[iterParams] <- paste("FM rate.par", params[iterParams], sep = " ")
colnames(inAM)[iterParams] <- paste("AM rate.par", params[iterParams], sep = " ")

}
}
}

output <- data.frame(inAM, inFM)

finalDataframe <- data.frame(data, output)
# we save result to file
savePath <- paste("../benchmarkResults", paste(choseAlgorithm, stand, by="_"), sep="/")
write.csv(finalDataframe, file = paste(savePath, "csv", sep = "."))

finalDataframe
}

```

2.2 Opis funkcji

Funkcja ta przyjmuje jako parametry wejściowe:

1. *choseAlgorithm* - wybrany algorytm do analizy skupień:
 - *spectral_clustering* - zaimplementowany przeze mnie algorytm spektralny
 - *hclust*
 - *genie*
 - *HCPC* - Hierarchical Clustering on Principal Components
 - *cmeans*
2. *stand* - wartość boolean informująca czy ma być przeprowadzana standaryzacja (*TRUE*) czy nie (*FALSE*)
3. *params* - wektor z wartościami parameterów danej funkcji

W metodach sprawdzam czy argumenty wejściowe są poprawne wpisane (poprzez wykorzystanie funkcji *stopifnot*).

2.3 Przykładowe wywołania funkcji

```

# spectral_clustering
params <- seq(2, 10, 1)
output_SC <- menu("spectral_clustering", stand=TRUE, params)

# hclust
output_HC <- menu("hclust", stand=TRUE, params=c(NA))

# genie
params <- seq(0.1, 0.9, by=0.1)
output_G <- menu("genie", stand=TRUE, params)

```

```
# HCPC
output_HCPC <- menu("HCPC", stand=TRUE, c(NA))

# cmeans
params <- c(2, 3, 4, 5, 6, 7, 8)
output_AP <- menu("cmeans", stand=TRUE, params=params)
```


3 Algorytm spektralny

3.1 Implementacja algorytmu

Implementacja poszczególnych funkcji algorytmu spektralnego znajdują się w pliku *spectral.R*. Dane linijki kodu komentowałem w trakcie pisania. Ewentualne inne rozwiązania danego zadania zamieściłem w komentarzach. Głównym powodem dla który wybierałem jeden sposób było przede wszystkim szybkość działania poszczególnego sposobu.

3.1.1 Funkcja *Mnn*

```
Mnn <- function(X, M){  
  # calculate the distance between two points and save it as a matrix  
  distOutput <- as.matrix(dist(X), method = "euclidean")  
  
  # order the matrix  
  orderedOutput <- apply(distOutput, 2, order)  
  # in first column is the same column value (1 - 1) so we want to delete it  
  orderedOutput <- orderedOutput[-1, ]  
  
  # choose only this rows, which are the closest  
  # t function to transpose result  
  S <- t(orderedOutput[1:M, ])  
}
```

3.1.2 Funkcja *Mnn_graph*

W tej funkcji dane składowe łączyłem za pomocą pętli *while*. Po testach zauważyłem, iż nie jest to wolny sposób. Jednym z decyzji, jakie podjąłem, było łączenie danych składowych (gdy liczba składowych jest większa niż 1). Postanowiłem, iż najlepszym rozwiązaniem (a zarazem najłatwiejszym) będzie połączenie poszczególnych składowych łącząc krawędzie o najniższych liczbach.

```
Mnn_graph <- function(S){  
  # convert into adjacency matrix  
  G <- matrix(0, nrow = nrow(S), ncol = nrow(S))  
  
  for(row in 1:nrow(S)) {  
    for(col in 1:ncol(S)) {  
      G[row, S[row, col]] <- 1  
      G[S[row, col], row] <- 1  
    }  
  }  
  
  # creating a graph from a adjacency matrix  
  ourGraph <- graph_from_adjacency_matrix(G, mode = c("undirected"),  
                                           weighted = NULL, diag = FALSE)  
  
  # calculating number of graph component  
  comp <- components(ourGraph)  
  componentsGroups <- groups(comp)  
  componentsNumber <- length(componentsGroups)
```

```

# if the number of component is bigger than 1 we add some edges
while (componentsNumber != 1) {
  G[componentsGroups[[componentsNumber]][1],
    componentsGroups[[componentsNumber-1]][1]] <- 1
  G[componentsGroups[[componentsNumber-1]][1],
    componentsGroups[[componentsNumber]][1]] <- 1
  componentsNumber <- componentsNumber - 1
}

G
}

```

3.1.3 Funkcja *Laplacian_eigen*

```

Laplacian_eigen <- function(G, k){
  stopifnot(k > 1)

  # creating graph to calculate a degree of a vertex
  # (optional solution: sum of a row or a column)
  ourGraph <- graph_from_adjacency_matrix(G, mode = c("undirected"),
                                           weighted = NULL, diag = FALSE)

  # first solution
  # calculating a degree of a vertex
  #vertexDegree <- degree(ourGraph)

  # using diag function create D matrix
  #D = diag(vertexDegree, nrow(G), ncol(G))
  #L = D - G

  # second solution
  L <- laplacian_matrix(ourGraph)

  #stopifnot(isSymmetric(L))
  #eigenStructure <- eigen(L, symmetric = TRUE) # <- too slow

  # SA - the smallest(leftmost) values
  eigenStructure <- eigs_sym(L, 10 * k, which = "SA")

  vectorNumbers <- k + 1

  E <- eigenStructure$vectors[, (ncol
                                (eigenStructure$vectors) - k + 1)
                               :ncol(eigenStructure$vectors)]

  # alternatives (when we use a eigen function):
  #E <- eigenStructure$vectors[, order(eigenStructure$values,
  # decreasing = FALSE)[1:vectorNumbers]]

  E
}

```

3.1.4 Funkcja *spectral_clustering*

Efektym finalnym jest funkcja *spectral_clustering*. Wykorzystuje ona wcześniej zaimplementowane funkcje oraz korzysta dodatkowo z funkcji *kmeans*, która w sposób losowy wybiera punkt początkowy. Rodzi to pewne problemy - testując daną funkcję najlepiej to wykonać kilka razy a następnie obliczyć średnią z danych eksperymentalnych. Ze względu na skomplikowość zadania postanowiłem tego nie wykonywać (badałem jedną próbę).

```
spectral_clustering <- function(X, M, k){  
  S <- Mnn(X, M)  
  G <- Mnn_graph(S)  
  E <- Laplacian_eigen(G, k)  
  kmeans(E, k)$cluster  
}
```

3.2 Badanie algorytmu na wszystkich zbiorach benchmarkowych

Jak już wspomniałem poprzez losowość funkcji *kmeans* wyniki przy danych wykonaniach mogą się różnić.

3.2.1 Bez standaryzacji zmiennych

Pierwszym etapem przy badaniu algorytmu *spectral_clustering* było wyznaczenie błędów metody dla różnych wartości parametrów *M*. Wszystkie wartości przedstawione w poniższej tabeli są również zapisane w pliku *spectral_clustering.csv*.

3.2.1.1 Zbiory dostarczone

Wartości indeksów AM dla różnych parametrów *M* są przedstawione ponżej:

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|--------------------------|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| fcps/atom.data.gz | 0.819 | 0.620 | 0.579 | 0.405 | 0.421 | 0.401 | 0.333 | 0.257 | 0.232 |
| fcps/chainlink.data.gz | 0.925 | 0.327 | 0.469 | 0.499 | 0.420 | 0.336 | 0.283 | 0.289 | 0.266 |
| fcps/engytime.data.gz | 0.397 | 0.342 | 0.243 | 0.214 | 0.211 | 0.241 | 0.194 | 0.192 | 0.171 |
| fcps/hepta.data.gz | 0.272 | 0.472 | 0.630 | 0.619 | 0.689 | 0.817 | 0.611 | 0.948 | 0.620 |
| fcps/lsun.data.gz | 0.739 | 0.439 | 0.558 | 0.401 | 0.398 | 0.517 | 0.416 | 0.319 | 0.343 |
| fcps/target.data.gz | 0.970 | 0.659 | 0.698 | 0.657 | 0.267 | 0.297 | 0.288 | 0.270 | 0.275 |
| fcps/tetra.data.gz | 0.332 | 0.297 | 0.674 | 0.935 | 0.531 | 0.540 | 0.730 | 0.730 | 0.595 |
| fcps/twodiamonds.data.gz | 0.362 | 0.290 | 0.264 | 0.386 | 0.175 | 0.159 | 0.311 | 0.205 | 0.191 |
| fcps/wingnut.data.gz | -0.001 | 0.056 | 0.113 | 0.045 | 0.117 | 0.100 | 0.115 | 0.131 | 0.070 |
| graves/dense.data.gz | 0.014 | 0.506 | 0.421 | 0.485 | 0.277 | 0.318 | 0.205 | 0.238 | 0.171 |
| graves/fuzzyx.data.gz | 0.091 | 0.311 | 0.702 | 0.622 | 0.548 | 0.438 | 0.530 | 0.585 | 0.597 |
| graves/line.data.gz | 0.510 | 0.274 | 0.048 | 0.101 | 0.138 | 0.071 | 0.052 | 0.085 | 0.027 |
| graves/parabolic.data.gz | 0.035 | 0.024 | 0.200 | 0.323 | 0.099 | 0.215 | 0.129 | 0.102 | 0.111 |
| graves/ring.data.gz | 0.757 | 0.720 | 0.496 | 0.354 | 0.387 | 0.270 | 0.229 | 0.176 | 0.200 |
| graves/zigzag.data.gz | 0.156 | -0.070 | 0.297 | 0.469 | 0.430 | 0.273 | 0.293 | 0.230 | 0.248 |
| other/iris.data.gz | 0.568 | 0.731 | 0.635 | 0.473 | 0.565 | 0.267 | 0.434 | 0.585 | 0.395 |
| other/iris5.data.gz | 0.396 | 0.218 | 0.201 | 0.377 | 0.344 | 0.293 | 0.302 | 0.228 | 0.133 |
| other/square.data.gz | 0.692 | 0.479 | 0.354 | 0.244 | 0.331 | 0.322 | 0.279 | 0.268 | 0.228 |
| sipu/a1.data.gz | 0.092 | 0.184 | 0.137 | 0.237 | 0.365 | 0.414 | 0.384 | 0.455 | 0.431 |
| sipu/a2.data.gz | 0.056 | 0.095 | 0.145 | 0.157 | 0.180 | 0.285 | 0.282 | 0.278 | 0.278 |
| sipu/a3.data.gz | 0.038 | 0.075 | 0.110 | 0.136 | 0.174 | 0.182 | 0.151 | 0.139 | 0.124 |
| sipu/aggregation.data.gz | 0.404 | 0.672 | 0.635 | 0.796 | 0.672 | 0.455 | 0.708 | 0.690 | 0.544 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------------------------|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| sipu/compound.data.gz | 0.484 | 0.740 | 0.753 | 0.461 | 0.560 | 0.264 | 0.373 | 0.325 | 0.320 |
| sipu/d31.data.gz | 0.004 | 0.113 | 0.106 | 0.210 | 0.229 | 0.291 | 0.215 | 0.361 | 0.370 |
| sipu/flame.data.gz | -0.030 | 0.564 | 0.417 | 0.390 | 0.406 | 0.104 | 0.223 | 0.211 | 0.112 |
| sipu/jain.data.gz | 0.324 | 0.295 | 0.216 | 0.107 | 0.056 | 0.146 | 0.041 | 0.184 | 0.136 |
| sipu/pathbased.data.gz | 0.398 | 0.490 | 0.540 | 0.612 | 0.601 | 0.677 | 0.565 | 0.414 | 0.394 |
| sipu/r15.data.gz | 0.121 | 0.193 | 0.273 | 0.229 | 0.193 | 0.497 | 0.575 | 0.536 | 0.591 |
| sipu/s1.data.gz | 0.132 | 0.042 | 0.213 | 0.285 | 0.485 | 0.506 | 0.571 | 0.580 | 0.596 |
| sipu/s2.data.gz | 0.119 | 0.231 | 0.326 | 0.408 | 0.374 | 0.523 | 0.475 | 0.450 | 0.690 |
| sipu/s3.data.gz | 0.108 | 0.208 | 0.288 | 0.310 | 0.385 | 0.440 | 0.482 | 0.497 | 0.488 |
| sipu/s4.data.gz | 0.111 | 0.188 | 0.228 | 0.291 | 0.321 | 0.360 | 0.385 | 0.445 | 0.505 |
| sipu/spiral.data.gz | 0.410 | 0.786 | 0.389 | 0.452 | 0.484 | 0.383 | 0.361 | 0.406 | 0.293 |
| sipu/unbalance.data.gz | 0.457 | 0.879 | 0.772 | 0.863 | 0.854 | 0.743 | 0.765 | 0.644 | 0.484 |
| wut/cross.data.gz | 0.335 | 0.441 | 0.543 | 0.786 | 0.651 | 0.398 | 0.569 | 0.479 | 0.487 |
| wut/smile.data.gz | 0.166 | 0.222 | 0.477 | 0.056 | 0.380 | 0.364 | 0.308 | 0.266 | 0.360 |
| wut/twosplashes.data.gz | 0.546 | 0.309 | 0.324 | 0.331 | 0.107 | 0.191 | 0.329 | 0.214 | 0.225 |
| wut/x1.data.gz | 0.593 | 0.499 | 0.870 | 0.815 | 0.697 | 0.566 | 0.561 | 0.464 | 0.379 |
| wut/x2.data.gz | 0.467 | 0.556 | 0.454 | 0.386 | 0.560 | 0.409 | 0.206 | 0.291 | 0.277 |
| wut/x3.data.gz | 0.481 | 0.643 | 0.898 | 0.656 | 0.708 | 0.710 | 0.718 | 0.520 | 0.380 |
| wut/z1.data.gz | 0.347 | 0.283 | 0.183 | 0.191 | 0.181 | 0.226 | 0.187 | 0.191 | 0.181 |
| wut/z2.data.gz | 0.576 | 0.395 | 0.320 | 0.600 | 0.643 | 0.466 | 0.399 | 0.425 | 0.387 |
| wut/z3.data.gz | 0.574 | 0.851 | 0.848 | 0.772 | 0.754 | 0.494 | 0.490 | 0.513 | 0.339 |

Wartości indeksów FM dla różnych parametrów M są przedstawione ponżej:

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|--------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| fcps/atom.data.gz | 0.910 | 0.790 | 0.762 | 0.637 | 0.649 | 0.634 | 0.577 | 0.510 | 0.482 |
| fcps/chainlink.data.gz | 0.963 | 0.615 | 0.685 | 0.706 | 0.648 | 0.580 | 0.532 | 0.538 | 0.516 |
| fcps/engytime.data.gz | 0.703 | 0.620 | 0.538 | 0.505 | 0.479 | 0.500 | 0.459 | 0.443 | 0.424 |
| fcps/hepta.data.gz | 0.524 | 0.637 | 0.731 | 0.723 | 0.756 | 0.850 | 0.683 | 0.956 | 0.675 |
| fcps/lsun.data.gz | 0.860 | 0.638 | 0.705 | 0.601 | 0.587 | 0.679 | 0.599 | 0.515 | 0.542 |
| fcps/target.data.gz | 0.985 | 0.813 | 0.832 | 0.807 | 0.532 | 0.539 | 0.531 | 0.513 | 0.519 |
| fcps/tetra.data.gz | 0.631 | 0.600 | 0.782 | 0.951 | 0.664 | 0.645 | 0.801 | 0.801 | 0.703 |
| fcps/twodiamonds.data.gz | 0.706 | 0.598 | 0.567 | 0.630 | 0.465 | 0.426 | 0.558 | 0.455 | 0.447 |
| fcps/wingnut.data.gz | 0.499 | 0.481 | 0.472 | 0.373 | 0.430 | 0.373 | 0.420 | 0.390 | 0.337 |
| graves/dense.data.gz | 0.579 | 0.731 | 0.668 | 0.697 | 0.531 | 0.564 | 0.455 | 0.487 | 0.415 |
| graves/fuzzyx.data.gz | 0.466 | 0.559 | 0.783 | 0.703 | 0.637 | 0.545 | 0.617 | 0.664 | 0.682 |
| graves/line.data.gz | 0.813 | 0.661 | 0.452 | 0.474 | 0.485 | 0.394 | 0.359 | 0.385 | 0.420 |
| graves/parabolic.data.gz | 0.517 | 0.469 | 0.501 | 0.575 | 0.416 | 0.483 | 0.388 | 0.347 | 0.362 |
| graves/ring.data.gz | 0.879 | 0.849 | 0.705 | 0.597 | 0.622 | 0.523 | 0.480 | 0.431 | 0.449 |
| graves/zigzag.data.gz | 0.562 | 0.436 | 0.560 | 0.662 | 0.633 | 0.503 | 0.517 | 0.459 | 0.474 |
| other/iris.data.gz | 0.771 | 0.822 | 0.746 | 0.624 | 0.695 | 0.461 | 0.597 | 0.712 | 0.561 |
| other/iris5.data.gz | 0.697 | 0.538 | 0.504 | 0.600 | 0.568 | 0.523 | 0.531 | 0.461 | 0.384 |
| other/square.data.gz | 0.848 | 0.721 | 0.614 | 0.519 | 0.580 | 0.569 | 0.534 | 0.520 | 0.480 |
| sipu/a1.data.gz | 0.309 | 0.382 | 0.346 | 0.421 | 0.511 | 0.544 | 0.524 | 0.573 | 0.555 |
| sipu/a2.data.gz | 0.237 | 0.277 | 0.323 | 0.333 | 0.353 | 0.436 | 0.434 | 0.430 | 0.430 |
| sipu/a3.data.gz | 0.197 | 0.241 | 0.277 | 0.302 | 0.337 | 0.343 | 0.316 | 0.305 | 0.291 |
| sipu/aggregation.data.gz | 0.644 | 0.783 | 0.750 | 0.849 | 0.769 | 0.574 | 0.771 | 0.758 | 0.635 |
| sipu/compound.data.gz | 0.698 | 0.830 | 0.823 | 0.592 | 0.661 | 0.461 | 0.507 | 0.465 | 0.459 |
| sipu/d31.data.gz | 0.185 | 0.299 | 0.293 | 0.381 | 0.396 | 0.440 | 0.384 | 0.494 | 0.500 |
| sipu/flame.data.gz | 0.605 | 0.769 | 0.664 | 0.640 | 0.653 | 0.446 | 0.489 | 0.476 | 0.382 |
| sipu/jain.data.gz | 0.700 | 0.702 | 0.566 | 0.459 | 0.401 | 0.460 | 0.403 | 0.476 | 0.411 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| sipu/pathbased.data.gz | 0.652 | 0.678 | 0.695 | 0.735 | 0.727 | 0.780 | 0.699 | 0.578 | 0.563 |
| sipu/r15.data.gz | 0.354 | 0.406 | 0.462 | 0.431 | 0.406 | 0.610 | 0.664 | 0.633 | 0.668 |
| sipu/s1.data.gz | 0.364 | 0.294 | 0.420 | 0.469 | 0.604 | 0.619 | 0.658 | 0.666 | 0.674 |
| sipu/s2.data.gz | 0.349 | 0.429 | 0.490 | 0.547 | 0.518 | 0.620 | 0.583 | 0.567 | 0.736 |
| sipu/s3.data.gz | 0.333 | 0.397 | 0.451 | 0.458 | 0.506 | 0.539 | 0.563 | 0.568 | 0.560 |
| sipu/s4.data.gz | 0.336 | 0.379 | 0.402 | 0.438 | 0.449 | 0.471 | 0.479 | 0.521 | 0.561 |
| sipu/spiral.data.gz | 0.688 | 0.858 | 0.575 | 0.610 | 0.633 | 0.551 | 0.534 | 0.581 | 0.473 |
| sipu/unbalance.data.gz | 0.705 | 0.920 | 0.838 | 0.901 | 0.895 | 0.813 | 0.830 | 0.742 | 0.621 |
| wut/cross.data.gz | 0.591 | 0.635 | 0.668 | 0.838 | 0.733 | 0.533 | 0.670 | 0.597 | 0.604 |
| wut/smile.data.gz | 0.507 | 0.493 | 0.624 | 0.353 | 0.538 | 0.521 | 0.473 | 0.439 | 0.523 |
| wut/twosplashes.data.gz | 0.773 | 0.598 | 0.592 | 0.589 | 0.397 | 0.453 | 0.573 | 0.463 | 0.475 |
| wut/x1.data.gz | 0.787 | 0.718 | 0.914 | 0.878 | 0.798 | 0.705 | 0.702 | 0.630 | 0.563 |
| wut/x2.data.gz | 0.731 | 0.713 | 0.621 | 0.568 | 0.694 | 0.576 | 0.425 | 0.476 | 0.462 |
| wut/x3.data.gz | 0.698 | 0.767 | 0.925 | 0.741 | 0.781 | 0.784 | 0.792 | 0.640 | 0.521 |
| wut/z1.data.gz | 0.621 | 0.566 | 0.433 | 0.410 | 0.389 | 0.417 | 0.379 | 0.375 | 0.363 |
| wut/z2.data.gz | 0.793 | 0.615 | 0.574 | 0.737 | 0.768 | 0.637 | 0.584 | 0.610 | 0.573 |
| wut/z3.data.gz | 0.758 | 0.900 | 0.892 | 0.839 | 0.827 | 0.630 | 0.629 | 0.646 | 0.503 |

3.2.1.2 Moje zbiory danych

Wyniki działania na moich zbiorach danych przedstawiają się następująco:

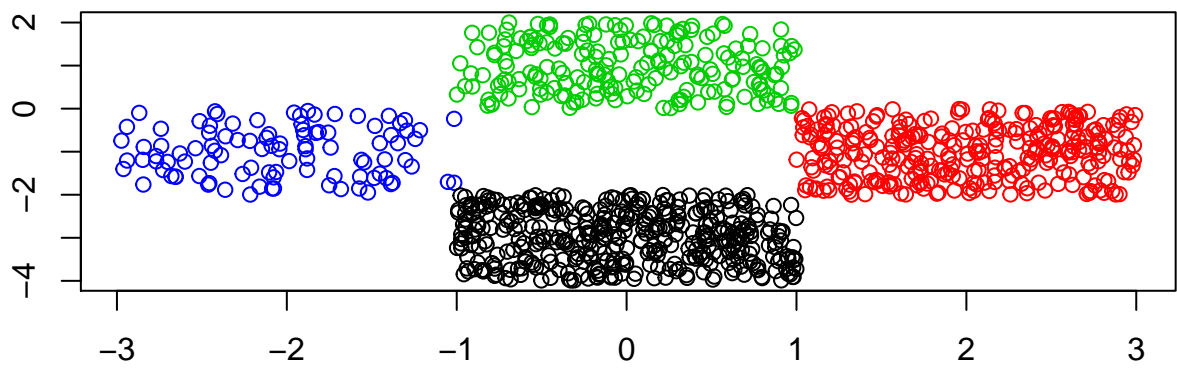
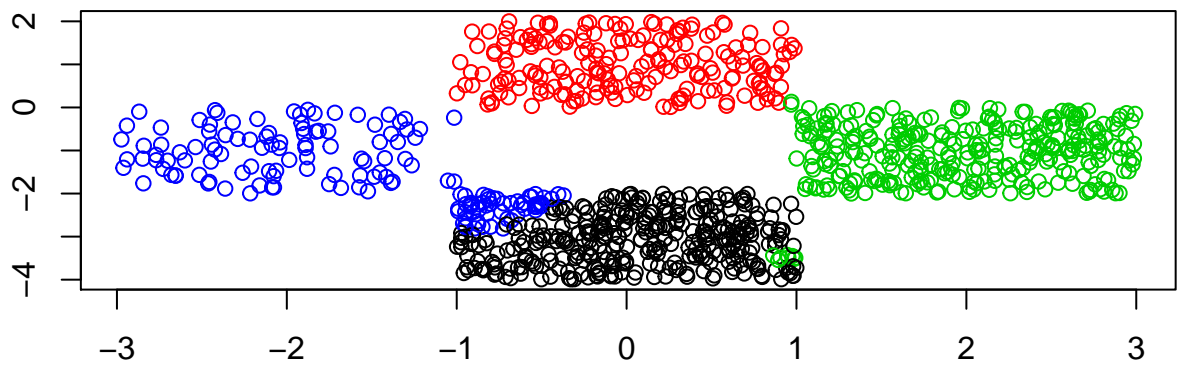
Wartości indeksów AM dla różnych parametrów M są przedstawione ponżej:

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.648 | 0.393 | 0.298 | 0.427 | 0.413 | 0.327 | 0.230 | 0.299 | 0.305 |
| benchmark2.data | 0.134 | 0.115 | 0.185 | 0.510 | 0.480 | 0.509 | 0.580 | 0.775 | 0.695 |
| benchmark3.data | 0.363 | 0.539 | 0.661 | 0.902 | 0.607 | 0.541 | 0.606 | 0.478 | 0.497 |
| benchmark4.data | 0.322 | 0.446 | 0.679 | 0.738 | 0.444 | 0.735 | 0.455 | 0.566 | 0.526 |

Wartości indeksów FM dla różnych parametrów M są przedstawione ponżej:

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.827 | 0.652 | 0.583 | 0.654 | 0.643 | 0.572 | 0.482 | 0.547 | 0.552 |
| benchmark2.data | 0.385 | 0.372 | 0.415 | 0.629 | 0.610 | 0.629 | 0.676 | 0.814 | 0.748 |
| benchmark3.data | 0.609 | 0.688 | 0.746 | 0.922 | 0.682 | 0.635 | 0.680 | 0.575 | 0.591 |
| benchmark4.data | 0.571 | 0.610 | 0.759 | 0.792 | 0.549 | 0.787 | 0.553 | 0.654 | 0.624 |

3.2.1.3 Przykładowy zbiór



3.2.2 Ze standaryzacją zmiennych

3.2.2.1 Zbiory dostaczone

Wartości indeksów AM dla różnych parametrów M :

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|--------------------------|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| fcps/atom.data.gz | 0.819 | 0.622 | 0.579 | 0.455 | 0.344 | 0.399 | 0.343 | 0.317 | 0.230 |
| fcps/chainlink.data.gz | 0.925 | 0.377 | 0.480 | 0.442 | 0.446 | 0.244 | 0.280 | 0.267 | 0.190 |
| fcps/engytime.data.gz | 0.397 | 0.342 | 0.092 | 0.220 | 0.329 | 0.186 | 0.233 | 0.206 | 0.151 |
| fcps/hepta.data.gz | 0.269 | 0.472 | 0.624 | 0.729 | 0.850 | 0.803 | 0.763 | 0.780 | 0.894 |
| fcps/lsun.data.gz | 0.739 | 0.439 | 0.558 | 0.535 | 0.400 | 0.452 | 0.338 | 0.421 | 0.293 |
| fcps/target.data.gz | 0.970 | 0.655 | 0.698 | 0.405 | 0.339 | 0.336 | 0.264 | 0.237 | 0.256 |
| fcps/tetra.data.gz | 0.332 | 0.713 | 0.670 | 0.581 | 0.715 | 0.791 | 0.600 | 0.459 | 0.554 |
| fcps/twodiamonds.data.gz | 0.362 | 0.467 | 0.430 | 0.368 | 0.198 | 0.209 | 0.214 | 0.208 | 0.181 |
| fcps/wingnut.data.gz | -0.001 | 0.057 | 0.143 | 0.044 | 0.113 | 0.090 | 0.130 | 0.068 | 0.115 |
| graves/dense.data.gz | 0.671 | 0.574 | 0.202 | 0.485 | 0.277 | 0.487 | 0.206 | 0.273 | 0.209 |
| graves/fuzzix.data.gz | 0.091 | 0.464 | 0.551 | 0.461 | 0.548 | 0.456 | 0.531 | 0.486 | 0.537 |
| graves/line.data.gz | 0.257 | 0.274 | 0.019 | 0.094 | 0.065 | 0.020 | 0.049 | 0.085 | 0.058 |
| graves/parabolic.data.gz | 0.035 | 0.456 | 0.214 | 0.218 | 0.222 | 0.164 | 0.139 | 0.078 | 0.144 |
| graves/ring.data.gz | 0.757 | 0.680 | 0.496 | 0.375 | 0.376 | 0.335 | 0.272 | 0.215 | 0.201 |
| graves/zigzag.data.gz | 0.156 | -0.070 | 0.568 | 0.469 | 0.232 | 0.276 | 0.293 | 0.215 | 0.222 |
| other/iris.data.gz | 0.568 | 0.731 | 0.320 | 0.454 | 0.321 | 0.371 | 0.369 | 0.584 | 0.360 |
| other/iris5.data.gz | 0.396 | 0.218 | 0.404 | 0.377 | 0.126 | 0.270 | 0.341 | 0.313 | 0.178 |
| other/square.data.gz | 0.692 | 0.479 | 0.439 | 0.336 | 0.285 | 0.580 | 0.209 | 0.241 | 0.181 |
| sipu/a1.data.gz | 0.092 | 0.184 | 0.208 | 0.312 | 0.203 | 0.428 | 0.401 | 0.469 | 0.404 |
| sipu/a2.data.gz | 0.057 | 0.095 | 0.145 | 0.189 | 0.215 | 0.261 | 0.298 | 0.287 | 0.270 |
| sipu/a3.data.gz | 0.038 | 0.075 | 0.100 | 0.141 | 0.174 | 0.198 | 0.206 | 0.240 | 0.245 |
| sipu/aggregation.data.gz | 0.404 | 0.672 | 0.797 | 0.429 | 0.866 | 0.755 | 0.569 | 0.622 | 0.682 |
| sipu/compound.data.gz | 0.484 | 0.406 | 0.783 | 0.578 | 0.411 | 0.535 | 0.379 | 0.344 | 0.571 |
| sipu/d31.data.gz | 0.004 | 0.071 | 0.112 | 0.066 | 0.214 | 0.267 | 0.320 | 0.257 | 0.395 |
| sipu/flame.data.gz | -0.030 | 0.564 | 0.429 | 0.075 | 0.163 | 0.249 | 0.109 | 0.203 | 0.181 |
| sipu/jain.data.gz | 0.324 | 0.295 | 0.118 | 0.139 | 0.221 | 0.075 | 0.069 | 0.156 | 0.076 |
| sipu/pathbased.data.gz | 0.398 | 0.490 | 0.540 | 0.612 | 0.612 | 0.496 | 0.467 | 0.441 | 0.356 |
| sipu/r15.data.gz | 0.121 | 0.181 | 0.221 | 0.237 | 0.246 | 0.499 | 0.575 | 0.442 | 0.546 |
| sipu/s1.data.gz | 0.132 | 0.181 | 0.213 | 0.396 | 0.477 | 0.466 | 0.477 | 0.571 | 0.669 |
| sipu/s2.data.gz | 0.119 | 0.231 | 0.326 | 0.317 | 0.453 | 0.510 | 0.464 | 0.625 | 0.614 |
| sipu/s3.data.gz | 0.107 | 0.208 | 0.288 | 0.327 | 0.372 | 0.406 | 0.490 | 0.528 | 0.571 |
| sipu/s4.data.gz | 0.111 | 0.188 | 0.228 | 0.291 | 0.321 | 0.361 | 0.404 | 0.435 | 0.451 |
| sipu/spiral.data.gz | 0.374 | 0.786 | 0.389 | 0.446 | 0.370 | 0.288 | 0.404 | 0.406 | 0.324 |
| sipu/unbalance.data.gz | 0.528 | 0.879 | 0.331 | 0.898 | 0.893 | 0.719 | 0.786 | 0.686 | 0.630 |
| wut/cross.data.gz | 0.335 | 0.441 | 0.543 | 0.786 | 0.695 | 0.493 | 0.541 | 0.511 | 0.504 |
| wut/smile.data.gz | 0.166 | 0.222 | 0.260 | 0.318 | 0.499 | 0.364 | 0.137 | 0.383 | 0.345 |
| wut/twosplashes.data.gz | 0.546 | 0.309 | 0.324 | 0.286 | 0.274 | 0.308 | 0.246 | 0.209 | 0.176 |
| wut/x1.data.gz | 0.593 | 1.000 | 0.446 | 0.693 | 0.697 | 0.438 | 0.539 | 0.467 | 0.387 |
| wut/x2.data.gz | 0.467 | 0.556 | 0.623 | 0.549 | 0.464 | 0.359 | 0.367 | 0.313 | 0.282 |
| wut/x3.data.gz | 0.481 | 0.613 | 0.601 | 0.558 | 0.503 | 0.618 | 0.553 | 0.477 | 0.457 |
| wut/z1.data.gz | 0.347 | 0.283 | 0.251 | 0.191 | 0.188 | 0.203 | 0.156 | 0.195 | 0.178 |
| wut/z2.data.gz | 0.576 | 0.395 | 0.321 | 0.246 | 0.816 | 0.436 | 0.399 | 0.481 | 0.422 |
| wut/z3.data.gz | 0.574 | 0.851 | 0.848 | 0.562 | 0.681 | 0.551 | 0.372 | 0.468 | 0.314 |

Wartości indeksów FM dla różnych parametrów M :

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|--------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| fcps/atom.data.gz | 0.910 | 0.791 | 0.762 | 0.676 | 0.587 | 0.632 | 0.586 | 0.563 | 0.479 |
| fcps/chainlink.data.gz | 0.963 | 0.640 | 0.693 | 0.665 | 0.668 | 0.501 | 0.534 | 0.517 | 0.442 |
| fcps/engytime.data.gz | 0.703 | 0.620 | 0.464 | 0.508 | 0.582 | 0.462 | 0.489 | 0.462 | 0.396 |
| fcps/hepta.data.gz | 0.522 | 0.637 | 0.728 | 0.796 | 0.881 | 0.839 | 0.800 | 0.810 | 0.911 |
| fcps/lsun.data.gz | 0.860 | 0.638 | 0.705 | 0.688 | 0.584 | 0.624 | 0.531 | 0.602 | 0.497 |
| fcps/target.data.gz | 0.985 | 0.811 | 0.832 | 0.678 | 0.576 | 0.574 | 0.508 | 0.481 | 0.500 |
| fcps/tetra.data.gz | 0.631 | 0.815 | 0.779 | 0.705 | 0.783 | 0.846 | 0.692 | 0.586 | 0.672 |
| fcps/twodiamonds.data.gz | 0.706 | 0.721 | 0.677 | 0.620 | 0.565 | 0.471 | 0.469 | 0.458 | 0.428 |
| fcps/wingnut.data.gz | 0.499 | 0.515 | 0.492 | 0.372 | 0.424 | 0.390 | 0.384 | 0.335 | 0.356 |
| graves/dense.data.gz | 0.836 | 0.761 | 0.509 | 0.697 | 0.531 | 0.697 | 0.459 | 0.522 | 0.457 |
| graves/fuzzzyx.data.gz | 0.466 | 0.632 | 0.666 | 0.583 | 0.637 | 0.557 | 0.619 | 0.583 | 0.624 |
| graves/line.data.gz | 0.694 | 0.661 | 0.461 | 0.464 | 0.444 | 0.375 | 0.377 | 0.385 | 0.344 |
| graves/parabolic.data.gz | 0.517 | 0.696 | 0.509 | 0.498 | 0.492 | 0.449 | 0.415 | 0.336 | 0.390 |
| graves/ring.data.gz | 0.879 | 0.826 | 0.705 | 0.614 | 0.613 | 0.578 | 0.522 | 0.465 | 0.449 |
| graves/zigzag.data.gz | 0.562 | 0.436 | 0.734 | 0.662 | 0.491 | 0.504 | 0.517 | 0.445 | 0.448 |
| other/iris.data.gz | 0.771 | 0.822 | 0.559 | 0.628 | 0.517 | 0.541 | 0.541 | 0.712 | 0.535 |
| other/iris5.data.gz | 0.697 | 0.538 | 0.623 | 0.600 | 0.410 | 0.505 | 0.566 | 0.541 | 0.418 |
| other/square.data.gz | 0.848 | 0.721 | 0.699 | 0.589 | 0.540 | 0.763 | 0.483 | 0.492 | 0.434 |
| sipu/a1.data.gz | 0.309 | 0.382 | 0.399 | 0.474 | 0.393 | 0.554 | 0.531 | 0.583 | 0.531 |
| sipu/a2.data.gz | 0.238 | 0.277 | 0.323 | 0.360 | 0.381 | 0.417 | 0.446 | 0.436 | 0.424 |
| sipu/a3.data.gz | 0.197 | 0.241 | 0.267 | 0.306 | 0.336 | 0.357 | 0.363 | 0.391 | 0.396 |
| sipu/aggregation.data.gz | 0.644 | 0.783 | 0.858 | 0.630 | 0.897 | 0.808 | 0.657 | 0.704 | 0.760 |
| sipu/compound.data.gz | 0.698 | 0.638 | 0.855 | 0.683 | 0.542 | 0.642 | 0.510 | 0.480 | 0.669 |
| sipu/d31.data.gz | 0.185 | 0.259 | 0.299 | 0.255 | 0.384 | 0.424 | 0.463 | 0.417 | 0.518 |
| sipu/flame.data.gz | 0.605 | 0.769 | 0.675 | 0.491 | 0.477 | 0.516 | 0.393 | 0.486 | 0.440 |
| sipu/jain.data.gz | 0.700 | 0.702 | 0.543 | 0.479 | 0.528 | 0.389 | 0.362 | 0.439 | 0.361 |
| sipu/pathbased.data.gz | 0.652 | 0.678 | 0.695 | 0.735 | 0.730 | 0.646 | 0.621 | 0.601 | 0.532 |
| sipu/r15.data.gz | 0.354 | 0.397 | 0.426 | 0.437 | 0.443 | 0.612 | 0.664 | 0.566 | 0.642 |
| sipu/s1.data.gz | 0.364 | 0.399 | 0.420 | 0.542 | 0.597 | 0.588 | 0.594 | 0.659 | 0.729 |
| sipu/s2.data.gz | 0.349 | 0.429 | 0.490 | 0.487 | 0.575 | 0.615 | 0.582 | 0.694 | 0.683 |
| sipu/s3.data.gz | 0.332 | 0.397 | 0.451 | 0.469 | 0.495 | 0.510 | 0.568 | 0.591 | 0.623 |
| sipu/s4.data.gz | 0.336 | 0.379 | 0.402 | 0.438 | 0.449 | 0.472 | 0.489 | 0.511 | 0.518 |
| sipu/spiral.data.gz | 0.669 | 0.858 | 0.575 | 0.606 | 0.540 | 0.471 | 0.568 | 0.581 | 0.505 |
| sipu/unbalance.data.gz | 0.736 | 0.920 | 0.605 | 0.927 | 0.923 | 0.797 | 0.846 | 0.776 | 0.732 |
| wut/cross.data.gz | 0.591 | 0.635 | 0.668 | 0.838 | 0.768 | 0.607 | 0.646 | 0.625 | 0.625 |
| wut/smile.data.gz | 0.507 | 0.493 | 0.476 | 0.496 | 0.633 | 0.521 | 0.360 | 0.543 | 0.507 |
| wut/twosplashes.data.gz | 0.773 | 0.598 | 0.592 | 0.558 | 0.534 | 0.559 | 0.497 | 0.459 | 0.421 |
| wut/x1.data.gz | 0.787 | 1.000 | 0.645 | 0.794 | 0.798 | 0.602 | 0.686 | 0.632 | 0.569 |
| wut/x2.data.gz | 0.731 | 0.713 | 0.745 | 0.686 | 0.621 | 0.535 | 0.554 | 0.509 | 0.467 |
| wut/x3.data.gz | 0.698 | 0.746 | 0.724 | 0.677 | 0.643 | 0.714 | 0.665 | 0.602 | 0.590 |
| wut/z1.data.gz | 0.621 | 0.566 | 0.535 | 0.410 | 0.391 | 0.398 | 0.348 | 0.379 | 0.360 |
| wut/z2.data.gz | 0.793 | 0.615 | 0.544 | 0.564 | 0.889 | 0.612 | 0.584 | 0.648 | 0.601 |
| wut/z3.data.gz | 0.758 | 0.900 | 0.892 | 0.687 | 0.772 | 0.676 | 0.528 | 0.612 | 0.479 |

3.2.2.2 Moje zbiory

Wyniki działania na moich zbiorach danych przedstawiają się następująco:

Wartości indeksów AM dla różnych parametrów M są przedstawione ponżej:

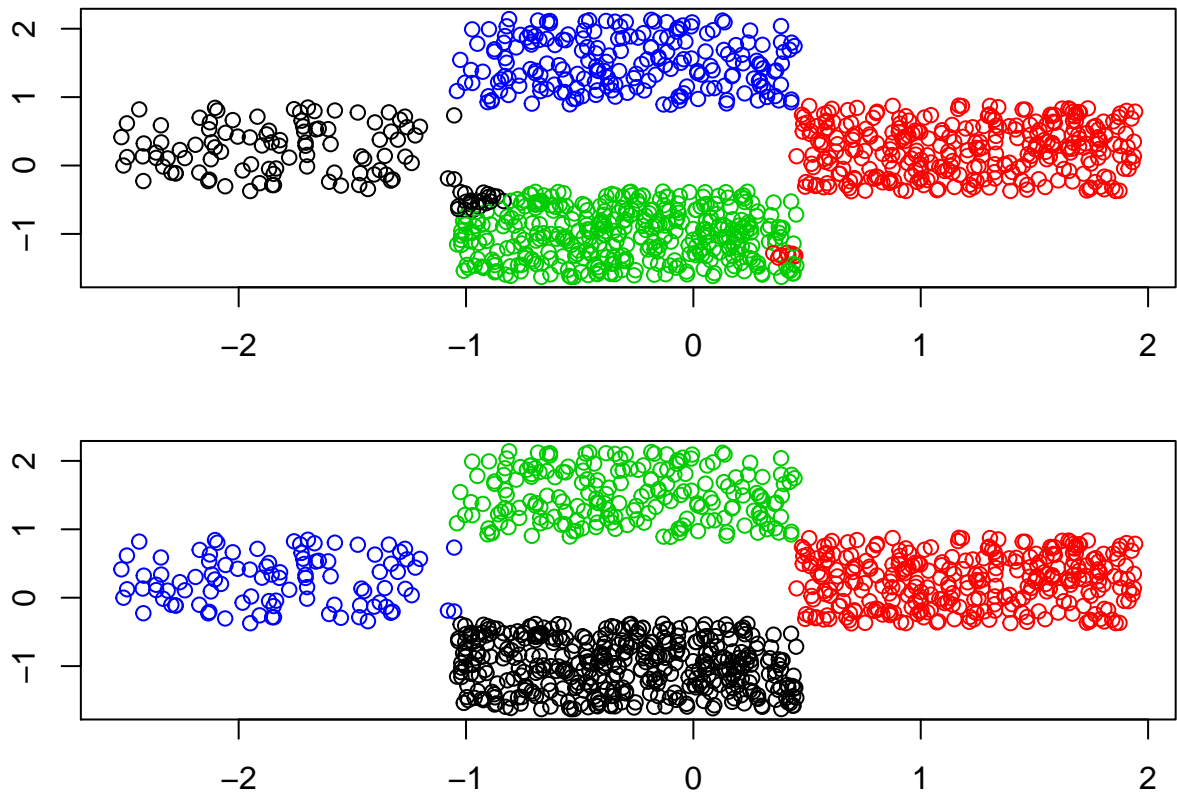
| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.851 | 0.458 | 0.266 | 0.391 | 0.352 | 0.379 | 0.267 | 0.275 | 0.219 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark2.data | 0.103 | 0.247 | 0.304 | 0.401 | 0.339 | 0.409 | 0.564 | 0.231 | 0.421 |
| benchmark3.data | 0.363 | 0.545 | 0.646 | 0.550 | 0.863 | 0.651 | 0.742 | 0.677 | 0.457 |
| benchmark4.data | 0.320 | 0.446 | 0.701 | 0.597 | 0.511 | 0.642 | 0.573 | 0.602 | 0.522 |

Wartości indeksów FM dla różnych parametrów M są przedstawione ponżej:

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.926 | 0.699 | 0.585 | 0.627 | 0.594 | 0.617 | 0.517 | 0.525 | 0.469 |
| benchmark2.data | 0.364 | 0.460 | 0.497 | 0.559 | 0.519 | 0.555 | 0.658 | 0.432 | 0.556 |
| benchmark3.data | 0.609 | 0.692 | 0.736 | 0.663 | 0.890 | 0.718 | 0.796 | 0.747 | 0.554 |
| benchmark4.data | 0.569 | 0.610 | 0.775 | 0.687 | 0.603 | 0.710 | 0.655 | 0.685 | 0.617 |

3.2.2.3 Przykładowy zbiór



4 Algorytm *hclust*

4.1 Badanie algorytmu na wszystkich zbiorach benchmarkowych

4.1.1 Bez standaryzacji zmiennych

4.1.1.1 Zbiory dostarczone

Wartości indeksów AM dla różnych algorytmów hierarchicznych:

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|--------------------------|--------|---------|--------|----------|---------|----------|--------|----------|
| fcps/atom.data.gz | 1.000 | 0.099 | 1.000 | 0.084 | 0.099 | 0.213 | 0.000 | 0.003 |
| fcps/chainlink.data.gz | 0.394 | 0.280 | 1.000 | 0.313 | 0.272 | 0.401 | 0.289 | 0.204 |
| fcps/engytime.data.gz | 0.754 | 0.718 | 0.000 | 0.041 | 0.051 | 0.032 | 0.023 | 0.000 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/lsun.data.gz | 0.366 | 0.369 | 1.000 | 0.405 | 0.361 | 0.307 | 0.471 | 0.711 |
| fcps/target.data.gz | 0.639 | 0.639 | 1.000 | 0.207 | 0.146 | 0.299 | 0.255 | 0.090 |
| fcps/tetra.data.gz | 1.000 | 0.967 | 0.000 | 0.987 | 0.993 | 1.000 | 0.671 | 0.967 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 | 0.000 | 0.985 | 0.995 | 0.995 | 0.648 | 0.995 |
| fcps/wingnut.data.gz | 1.000 | 0.601 | 1.000 | 0.469 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/dense.data.gz | 1.000 | 0.921 | 0.001 | 0.257 | 0.921 | 0.004 | 0.921 | 0.921 |
| graves/fuzzyx.data.gz | 0.694 | 0.637 | 0.000 | 0.654 | 0.743 | 0.615 | 0.566 | 0.270 |
| graves/line.data.gz | -0.064 | -0.104 | 1.000 | -0.071 | -0.066 | -0.017 | -0.078 | -0.066 |
| graves/parabolic.data.gz | 0.391 | 0.621 | 0.000 | 0.589 | 0.599 | 0.602 | 0.689 | 0.509 |
| graves/ring.data.gz | 0.150 | 0.116 | 1.000 | 0.209 | 0.114 | 0.159 | 0.040 | 0.029 |
| graves/zigzag.data.gz | 0.142 | 0.142 | 1.000 | 0.130 | 0.129 | 0.132 | 0.143 | 0.182 |
| other/iris.data.gz | 0.759 | 0.731 | 0.564 | 0.642 | 0.759 | 0.746 | 0.745 | 0.564 |
| other/iris5.data.gz | 0.558 | 0.506 | 0.148 | 0.337 | 0.558 | 0.532 | 0.531 | 0.148 |
| other/square.data.gz | 0.004 | 0.253 | 1.000 | 0.168 | 0.152 | 0.220 | 0.218 | 0.184 |
| sipu/a1.data.gz | 0.920 | 0.914 | 0.444 | 0.916 | 0.925 | 0.804 | 0.668 | 0.866 |
| sipu/a2.data.gz | 0.932 | 0.922 | 0.348 | 0.908 | 0.934 | 0.857 | 0.625 | 0.815 |
| sipu/a3.data.gz | 0.947 | 0.937 | 0.315 | 0.918 | 0.944 | 0.850 | 0.642 | 0.843 |
| sipu/aggregation.data.gz | 0.794 | 0.813 | 0.804 | 0.774 | 0.993 | 0.683 | 0.949 | 0.993 |
| sipu/compound.data.gz | 0.558 | 0.552 | 0.742 | 0.793 | 0.803 | 0.817 | 0.773 | 0.799 |
| sipu/d31.data.gz | 0.916 | 0.920 | 0.174 | 0.924 | 0.907 | 0.775 | 0.602 | 0.840 |
| sipu/flame.data.gz | 0.367 | 0.187 | 0.013 | -0.042 | 0.442 | 0.409 | 0.419 | 0.013 |
| sipu/jain.data.gz | 0.515 | 0.515 | 0.256 | 0.779 | 0.779 | 0.515 | 0.477 | 0.768 |
| sipu/pathbased.data.gz | 0.424 | 0.485 | 0.001 | 0.346 | 0.444 | 0.355 | 0.162 | 0.425 |
| sipu/r15.data.gz | 0.982 | 0.982 | 0.542 | 0.979 | 0.989 | 0.975 | 0.772 | 0.982 |
| sipu/s1.data.gz | 0.981 | 0.983 | 0.464 | 0.971 | 0.982 | 0.802 | 0.648 | 0.981 |
| sipu/s2.data.gz | 0.900 | 0.906 | 0.000 | 0.791 | 0.913 | 0.679 | 0.481 | 0.779 |
| sipu/s3.data.gz | 0.665 | 0.677 | 0.000 | 0.509 | 0.596 | 0.419 | 0.407 | 0.494 |
| sipu/s4.data.gz | 0.590 | 0.553 | 0.000 | 0.417 | 0.493 | 0.371 | 0.284 | 0.430 |
| sipu/spiral.data.gz | 0.008 | -0.001 | 1.000 | 0.002 | -0.002 | 0.002 | -0.003 | 0.009 |
| sipu/unbalance.data.gz | 1.000 | 1.000 | 0.999 | 0.612 | 1.000 | 0.613 | 0.613 | 1.000 |
| wut/cross.data.gz | 0.154 | 0.072 | 0.000 | 0.359 | 0.001 | 0.045 | 0.006 | 0.000 |
| wut/smile.data.gz | 0.439 | 0.452 | 1.000 | 0.499 | 0.575 | 0.484 | 0.481 | 0.605 |
| wut/twosplashes.data.gz | 0.214 | 0.192 | 0.000 | 0.219 | 0.002 | 0.192 | 0.279 | 0.002 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.669 | 0.191 | -0.007 | 0.209 | 0.250 | -0.008 | -0.007 | 0.271 |
| wut/x3.data.gz | 0.821 | 0.869 | 0.007 | 0.576 | 0.505 | 0.534 | 0.644 | 0.567 |
| wut/z1.data.gz | 0.202 | 0.184 | 0.122 | 0.181 | 0.214 | 0.200 | 0.200 | 0.047 |
| wut/z2.data.gz | 0.477 | 0.434 | 0.735 | 0.441 | 0.512 | 0.386 | 0.523 | 0.574 |

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|----------------|--------|---------|--------|----------|---------|----------|--------|----------|
| wut/z3.data.gz | 0.995 | 1.000 | 0.739 | 0.933 | 0.998 | 0.940 | 0.995 | 1.000 |

Wartości indeksów FM dla różnych algorytmów hierarchicznych:

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|--------------------------|--------|---------|--------|----------|---------|----------|--------|----------|
| fcps/atom.data.gz | 1.000 | 0.646 | 1.000 | 0.647 | 0.646 | 0.661 | 0.702 | 0.687 |
| fcps/chainlink.data.gz | 0.718 | 0.679 | 1.000 | 0.689 | 0.677 | 0.721 | 0.682 | 0.659 |
| fcps/engytime.data.gz | 0.877 | 0.860 | 0.707 | 0.654 | 0.650 | 0.659 | 0.654 | 0.702 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/lsun.data.gz | 0.603 | 0.605 | 1.000 | 0.627 | 0.604 | 0.574 | 0.669 | 0.833 |
| fcps/target.data.gz | 0.795 | 0.795 | 1.000 | 0.663 | 0.658 | 0.684 | 0.672 | 0.667 |
| fcps/tetra.data.gz | 1.000 | 0.975 | 0.494 | 0.990 | 0.995 | 1.000 | 0.786 | 0.975 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 | 0.706 | 0.993 | 0.997 | 0.997 | 0.827 | 0.997 |
| fcps/wingnut.data.gz | 1.000 | 0.800 | 1.000 | 0.748 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/dense.data.gz | 1.000 | 0.960 | 0.695 | 0.671 | 0.960 | 0.683 | 0.960 | 0.960 |
| graves/fuzzyx.data.gz | 0.758 | 0.713 | 0.447 | 0.730 | 0.795 | 0.701 | 0.663 | 0.496 |
| graves/line.data.gz | 0.603 | 0.643 | 1.000 | 0.606 | 0.603 | 0.597 | 0.609 | 0.603 |
| graves/parabolic.data.gz | 0.701 | 0.811 | 0.706 | 0.796 | 0.799 | 0.803 | 0.845 | 0.758 |
| graves/ring.data.gz | 0.650 | 0.647 | 1.000 | 0.660 | 0.647 | 0.651 | 0.655 | 0.660 |
| graves/zigzag.data.gz | 0.497 | 0.497 | 1.000 | 0.504 | 0.509 | 0.497 | 0.491 | 0.528 |
| other/iris.data.gz | 0.841 | 0.822 | 0.764 | 0.769 | 0.841 | 0.832 | 0.831 | 0.764 |
| other/iris5.data.gz | 0.765 | 0.738 | 0.691 | 0.665 | 0.765 | 0.753 | 0.750 | 0.691 |
| other/square.data.gz | 0.519 | 0.671 | 1.000 | 0.652 | 0.650 | 0.663 | 0.662 | 0.655 |
| sipu/a1.data.gz | 0.924 | 0.918 | 0.564 | 0.920 | 0.929 | 0.815 | 0.694 | 0.877 |
| sipu/a2.data.gz | 0.934 | 0.924 | 0.480 | 0.911 | 0.936 | 0.861 | 0.652 | 0.830 |
| sipu/a3.data.gz | 0.948 | 0.939 | 0.449 | 0.919 | 0.945 | 0.853 | 0.661 | 0.853 |
| sipu/aggregation.data.gz | 0.841 | 0.856 | 0.861 | 0.824 | 0.995 | 0.750 | 0.961 | 0.995 |
| sipu/compound.data.gz | 0.659 | 0.654 | 0.830 | 0.855 | 0.862 | 0.870 | 0.833 | 0.861 |
| sipu/d31.data.gz | 0.919 | 0.923 | 0.349 | 0.926 | 0.910 | 0.784 | 0.635 | 0.851 |
| sipu/flame.data.gz | 0.696 | 0.624 | 0.730 | 0.623 | 0.731 | 0.715 | 0.724 | 0.730 |
| sipu/jain.data.gz | 0.790 | 0.790 | 0.804 | 0.922 | 0.922 | 0.790 | 0.841 | 0.918 |
| sipu/pathbased.data.gz | 0.648 | 0.674 | 0.573 | 0.595 | 0.653 | 0.580 | 0.502 | 0.648 |
| sipu/r15.data.gz | 0.983 | 0.983 | 0.637 | 0.980 | 0.990 | 0.977 | 0.800 | 0.983 |
| sipu/s1.data.gz | 0.982 | 0.984 | 0.589 | 0.973 | 0.983 | 0.819 | 0.680 | 0.982 |
| sipu/s2.data.gz | 0.906 | 0.912 | 0.257 | 0.807 | 0.918 | 0.703 | 0.539 | 0.806 |
| sipu/s3.data.gz | 0.688 | 0.699 | 0.257 | 0.548 | 0.636 | 0.470 | 0.467 | 0.574 |
| sipu/s4.data.gz | 0.618 | 0.585 | 0.257 | 0.468 | 0.546 | 0.438 | 0.380 | 0.512 |
| sipu/spiral.data.gz | 0.337 | 0.337 | 1.000 | 0.338 | 0.357 | 0.346 | 0.332 | 0.386 |
| sipu/unbalance.data.gz | 1.000 | 1.000 | 0.999 | 0.775 | 1.000 | 0.775 | 0.775 | 1.000 |
| wut/cross.data.gz | 0.455 | 0.440 | 0.498 | 0.597 | 0.490 | 0.475 | 0.476 | 0.495 |
| wut/smile.data.gz | 0.583 | 0.594 | 1.000 | 0.647 | 0.690 | 0.624 | 0.626 | 0.713 |
| wut/twosplashes.data.gz | 0.607 | 0.602 | 0.701 | 0.609 | 0.690 | 0.602 | 0.639 | 0.690 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.783 | 0.543 | 0.571 | 0.551 | 0.587 | 0.510 | 0.557 | 0.597 |
| wut/x3.data.gz | 0.869 | 0.903 | 0.512 | 0.694 | 0.679 | 0.696 | 0.779 | 0.729 |
| wut/z1.data.gz | 0.466 | 0.464 | 0.540 | 0.457 | 0.511 | 0.474 | 0.474 | 0.516 |
| wut/z2.data.gz | 0.644 | 0.612 | 0.857 | 0.623 | 0.676 | 0.594 | 0.698 | 0.776 |
| wut/z3.data.gz | 0.997 | 1.000 | 0.842 | 0.954 | 0.998 | 0.957 | 0.997 | 1.000 |

4.1.1.2 Moje zbiory

Wyniki działania na moich zbiorach danych przedstawiają się następująco:

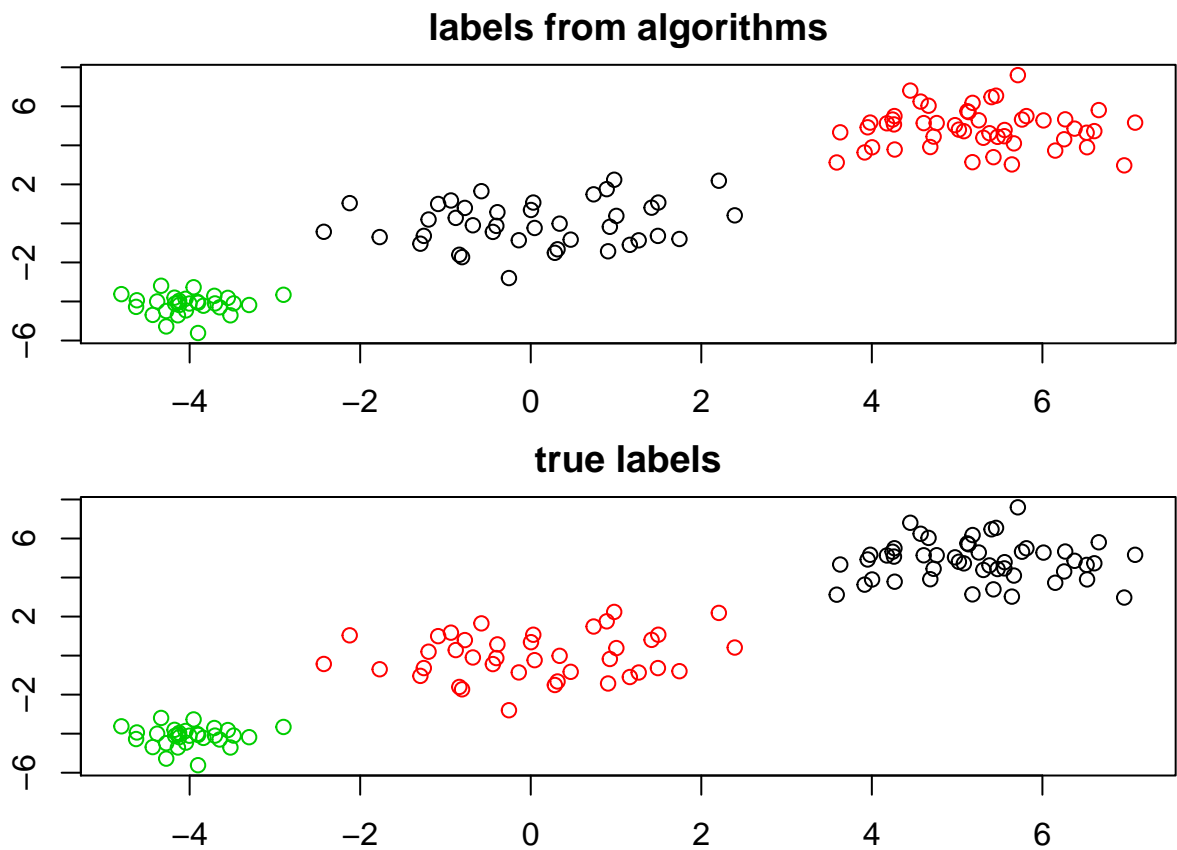
Wartości indeksów AM dla różnych algorytmów hierarchicznych są przedstawione poniżej:

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|-----------------|--------|---------|--------|----------|---------|----------|--------|----------|
| benchmark1.data | 0.739 | 0.529 | 0.000 | 0.482 | 0.000 | 0.138 | 0.033 | 0.000 |
| benchmark2.data | 0.986 | 0.987 | 0.774 | 0.984 | 0.996 | 0.989 | 0.868 | 0.998 |
| benchmark3.data | 0.927 | 0.917 | 0.460 | 0.894 | 0.904 | 0.834 | 0.845 | 0.917 |
| benchmark4.data | 0.382 | 0.372 | 1.000 | 0.337 | 0.378 | 0.359 | 0.275 | 0.295 |

Wartości indeksów FM dla różnych algorytmów hierarchicznych są przedstawione poniżej:

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|-----------------|--------|---------|--------|----------|---------|----------|--------|----------|
| benchmark1.data | 0.871 | 0.765 | 0.706 | 0.741 | 0.703 | 0.641 | 0.649 | 0.705 |
| benchmark2.data | 0.987 | 0.988 | 0.813 | 0.985 | 0.997 | 0.990 | 0.883 | 0.998 |
| benchmark3.data | 0.941 | 0.934 | 0.653 | 0.915 | 0.923 | 0.868 | 0.876 | 0.934 |
| benchmark4.data | 0.509 | 0.502 | 1.000 | 0.476 | 0.506 | 0.492 | 0.450 | 0.507 |

4.1.1.3 Przykładowy zbiór



4.1.2 Ze standaryzacją zmiennych

4.1.2.1 Zbiory dostarczone

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|--------------------------|--------|---------|--------|----------|---------|----------|--------|----------|
| fcps/atom.data.gz | 1.000 | 0.067 | 1.000 | 0.310 | 0.052 | 0.254 | 0.002 | 0.000 |
| fcps/chainlink.data.gz | -0.001 | -0.001 | 1.000 | -0.001 | 0.188 | 0.158 | 0.096 | 0.090 |
| fcps/engytime.data.gz | 0.824 | 0.808 | 0.000 | 0.105 | 0.010 | 0.048 | 0.001 | 0.000 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/lsun.data.gz | 0.378 | 0.385 | 1.000 | 0.367 | 0.398 | 0.421 | 0.428 | 0.340 |
| fcps/target.data.gz | 0.639 | 0.639 | 1.000 | 0.391 | 0.146 | 0.155 | 0.111 | 0.095 |
| fcps/tetra.data.gz | 1.000 | 0.967 | 0.000 | 0.987 | 0.993 | 0.987 | 0.671 | 0.967 |
| fcps/twodiamonds.data.gz | 0.985 | 1.000 | 0.000 | 0.005 | 0.970 | 0.085 | 0.001 | 0.003 |
| fcps/wingnut.data.gz | 1.000 | 1.000 | 1.000 | 0.482 | 1.000 | 0.491 | 0.635 | 0.319 |
| graves/dense.data.gz | 1.000 | 0.921 | 0.001 | 0.474 | 0.921 | 0.502 | 0.357 | 0.000 |
| graves/fuzzyx.data.gz | 0.699 | 0.664 | 0.000 | 0.599 | 0.743 | 0.499 | 0.510 | 0.270 |
| graves/line.data.gz | -0.064 | -0.104 | 1.000 | -0.071 | -0.066 | -0.017 | -0.078 | -0.066 |
| graves/parabolic.data.gz | 0.553 | 0.541 | 0.000 | 0.565 | 0.596 | 0.538 | 0.033 | 0.571 |
| graves/ring.data.gz | 0.222 | 0.116 | 1.000 | 0.209 | 0.114 | 0.159 | 0.040 | 0.029 |
| graves/zigzag.data.gz | 0.214 | 0.175 | 1.000 | 0.196 | 0.207 | 0.137 | -0.106 | 0.182 |
| other/iris.data.gz | 0.630 | 0.615 | 0.558 | 0.573 | 0.562 | 0.348 | 0.573 | 0.558 |
| other/iris5.data.gz | 0.324 | 0.322 | 0.150 | 0.271 | 0.459 | 0.393 | 0.147 | 0.147 |
| other/square.data.gz | 0.155 | 0.234 | 1.000 | 0.084 | 0.036 | 0.274 | 0.029 | 0.063 |
| sipu/a1.data.gz | 0.943 | 0.890 | 0.217 | 0.641 | 0.892 | 0.602 | 0.569 | 0.757 |
| sipu/a2.data.gz | 0.934 | 0.933 | 0.248 | 0.910 | 0.943 | 0.800 | 0.619 | 0.872 |
| sipu/a3.data.gz | 0.947 | 0.939 | 0.315 | 0.915 | 0.944 | 0.843 | 0.637 | 0.843 |
| sipu/aggregation.data.gz | 0.796 | 0.802 | 0.806 | 0.753 | 0.991 | 0.759 | 0.751 | 1.000 |
| sipu/compound.data.gz | 0.499 | 0.501 | 0.756 | 0.713 | 0.803 | 0.727 | 0.740 | 0.793 |
| sipu/d31.data.gz | 0.928 | 0.917 | 0.154 | 0.873 | 0.926 | 0.757 | 0.532 | 0.839 |
| sipu/flame.data.gz | 0.180 | 0.289 | 0.013 | 0.289 | 0.357 | 0.625 | 0.013 | 0.013 |
| sipu/jain.data.gz | 0.515 | 0.569 | 0.010 | 0.515 | 0.515 | 1.000 | 0.515 | 0.324 |
| sipu/pathbased.data.gz | 0.457 | 0.457 | 0.001 | 0.392 | 0.472 | 0.457 | 0.457 | 0.088 |
| sipu/r15.data.gz | 0.982 | 0.982 | 0.542 | 0.979 | 0.989 | 0.975 | 0.772 | 0.982 |
| sipu/s1.data.gz | 0.982 | 0.983 | 0.464 | 0.904 | 0.982 | 0.859 | 0.743 | 0.979 |
| sipu/s2.data.gz | 0.911 | 0.916 | 0.000 | 0.683 | 0.913 | 0.774 | 0.440 | 0.802 |
| sipu/s3.data.gz | 0.680 | 0.685 | 0.000 | 0.509 | 0.604 | 0.466 | 0.442 | 0.477 |
| sipu/s4.data.gz | 0.576 | 0.595 | 0.000 | 0.500 | 0.500 | 0.404 | 0.289 | 0.420 |
| sipu/spiral.data.gz | 0.000 | -0.004 | 1.000 | 0.003 | 0.020 | 0.008 | -0.005 | 0.012 |
| sipu/unbalance.data.gz | 0.743 | 1.000 | 0.612 | 0.611 | 0.612 | 0.611 | 0.611 | 0.611 |
| wut/cross.data.gz | 0.154 | 0.072 | 0.000 | 0.087 | 0.001 | 0.045 | 0.005 | 0.000 |
| wut/smile.data.gz | 0.446 | 0.455 | 1.000 | 0.474 | 0.471 | 0.470 | 0.574 | 0.606 |
| wut/twosplashes.data.gz | 0.301 | 0.301 | 0.000 | 0.007 | 0.018 | 0.007 | 0.005 | 0.018 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.669 | 0.191 | -0.007 | 0.209 | 0.191 | 0.156 | 0.005 | 0.271 |
| wut/x3.data.gz | 0.821 | 0.869 | 0.007 | 0.774 | 0.505 | 0.701 | 0.621 | 0.562 |
| wut/z1.data.gz | 0.079 | 0.080 | 0.000 | 0.249 | 0.080 | 0.247 | 0.114 | 0.044 |
| wut/z2.data.gz | 0.535 | 0.342 | 0.735 | 0.303 | 0.403 | 0.401 | 0.282 | 0.476 |
| wut/z3.data.gz | 0.995 | 0.995 | 0.739 | 0.981 | 0.959 | 0.940 | 0.983 | 0.995 |

Wartości indeksów FM dla różnych algorytmów hierarchicznych:

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|-----|--------|---------|--------|----------|---------|----------|--------|----------|
|-----|--------|---------|--------|----------|---------|----------|--------|----------|

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|--------------------------|--------|---------|--------|----------|---------|----------|--------|----------|
| fcps/atom.data.gz | 1.000 | 0.649 | 1.000 | 0.688 | 0.651 | 0.671 | 0.692 | 0.702 |
| fcps/chainlink.data.gz | 0.509 | 0.500 | 1.000 | 0.504 | 0.656 | 0.651 | 0.646 | 0.647 |
| fcps/engytime.data.gz | 0.912 | 0.904 | 0.707 | 0.629 | 0.674 | 0.582 | 0.698 | 0.702 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/lsun.data.gz | 0.613 | 0.617 | 1.000 | 0.604 | 0.623 | 0.636 | 0.652 | 0.595 |
| fcps/target.data.gz | 0.795 | 0.795 | 1.000 | 0.679 | 0.658 | 0.658 | 0.661 | 0.665 |
| fcps/tetra.data.gz | 1.000 | 0.975 | 0.494 | 0.990 | 0.995 | 0.990 | 0.786 | 0.975 |
| fcps/twodiamonds.data.gz | 0.993 | 1.000 | 0.706 | 0.579 | 0.985 | 0.595 | 0.663 | 0.688 |
| fcps/wingnut.data.gz | 1.000 | 1.000 | 1.000 | 0.753 | 1.000 | 0.745 | 0.818 | 0.691 |
| graves/dense.data.gz | 1.000 | 0.960 | 0.695 | 0.748 | 0.960 | 0.760 | 0.703 | 0.702 |
| graves/fuzzxyx.data.gz | 0.762 | 0.734 | 0.447 | 0.690 | 0.795 | 0.619 | 0.648 | 0.496 |
| graves/line.data.gz | 0.603 | 0.643 | 1.000 | 0.606 | 0.603 | 0.597 | 0.609 | 0.603 |
| graves/parabolic.data.gz | 0.777 | 0.771 | 0.706 | 0.783 | 0.798 | 0.769 | 0.663 | 0.786 |
| graves/ring.data.gz | 0.663 | 0.647 | 1.000 | 0.660 | 0.647 | 0.651 | 0.655 | 0.660 |
| graves/zigzag.data.gz | 0.521 | 0.511 | 1.000 | 0.524 | 0.547 | 0.493 | 0.476 | 0.510 |
| other/iris.data.gz | 0.762 | 0.750 | 0.764 | 0.728 | 0.760 | 0.597 | 0.728 | 0.764 |
| other/iris5.data.gz | 0.594 | 0.666 | 0.703 | 0.599 | 0.716 | 0.683 | 0.678 | 0.684 |
| other/square.data.gz | 0.650 | 0.666 | 1.000 | 0.568 | 0.657 | 0.677 | 0.660 | 0.649 |
| sipu/a1.data.gz | 0.946 | 0.896 | 0.404 | 0.666 | 0.898 | 0.629 | 0.611 | 0.785 |
| sipu/a2.data.gz | 0.936 | 0.935 | 0.405 | 0.913 | 0.945 | 0.807 | 0.644 | 0.878 |
| sipu/a3.data.gz | 0.948 | 0.940 | 0.449 | 0.917 | 0.945 | 0.847 | 0.659 | 0.853 |
| sipu/aggregation.data.gz | 0.843 | 0.845 | 0.863 | 0.806 | 0.993 | 0.811 | 0.809 | 1.000 |
| sipu/compound.data.gz | 0.612 | 0.613 | 0.835 | 0.788 | 0.861 | 0.802 | 0.816 | 0.856 |
| sipu/d31.data.gz | 0.931 | 0.919 | 0.333 | 0.877 | 0.928 | 0.768 | 0.588 | 0.851 |
| sipu/flame.data.gz | 0.621 | 0.662 | 0.730 | 0.662 | 0.691 | 0.821 | 0.730 | 0.730 |
| sipu/jain.data.gz | 0.790 | 0.816 | 0.784 | 0.790 | 0.790 | 1.000 | 0.790 | 0.700 |
| sipu/pathbased.data.gz | 0.660 | 0.660 | 0.572 | 0.597 | 0.666 | 0.660 | 0.660 | 0.529 |
| sipu/r15.data.gz | 0.983 | 0.983 | 0.637 | 0.980 | 0.990 | 0.977 | 0.800 | 0.983 |
| sipu/s1.data.gz | 0.983 | 0.984 | 0.589 | 0.912 | 0.983 | 0.870 | 0.768 | 0.981 |
| sipu/s2.data.gz | 0.917 | 0.922 | 0.257 | 0.709 | 0.919 | 0.791 | 0.505 | 0.822 |
| sipu/s3.data.gz | 0.702 | 0.706 | 0.257 | 0.549 | 0.644 | 0.508 | 0.503 | 0.566 |
| sipu/s4.data.gz | 0.604 | 0.622 | 0.257 | 0.541 | 0.552 | 0.454 | 0.388 | 0.511 |
| sipu/spiral.data.gz | 0.336 | 0.348 | 1.000 | 0.363 | 0.358 | 0.337 | 0.330 | 0.505 |
| sipu/unbalance.data.gz | 0.818 | 1.000 | 0.775 | 0.774 | 0.775 | 0.774 | 0.774 | 0.774 |
| wut/cross.data.gz | 0.455 | 0.440 | 0.498 | 0.444 | 0.490 | 0.475 | 0.485 | 0.495 |
| wut/smile.data.gz | 0.589 | 0.596 | 1.000 | 0.618 | 0.622 | 0.616 | 0.689 | 0.714 |
| wut/twosplashes.data.gz | 0.649 | 0.649 | 0.701 | 0.679 | 0.667 | 0.679 | 0.684 | 0.667 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.783 | 0.543 | 0.571 | 0.551 | 0.543 | 0.528 | 0.518 | 0.597 |
| wut/x3.data.gz | 0.869 | 0.903 | 0.512 | 0.835 | 0.679 | 0.784 | 0.764 | 0.724 |
| wut/z1.data.gz | 0.417 | 0.418 | 0.568 | 0.502 | 0.418 | 0.504 | 0.428 | 0.394 |
| wut/z2.data.gz | 0.688 | 0.541 | 0.857 | 0.535 | 0.596 | 0.603 | 0.522 | 0.752 |
| wut/z3.data.gz | 0.997 | 0.997 | 0.842 | 0.987 | 0.972 | 0.959 | 0.988 | 0.997 |

4.1.2.2 Moje zbiory

Wyniki działania na moich zbiorach danych przedstawiają się następująco:

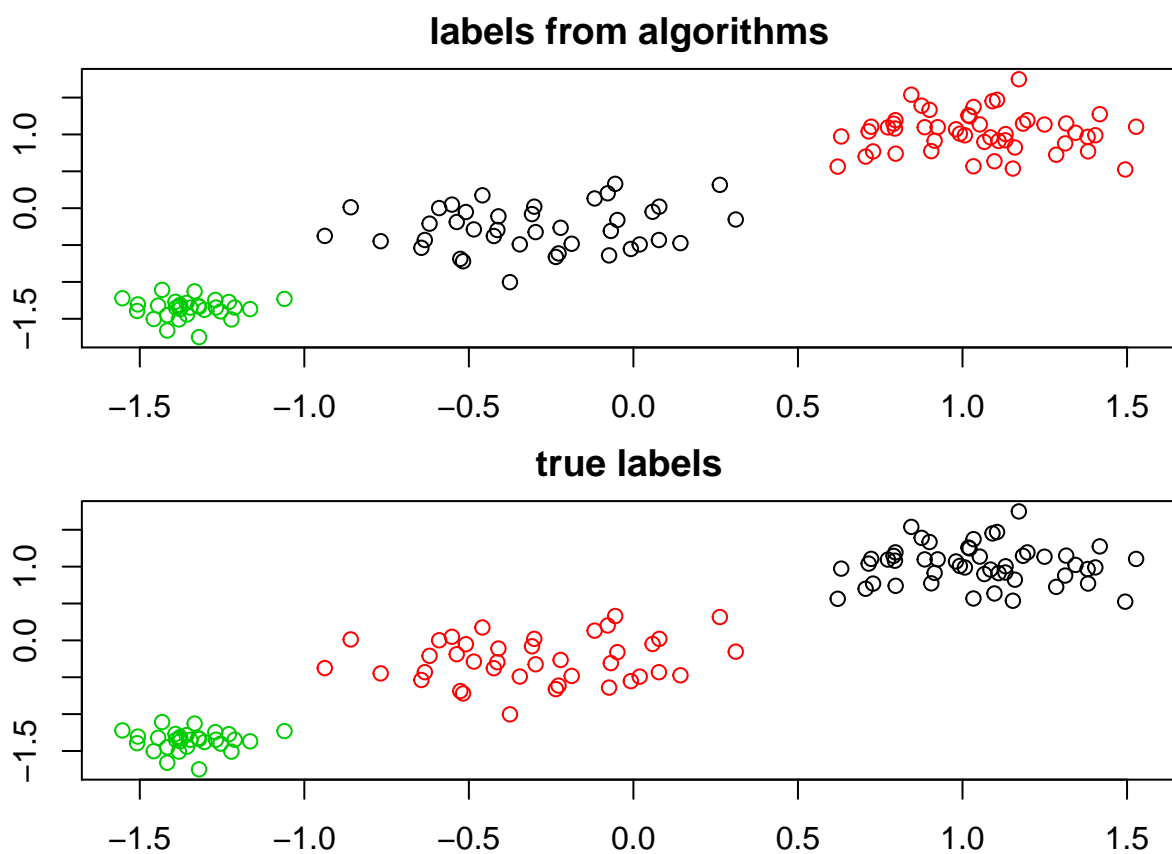
Wartości indeksów AM dla różnych algorytmów hierarchicznych są przedstawione poniżej:

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|-----------------|--------|---------|--------|----------|---------|----------|--------|----------|
| benchmark1.data | 1.000 | 1.000 | 1.000 | 0.668 | 1.000 | 0.874 | 0.000 | 1.000 |
| benchmark2.data | 0.993 | 0.998 | 0.838 | 0.996 | 0.993 | 0.993 | 0.993 | 0.987 |
| benchmark3.data | 0.927 | 0.922 | 0.459 | 0.838 | 0.728 | 0.713 | 0.837 | 0.913 |
| benchmark4.data | 0.273 | 0.199 | 0.483 | 0.208 | 0.257 | 0.257 | 0.058 | 0.000 |

Wartości indeksów FM dla różnych algorytmów hierarchicznych są przedstawione poniżej:

| set | ward.D | ward.D2 | single | complete | average | mcquitty | median | centroid |
|-----------------|--------|---------|--------|----------|---------|----------|--------|----------|
| benchmark1.data | 1.000 | 1.000 | 1.000 | 0.837 | 1.000 | 0.937 | 0.703 | 1.000 |
| benchmark2.data | 0.993 | 0.998 | 0.862 | 0.997 | 0.993 | 0.993 | 0.993 | 0.988 |
| benchmark3.data | 0.941 | 0.937 | 0.651 | 0.871 | 0.800 | 0.787 | 0.869 | 0.930 |
| benchmark4.data | 0.422 | 0.363 | 0.672 | 0.380 | 0.428 | 0.420 | 0.419 | 0.322 |

4.1.2.3 Przykładowy zbiór



5 Algorytm *Genie* z pakietu *genie*

5.1 Badanie algorytmu na wszystkich zbiorach benchmarkowych

5.1.1 Bez standaryzacji zmiennych

5.1.1.1 Zbiory dostaczone

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|--------------------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| fcps/atom.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/chainlink.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/engytime.data.gz | 0.803 | 0.803 | 0.803 | 0.803 | 0.803 | 0.007 | 0.007 | 0.007 | 0.002 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/lsun.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/target.data.gz | 0.357 | 0.357 | 0.419 | 0.638 | 0.638 | 0.687 | 0.705 | 1.000 | 1.000 |
| fcps/tetra.data.gz | 0.993 | 0.993 | 0.993 | 0.993 | 0.993 | 0.993 | 0.993 | 0.709 | 0.709 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.000 |
| fcps/wingnut.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/dense.data.gz | 0.960 | 0.960 | 0.960 | 0.960 | 0.960 | 0.960 | 0.960 | 0.001 | 0.001 |
| graves/fuzzzyx.data.gz | 0.539 | 0.431 | 0.431 | 0.298 | 0.298 | 0.288 | 0.073 | 0.073 | 0.029 |
| graves/line.data.gz | 0.015 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/parabolic.data.gz | 0.732 | 0.732 | 0.732 | 0.732 | 0.732 | 0.000 | 0.000 | 0.000 | 0.000 |
| graves/ring.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/zigzag.data.gz | 0.469 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| other/iris.data.gz | 0.690 | 0.690 | 0.690 | 0.690 | 0.560 | 0.560 | 0.561 | 0.561 | 0.561 |
| other/iris5.data.gz | 0.297 | 0.496 | 0.496 | 0.496 | 0.148 | 0.148 | 0.148 | 0.148 | 0.147 |
| other/square.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| sipu/a1.data.gz | 0.929 | 0.929 | 0.929 | 0.896 | 0.897 | 0.792 | 0.632 | 0.364 | 0.329 |
| sipu/a2.data.gz | 0.942 | 0.942 | 0.914 | 0.893 | 0.831 | 0.680 | 0.558 | 0.420 | 0.353 |
| sipu/a3.data.gz | 0.956 | 0.956 | 0.918 | 0.902 | 0.856 | 0.710 | 0.553 | 0.413 | 0.347 |
| sipu/aggregation.data.gz | 0.592 | 0.592 | 0.540 | 0.690 | 0.912 | 0.855 | 0.876 | 0.799 | 0.806 |
| sipu/compound.data.gz | 0.448 | 0.542 | 0.542 | 0.564 | 0.867 | 0.811 | 0.769 | 0.760 | 0.742 |
| sipu/d31.data.gz | 0.932 | 0.932 | 0.903 | 0.876 | 0.694 | 0.651 | 0.546 | 0.310 | 0.154 |
| sipu/flame.data.gz | 0.967 | 0.967 | 0.967 | 0.967 | 0.967 | 0.967 | 0.967 | 0.967 | 0.013 |
| sipu/jain.data.gz | 0.044 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.246 | 0.246 | 0.108 |
| sipu/pathbased.data.gz | 0.613 | 0.613 | 0.613 | 0.613 | 0.613 | 0.613 | 0.613 | 0.557 | 0.557 |
| sipu/r15.data.gz | 0.989 | 0.989 | 0.989 | 0.989 | 0.914 | 0.548 | 0.548 | 0.548 | 0.548 |
| sipu/s1.data.gz | 0.988 | 0.988 | 0.988 | 0.988 | 0.988 | 0.988 | 0.859 | 0.586 | 0.531 |
| sipu/s2.data.gz | 0.924 | 0.924 | 0.924 | 0.784 | 0.783 | 0.735 | 0.472 | 0.379 | 0.143 |
| sipu/s3.data.gz | 0.700 | 0.700 | 0.580 | 0.583 | 0.555 | 0.488 | 0.421 | 0.182 | 0.081 |
| sipu/s4.data.gz | 0.611 | 0.611 | 0.550 | 0.442 | 0.403 | 0.321 | 0.217 | 0.168 | 0.034 |
| sipu/spiral.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| sipu/unbalance.data.gz | 0.541 | 0.591 | 0.668 | 0.695 | 0.789 | 0.869 | 1.000 | 1.000 | 1.000 |
| wut/cross.data.gz | 0.753 | 0.753 | 0.461 | 0.312 | 0.103 | 0.103 | 0.064 | 0.027 | 0.006 |
| wut/smile.data.gz | 0.446 | 0.701 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/twosplashes.data.gz | 0.238 | 0.238 | 0.238 | 0.238 | 0.238 | 0.238 | 0.002 | 0.002 | 0.002 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.688 | 0.688 | 0.688 | 0.688 | 0.221 | 0.221 | 0.221 | 0.261 | -0.014 |
| wut/x3.data.gz | 0.871 | 0.871 | 0.484 | 0.484 | 0.484 | 0.320 | 0.320 | 0.222 | 0.194 |
| wut/z1.data.gz | 0.340 | 0.340 | 0.340 | 0.340 | 0.340 | 0.340 | 0.340 | 0.340 | 0.340 |
| wut/z2.data.gz | 0.440 | 0.440 | 0.539 | 0.822 | 0.822 | 1.000 | 1.000 | 1.000 | 0.735 |
| wut/z3.data.gz | 0.464 | 0.464 | 0.755 | 0.755 | 0.772 | 0.772 | 0.731 | 0.733 | 0.742 |

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|--------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| fcps/atom.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/chainlink.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/engytime.data.gz | 0.902 | 0.902 | 0.902 | 0.902 | 0.902 | 0.680 | 0.680 | 0.680 | 0.691 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/lsun.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/target.data.gz | 0.591 | 0.591 | 0.641 | 0.794 | 0.794 | 0.825 | 0.836 | 1.000 | 1.000 |
| fcps/tetra.data.gz | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.812 | 0.812 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.706 |
| fcps/wingnut.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/dense.data.gz | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 0.692 | 0.692 |
| graves/fuzzzyx.data.gz | 0.648 | 0.571 | 0.571 | 0.503 | 0.503 | 0.514 | 0.423 | 0.423 | 0.427 |
| graves/line.data.gz | 0.601 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/parabolic.data.gz | 0.866 | 0.866 | 0.866 | 0.866 | 0.866 | 0.689 | 0.689 | 0.689 | 0.689 |
| graves/ring.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/zigzag.data.gz | 0.682 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| other/iris.data.gz | 0.793 | 0.793 | 0.793 | 0.793 | 0.754 | 0.754 | 0.757 | 0.757 | 0.757 |
| other/iris5.data.gz | 0.573 | 0.706 | 0.706 | 0.706 | 0.616 | 0.616 | 0.616 | 0.673 | 0.678 |
| other/square.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| sipu/a1.data.gz | 0.932 | 0.932 | 0.932 | 0.902 | 0.903 | 0.814 | 0.693 | 0.508 | 0.484 |
| sipu/a2.data.gz | 0.943 | 0.943 | 0.917 | 0.897 | 0.843 | 0.719 | 0.629 | 0.533 | 0.484 |
| sipu/a3.data.gz | 0.957 | 0.957 | 0.920 | 0.905 | 0.863 | 0.742 | 0.622 | 0.522 | 0.473 |
| sipu/aggregation.data.gz | 0.675 | 0.675 | 0.636 | 0.763 | 0.931 | 0.891 | 0.909 | 0.856 | 0.863 |
| sipu/compound.data.gz | 0.569 | 0.646 | 0.646 | 0.669 | 0.904 | 0.866 | 0.841 | 0.837 | 0.830 |
| sipu/d31.data.gz | 0.934 | 0.934 | 0.906 | 0.881 | 0.725 | 0.694 | 0.620 | 0.451 | 0.333 |
| sipu/flame.data.gz | 0.985 | 0.985 | 0.985 | 0.985 | 0.985 | 0.985 | 0.985 | 0.985 | 0.730 |
| sipu/jain.data.gz | 0.592 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.803 | 0.803 | 0.789 |
| sipu/pathbased.data.gz | 0.751 | 0.751 | 0.751 | 0.751 | 0.751 | 0.751 | 0.751 | 0.767 | 0.767 |
| sipu/r15.data.gz | 0.990 | 0.990 | 0.990 | 0.990 | 0.922 | 0.643 | 0.643 | 0.643 | 0.643 |
| sipu/s1.data.gz | 0.989 | 0.989 | 0.989 | 0.989 | 0.989 | 0.989 | 0.875 | 0.671 | 0.635 |
| sipu/s2.data.gz | 0.929 | 0.929 | 0.929 | 0.808 | 0.808 | 0.771 | 0.584 | 0.525 | 0.366 |
| sipu/s3.data.gz | 0.720 | 0.720 | 0.621 | 0.625 | 0.604 | 0.565 | 0.529 | 0.375 | 0.310 |
| sipu/s4.data.gz | 0.637 | 0.637 | 0.586 | 0.508 | 0.488 | 0.438 | 0.380 | 0.357 | 0.267 |
| sipu/spiral.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| sipu/unbalance.data.gz | 0.668 | 0.706 | 0.762 | 0.782 | 0.849 | 0.906 | 1.000 | 1.000 | 1.000 |
| wut/cross.data.gz | 0.817 | 0.817 | 0.622 | 0.543 | 0.440 | 0.440 | 0.454 | 0.463 | 0.466 |
| wut/smile.data.gz | 0.589 | 0.786 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/twosplashes.data.gz | 0.620 | 0.620 | 0.620 | 0.620 | 0.620 | 0.620 | 0.690 | 0.690 | 0.690 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.794 | 0.794 | 0.794 | 0.794 | 0.559 | 0.559 | 0.559 | 0.589 | 0.553 |
| wut/x3.data.gz | 0.905 | 0.905 | 0.664 | 0.664 | 0.664 | 0.591 | 0.591 | 0.582 | 0.576 |
| wut/z1.data.gz | 0.592 | 0.592 | 0.592 | 0.592 | 0.592 | 0.592 | 0.592 | 0.592 | 0.592 |
| wut/z2.data.gz | 0.618 | 0.618 | 0.693 | 0.893 | 0.893 | 1.000 | 1.000 | 1.000 | 0.857 |
| wut/z3.data.gz | 0.617 | 0.617 | 0.830 | 0.830 | 0.851 | 0.851 | 0.836 | 0.838 | 0.844 |

5.1.1.2 Moje zbiory

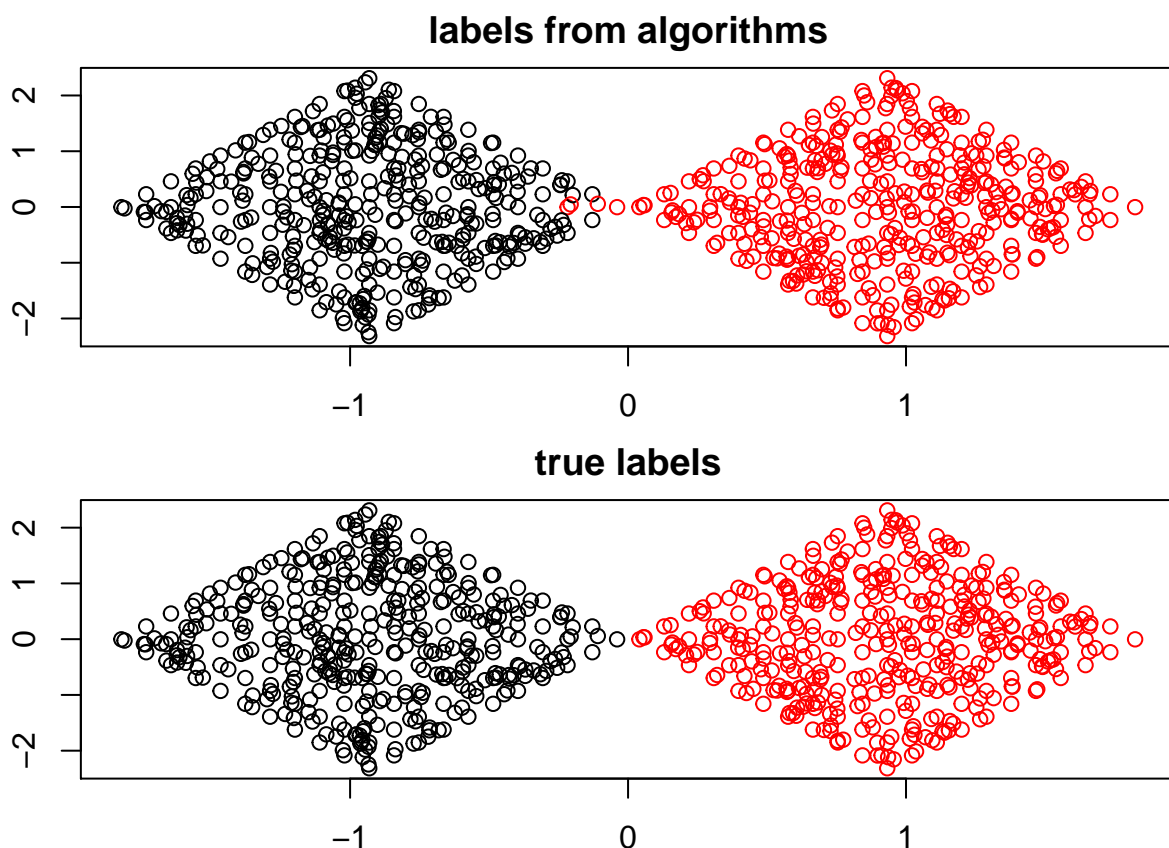
Wyniki działania na moich zbiorach danych przedstawiają się następująco:

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.990 | 0.990 | 0.990 | 0.990 | 0.990 | 0.990 | 0.990 | 0.990 | 0.000 |
| benchmark2.data | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.914 | 0.913 | 0.913 |

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark3.data | 0.927 | 0.927 | 0.927 | 0.927 | 0.719 | 0.719 | 0.743 | 0.734 | 0.734 |
| benchmark4.data | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.705 |
| benchmark2.data | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.924 | 0.923 | 0.923 |
| benchmark3.data | 0.941 | 0.941 | 0.941 | 0.941 | 0.793 | 0.793 | 0.813 | 0.806 | 0.806 |
| benchmark4.data | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |

5.1.1.3 przykładowy zbiór



5.1.2 Ze standaryzacją zmiennych

5.1.2.1 Zbiory dostarczone

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|--------------------------|-------|-------|-------|-------|-------|--------|-------|-------|--------|
| fcps/atom.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/chainlink.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/engytime.data.gz | 0.808 | 0.808 | 0.808 | 0.808 | 0.808 | 0.008 | 0.008 | 0.008 | 0.001 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/lsun.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/target.data.gz | 0.357 | 0.357 | 0.419 | 0.638 | 0.638 | 0.687 | 0.738 | 1.000 | 1.000 |
| fcps/tetra.data.gz | 0.993 | 0.993 | 0.993 | 0.993 | 0.993 | 0.993 | 0.993 | 0.709 | 0.709 |
| fcps/twodiamonds.data.gz | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 0.000 | 0.000 | 0.000 |
| fcps/wingnut.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.000 |
| graves/dense.data.gz | 0.960 | 0.960 | 0.960 | 0.960 | 0.960 | 0.960 | 0.960 | 0.001 | 0.001 |
| graves/fuzzix.data.gz | 0.539 | 0.431 | 0.431 | 0.467 | 0.329 | 0.288 | 0.073 | 0.073 | 0.029 |
| graves/line.data.gz | 0.015 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/parabolic.data.gz | 0.732 | 0.732 | 0.732 | 0.732 | 0.732 | 0.732 | 0.001 | 0.001 | 0.001 |
| graves/ring.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/zigzag.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| other/iris.data.gz | 0.706 | 0.644 | 0.644 | 0.644 | 0.644 | 0.560 | 0.560 | 0.558 | 0.558 |
| other/iris5.data.gz | 0.327 | 0.247 | 0.247 | 0.247 | 0.314 | 0.164 | 0.151 | 0.151 | 0.148 |
| other/square.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| sipu/a1.data.gz | 0.915 | 0.853 | 0.858 | 0.826 | 0.757 | 0.678 | 0.678 | 0.483 | 0.208 |
| sipu/a2.data.gz | 0.943 | 0.943 | 0.896 | 0.864 | 0.837 | 0.564 | 0.502 | 0.443 | 0.362 |
| sipu/a3.data.gz | 0.956 | 0.956 | 0.918 | 0.902 | 0.856 | 0.710 | 0.553 | 0.413 | 0.347 |
| sipu/aggregation.data.gz | 0.545 | 0.507 | 0.507 | 0.685 | 0.912 | 0.855 | 0.781 | 0.806 | 0.806 |
| sipu/compound.data.gz | 0.518 | 0.538 | 0.514 | 0.514 | 0.618 | 0.862 | 0.777 | 0.748 | 0.750 |
| sipu/d31.data.gz | 0.933 | 0.933 | 0.899 | 0.873 | 0.712 | 0.624 | 0.487 | 0.322 | 0.155 |
| sipu/flame.data.gz | 0.967 | 0.967 | 0.967 | 0.967 | 0.967 | 0.967 | 0.967 | 0.967 | 0.013 |
| sipu/jain.data.gz | 0.044 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.020 |
| sipu/pathbased.data.gz | 0.667 | 0.667 | 0.667 | 0.667 | 0.667 | 0.667 | 0.524 | 0.552 | 0.001 |
| sipu/r15.data.gz | 0.989 | 0.989 | 0.989 | 0.989 | 0.914 | 0.548 | 0.548 | 0.548 | 0.548 |
| sipu/s1.data.gz | 0.988 | 0.988 | 0.988 | 0.988 | 0.988 | 0.988 | 0.859 | 0.586 | 0.464 |
| sipu/s2.data.gz | 0.913 | 0.913 | 0.913 | 0.773 | 0.770 | 0.730 | 0.466 | 0.338 | 0.102 |
| sipu/s3.data.gz | 0.698 | 0.660 | 0.580 | 0.583 | 0.554 | 0.484 | 0.421 | 0.182 | 0.082 |
| sipu/s4.data.gz | 0.606 | 0.606 | 0.584 | 0.462 | 0.407 | 0.319 | 0.256 | 0.171 | 0.035 |
| sipu/spiral.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| sipu/unbalance.data.gz | 0.565 | 0.644 | 0.635 | 0.627 | 0.787 | 0.897 | 1.000 | 1.000 | 1.000 |
| wut/cross.data.gz | 0.753 | 0.753 | 0.461 | 0.312 | 0.103 | 0.103 | 0.064 | 0.027 | 0.006 |
| wut/smile.data.gz | 0.448 | 0.701 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/twosplashes.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.000 | 0.000 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.688 | 0.688 | 0.688 | 0.688 | 0.221 | 0.221 | 0.221 | 0.261 | -0.014 |
| wut/x3.data.gz | 0.706 | 0.706 | 0.485 | 0.485 | 0.485 | 0.320 | 0.320 | 0.194 | 0.194 |
| wut/z1.data.gz | 0.055 | 0.055 | 0.055 | 0.055 | 0.055 | -0.007 | 0.000 | 0.000 | 0.000 |
| wut/z2.data.gz | 0.519 | 0.519 | 0.519 | 0.831 | 0.831 | 1.000 | 1.000 | 1.000 | 0.480 |
| wut/z3.data.gz | 0.649 | 0.649 | 0.929 | 0.929 | 0.929 | 0.728 | 0.733 | 0.733 | 0.739 |

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| fcps/atom.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/chainlink.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|--------------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| fcps/engytime.data.gz | 0.904 | 0.904 | 0.904 | 0.904 | 0.904 | 0.679 | 0.679 | 0.679 | 0.695 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/lsun.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/target.data.gz | 0.592 | 0.592 | 0.641 | 0.794 | 0.794 | 0.825 | 0.856 | 1.000 | 1.000 |
| fcps/tetra.data.gz | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.812 | 0.812 |
| fcps/twodiamonds.data.gz | 0.990 | 0.990 | 0.990 | 0.990 | 0.990 | 0.990 | 0.700 | 0.700 | 0.700 |
| fcps/wingnut.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.704 |
| graves/dense.data.gz | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 0.980 | 0.692 | 0.692 |
| graves/fuzzzyx.data.gz | 0.648 | 0.571 | 0.571 | 0.609 | 0.531 | 0.514 | 0.423 | 0.423 | 0.427 |
| graves/line.data.gz | 0.601 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/parabolic.data.gz | 0.866 | 0.866 | 0.866 | 0.866 | 0.866 | 0.866 | 0.688 | 0.688 | 0.688 |
| graves/ring.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| graves/zigzag.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| other/iris.data.gz | 0.806 | 0.772 | 0.772 | 0.772 | 0.772 | 0.749 | 0.749 | 0.764 | 0.764 |
| other/iris5.data.gz | 0.594 | 0.579 | 0.579 | 0.579 | 0.639 | 0.600 | 0.663 | 0.663 | 0.691 |
| other/square.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| sipu/a1.data.gz | 0.920 | 0.863 | 0.867 | 0.842 | 0.785 | 0.725 | 0.727 | 0.589 | 0.397 |
| sipu/a2.data.gz | 0.944 | 0.944 | 0.900 | 0.870 | 0.848 | 0.632 | 0.590 | 0.547 | 0.491 |
| sipu/a3.data.gz | 0.957 | 0.957 | 0.920 | 0.905 | 0.863 | 0.742 | 0.622 | 0.522 | 0.473 |
| sipu/aggregation.data.gz | 0.636 | 0.607 | 0.607 | 0.759 | 0.931 | 0.891 | 0.841 | 0.863 | 0.863 |
| sipu/compound.data.gz | 0.626 | 0.643 | 0.629 | 0.629 | 0.709 | 0.901 | 0.845 | 0.832 | 0.832 |
| sipu/d31.data.gz | 0.936 | 0.936 | 0.903 | 0.879 | 0.740 | 0.674 | 0.576 | 0.460 | 0.333 |
| sipu/flame.data.gz | 0.985 | 0.985 | 0.985 | 0.985 | 0.985 | 0.985 | 0.985 | 0.985 | 0.730 |
| sipu/jain.data.gz | 0.592 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.784 |
| sipu/pathbased.data.gz | 0.781 | 0.781 | 0.781 | 0.781 | 0.781 | 0.781 | 0.747 | 0.763 | 0.572 |
| sipu/r15.data.gz | 0.990 | 0.990 | 0.990 | 0.990 | 0.922 | 0.643 | 0.643 | 0.643 | 0.643 |
| sipu/s1.data.gz | 0.989 | 0.989 | 0.989 | 0.989 | 0.989 | 0.989 | 0.875 | 0.671 | 0.589 |
| sipu/s2.data.gz | 0.919 | 0.919 | 0.919 | 0.798 | 0.797 | 0.767 | 0.580 | 0.497 | 0.336 |
| sipu/s3.data.gz | 0.719 | 0.685 | 0.620 | 0.624 | 0.604 | 0.560 | 0.528 | 0.375 | 0.311 |
| sipu/s4.data.gz | 0.632 | 0.632 | 0.614 | 0.525 | 0.492 | 0.439 | 0.402 | 0.358 | 0.266 |
| sipu/spiral.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| sipu/unbalance.data.gz | 0.687 | 0.744 | 0.738 | 0.732 | 0.849 | 0.926 | 1.000 | 1.000 | 1.000 |
| wut/cross.data.gz | 0.817 | 0.817 | 0.622 | 0.543 | 0.440 | 0.440 | 0.454 | 0.463 | 0.466 |
| wut/smile.data.gz | 0.590 | 0.786 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/twosplashes.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.701 | 0.701 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.794 | 0.794 | 0.794 | 0.794 | 0.559 | 0.559 | 0.559 | 0.589 | 0.553 |
| wut/x3.data.gz | 0.787 | 0.787 | 0.665 | 0.665 | 0.665 | 0.591 | 0.591 | 0.576 | 0.576 |
| wut/z1.data.gz | 0.404 | 0.404 | 0.404 | 0.404 | 0.404 | 0.397 | 0.562 | 0.562 | 0.562 |
| wut/z2.data.gz | 0.676 | 0.676 | 0.676 | 0.892 | 0.892 | 1.000 | 1.000 | 1.000 | 0.754 |
| wut/z3.data.gz | 0.749 | 0.749 | 0.950 | 0.950 | 0.950 | 0.834 | 0.838 | 0.838 | 0.842 |

5.1.2.2 Moje zbiory

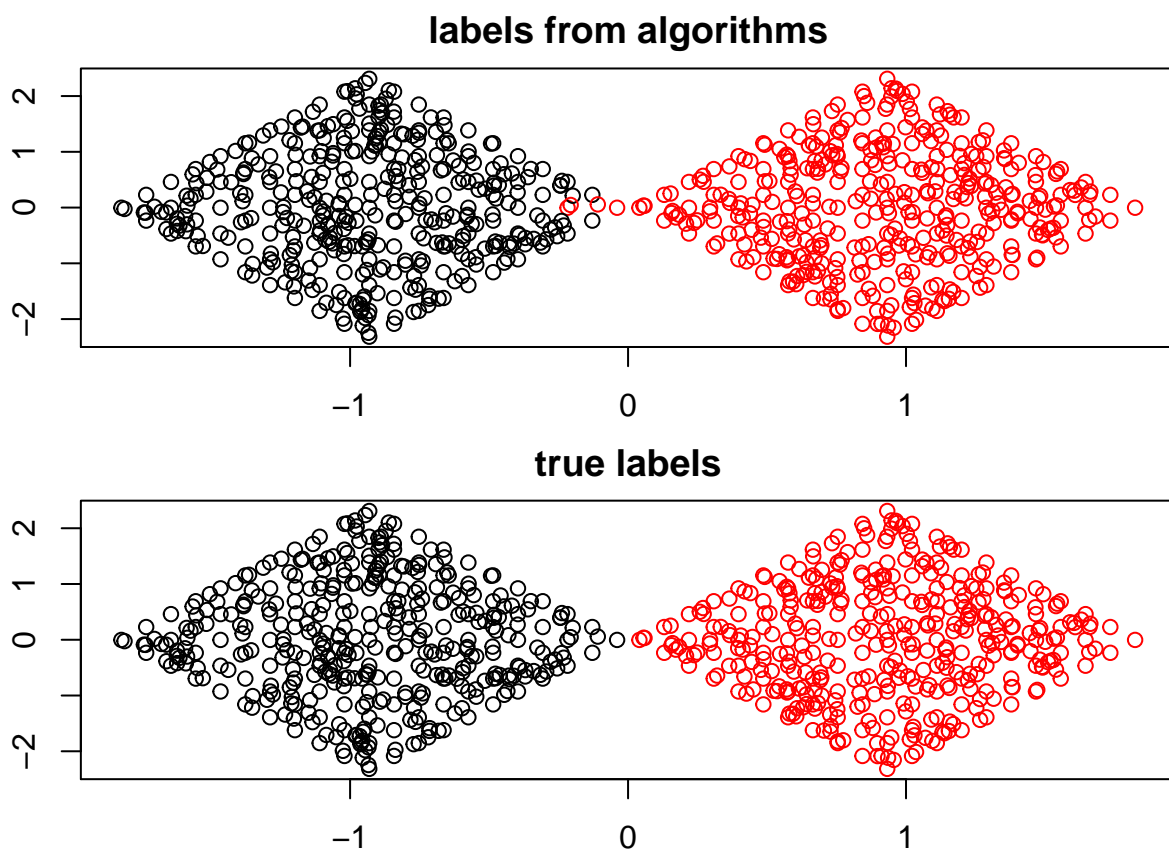
Wyniki działania na moich zbiorach danych przedstawiają się następująco:

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| benchmark2.data | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.910 | 0.838 | 0.839 | 0.839 |
| benchmark3.data | 0.931 | 0.931 | 0.931 | 0.699 | 0.721 | 0.721 | 0.729 | 0.459 | 0.459 |
| benchmark4.data | 0.995 | 0.995 | 0.995 | 0.995 | 0.995 | 0.781 | 0.612 | 0.612 | 0.612 |

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| | NA | NA | NA | NA | NA | NA | NA | NA | NA |

| set | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 | 0.9 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| benchmark2.data | 0.998 | 0.998 | 0.998 | 0.998 | 0.998 | 0.920 | 0.862 | 0.863 | 0.863 |
| benchmark3.data | 0.945 | 0.945 | 0.945 | 0.776 | 0.795 | 0.795 | 0.802 | 0.652 | 0.652 |
| benchmark4.data | 0.996 | 0.996 | 0.996 | 0.996 | 0.996 | 0.843 | 0.741 | 0.741 | 0.741 |
| | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| | NA | NA | NA | NA | NA | NA | NA | NA | NA |

5.1.2.3 przykładowy zbiór



6 Algorytm *HCPC*

6.1 Badanie algorytmu na wszystkich zbiorach benchmarkowych

6.1.1 Bez standaryzacji zmiennych

6.1.1.1 Zbiory dostarczone

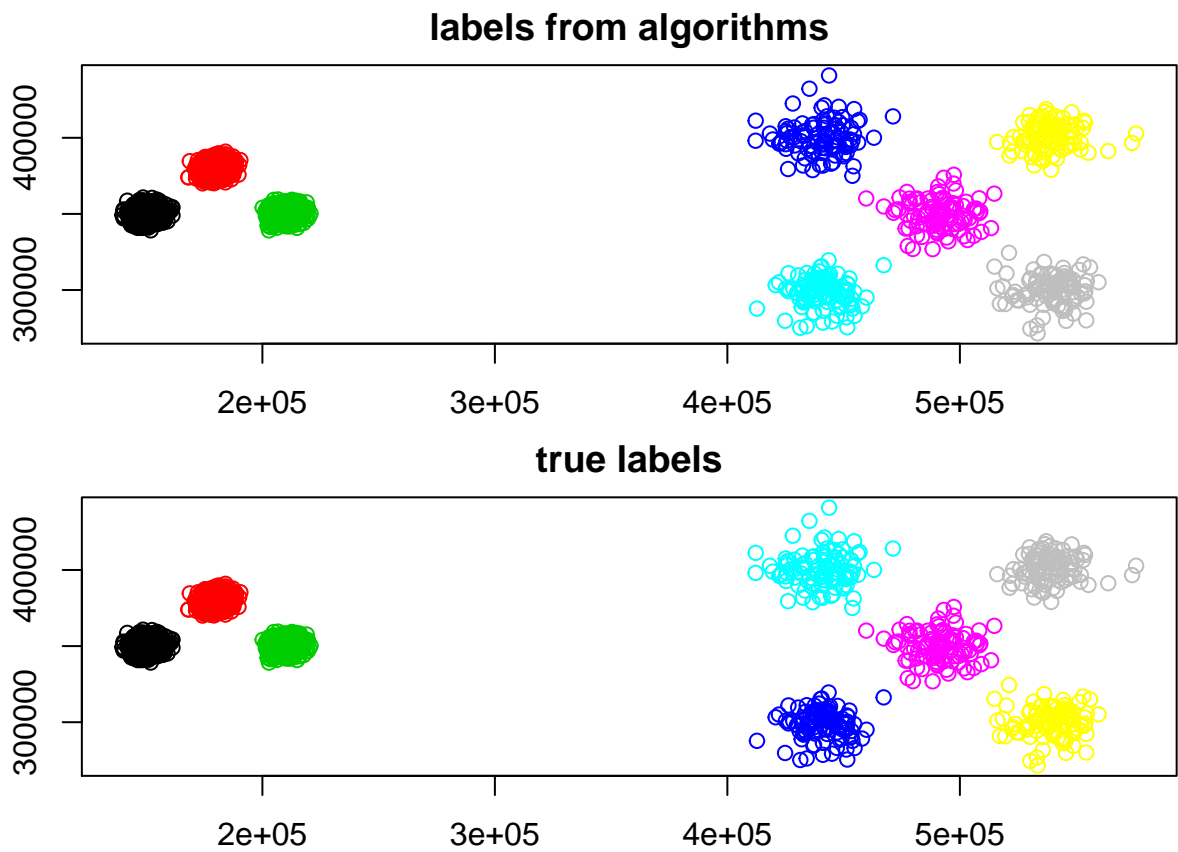
| set | FM | AM |
|--------------------------|--------|-------|
| fcps/atom.data.gz | 0.146 | 0.649 |
| fcps/chainlink.data.gz | 0.093 | 0.546 |
| fcps/engytime.data.gz | 0.815 | 0.908 |
| fcps/hepta.data.gz | 1.000 | 1.000 |
| fcps/lsun.data.gz | 0.429 | 0.638 |
| fcps/target.data.gz | 0.636 | 0.793 |
| fcps/tetra.data.gz | 1.000 | 1.000 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 |
| fcps/wingnut.data.gz | 0.859 | 0.930 |
| graves/dense.data.gz | 0.809 | 0.905 |
| graves/fuzzyx.data.gz | 0.607 | 0.690 |
| graves/line.data.gz | -0.009 | 0.598 |
| graves/parabolic.data.gz | 0.589 | 0.795 |
| graves/ring.data.gz | 0.166 | 0.652 |
| graves/zigzag.data.gz | 0.136 | 0.496 |
| other/iris.data.gz | 0.730 | 0.821 |
| other/iris5.data.gz | 0.505 | 0.735 |
| other/square.data.gz | 0.028 | 0.517 |
| sipu/a1.data.gz | 0.966 | 0.968 |
| sipu/a2.data.gz | 0.968 | 0.969 |
| sipu/a3.data.gz | 0.972 | 0.973 |
| sipu/aggregation.data.gz | 0.762 | 0.816 |
| sipu/compound.data.gz | 0.538 | 0.642 |
| sipu/d31.data.gz | 0.953 | 0.955 |
| sipu/flame.data.gz | 0.488 | 0.753 |
| sipu/jain.data.gz | 0.318 | 0.698 |
| sipu/pathbased.data.gz | 0.462 | 0.662 |
| sipu/r15.data.gz | 0.993 | 0.993 |
| sipu/s1.data.gz | 0.987 | 0.988 |
| sipu/s2.data.gz | 0.937 | 0.942 |
| sipu/s3.data.gz | 0.725 | 0.743 |
| sipu/s4.data.gz | 0.594 | 0.623 |
| sipu/spiral.data.gz | -0.006 | 0.328 |
| sipu/unbalance.data.gz | 1.000 | 1.000 |
| wut/cross.data.gz | 0.109 | 0.441 |
| wut/smile.data.gz | 0.441 | 0.585 |
| wut/twosplashes.data.gz | 0.279 | 0.639 |
| wut/x1.data.gz | 1.000 | 1.000 |
| wut/x2.data.gz | 0.209 | 0.551 |
| wut/x3.data.gz | 0.856 | 0.894 |
| wut/z1.data.gz | 0.203 | 0.466 |
| wut/z2.data.gz | 0.457 | 0.629 |
| wut/z3.data.gz | 1.000 | 1.000 |

6.1.1.2 Moje zbiory

Wyniki działania na moich zbiorach danych przedstawiają się następująco:

| set | FM | AM |
|-----------------|-------|-------|
| benchmark1.data | 0.493 | 0.746 |
| benchmark2.data | 0.998 | 0.998 |
| benchmark3.data | 0.941 | 0.953 |
| benchmark4.data | 0.373 | 0.502 |

6.1.1.3 Przykładowy zbiór



6.1.2 Ze standaryzacją zmiennych

6.1.2.1 Zbiory dostarczone

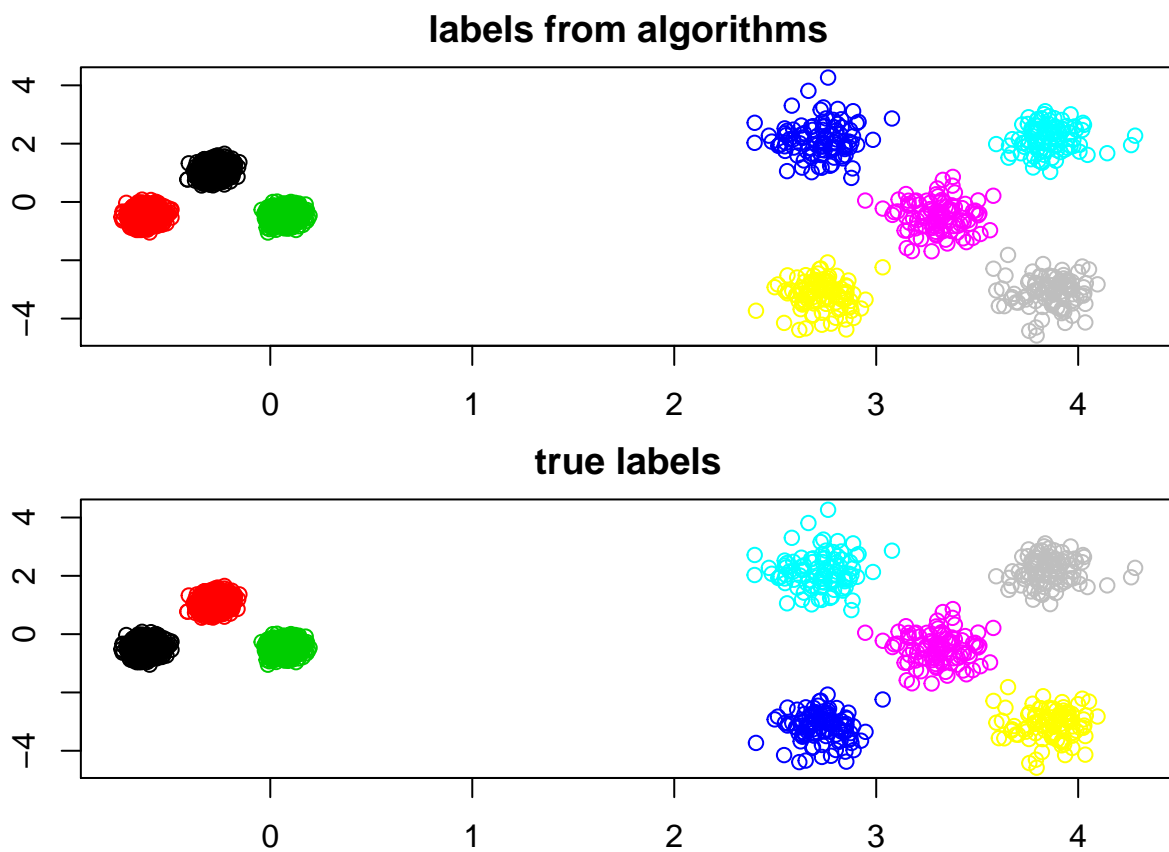
| set | FM | AM |
|--------------------------|--------|-------|
| fcps/atom.data.gz | 0.149 | 0.650 |
| fcps/chainlink.data.gz | -0.001 | 0.499 |
| fcps/engytime.data.gz | 0.812 | 0.907 |
| fcps/hepta.data.gz | 1.000 | 1.000 |
| fcps/lsun.data.gz | 0.405 | 0.624 |
| fcps/target.data.gz | 0.636 | 0.793 |
| fcps/tetra.data.gz | 1.000 | 1.000 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 |
| fcps/wingnut.data.gz | 0.670 | 0.835 |
| graves/dense.data.gz | 0.738 | 0.870 |
| graves/fuzzyx.data.gz | 0.598 | 0.683 |
| graves/line.data.gz | -0.009 | 0.598 |
| graves/parabolic.data.gz | 0.550 | 0.775 |
| graves/ring.data.gz | 0.166 | 0.652 |
| graves/zigzag.data.gz | 0.182 | 0.510 |
| other/iris.data.gz | 0.620 | 0.745 |
| other/iris5.data.gz | 0.408 | 0.675 |
| other/square.data.gz | -0.001 | 0.499 |
| sipu/a1.data.gz | 0.918 | 0.922 |
| sipu/a2.data.gz | 0.956 | 0.957 |
| sipu/a3.data.gz | 0.972 | 0.973 |
| sipu/aggregation.data.gz | 0.713 | 0.775 |
| sipu/compound.data.gz | 0.499 | 0.611 |
| sipu/d3l.data.gz | 0.956 | 0.957 |
| sipu/flame.data.gz | 0.431 | 0.725 |
| sipu/jain.data.gz | 0.553 | 0.809 |
| sipu/pathbased.data.gz | 0.480 | 0.670 |
| sipu/r15.data.gz | 0.993 | 0.993 |
| sipu/s1.data.gz | 0.987 | 0.988 |
| sipu/s2.data.gz | 0.938 | 0.942 |
| sipu/s3.data.gz | 0.725 | 0.743 |
| sipu/s4.data.gz | 0.633 | 0.657 |
| sipu/spiral.data.gz | -0.006 | 0.328 |
| sipu/unbalance.data.gz | 1.000 | 1.000 |
| wut/cross.data.gz | 0.109 | 0.441 |
| wut/smile.data.gz | 0.440 | 0.584 |
| wut/twosplashes.data.gz | 0.091 | 0.544 |
| wut/x1.data.gz | 1.000 | 1.000 |
| wut/x2.data.gz | 0.209 | 0.551 |
| wut/x3.data.gz | 0.856 | 0.894 |
| wut/z1.data.gz | 0.167 | 0.445 |
| wut/z2.data.gz | 0.428 | 0.607 |
| wut/z3.data.gz | 1.000 | 1.000 |

6.1.2.2 Moje zbiory

Wyniki działania na moich zbiorach danych przedstawiają się następująco:

| set | FM | AM |
|-----------------|-------|-------|
| benchmark1.data | 0.975 | 0.988 |
| benchmark2.data | 0.998 | 0.998 |
| benchmark3.data | 0.931 | 0.945 |
| benchmark4.data | 0.252 | 0.401 |

6.1.2.3 Przykładowy zbiór



7 Algorytm *cmeans* z pakietu e1071

7.1 Badanie algorytmu na wszystkich zbiorach benchmarkowych

7.1.1 Bez standaryzacji zmiennych

7.1.1.1 Zbiory dostarczone

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|--------------------------|--------|--------|--------|--------|--------|--------|--------|
| fcps/atom.data.gz | 0.223 | 0.014 | 0.009 | 0.010 | 0.010 | 0.010 | 0.010 |
| fcps/chainlink.data.gz | 0.087 | 0.089 | 0.089 | 0.088 | 0.088 | 0.089 | 0.090 |
| fcps/engytime.data.gz | 0.783 | 0.788 | 0.791 | 0.795 | 0.798 | 0.798 | 0.798 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 0.651 | 0.816 | 0.641 |
| fcps/lsun.data.gz | 0.505 | 0.545 | 0.561 | 0.572 | 0.583 | 0.589 | 0.613 |
| fcps/target.data.gz | 0.383 | 0.161 | 0.102 | 0.290 | 0.034 | 0.007 | 0.178 |
| fcps/tetra.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 0.706 | 0.791 | 0.467 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/wingnut.data.gz | 0.771 | 0.750 | 0.736 | 0.736 | 0.736 | 0.730 | 0.730 |
| graves/dense.data.gz | 0.791 | 0.791 | 0.791 | 0.791 | 0.791 | 0.791 | 0.791 |
| graves/fuzzxyx.data.gz | 0.569 | 0.577 | 0.625 | 0.715 | 0.732 | 0.744 | 0.753 |
| graves/line.data.gz | -0.009 | 0.015 | 0.015 | 0.015 | 0.015 | 0.015 | 0.015 |
| graves/parabolic.data.gz | 0.605 | 0.605 | 0.599 | 0.599 | 0.599 | 0.599 | 0.599 |
| graves/ring.data.gz | -0.001 | -0.001 | -0.001 | -0.001 | -0.001 | -0.001 | -0.001 |
| graves/zigzag.data.gz | 0.144 | 0.173 | 0.157 | 0.165 | 0.165 | 0.165 | 0.164 |
| other/iris.data.gz | 0.729 | 0.743 | 0.757 | 0.742 | 0.771 | 0.771 | 0.771 |
| other/iris5.data.gz | 0.505 | 0.363 | 0.367 | 0.371 | 0.371 | 0.375 | 0.375 |
| other/square.data.gz | 0.027 | 0.022 | 0.019 | 0.020 | 0.020 | 0.020 | 0.020 |
| sipu/a1.data.gz | 0.822 | 0.825 | 0.779 | 0.648 | 0.605 | 0.592 | 0.581 |
| sipu/a2.data.gz | 0.842 | 0.760 | 0.720 | 0.605 | 0.552 | 0.574 | 0.604 |
| sipu/a3.data.gz | 0.717 | 0.722 | 0.658 | 0.576 | 0.586 | 0.551 | 0.625 |
| sipu/aggregation.data.gz | 0.611 | 0.533 | 0.696 | 0.677 | 0.549 | 0.610 | 0.548 |
| sipu/compound.data.gz | 0.536 | 0.534 | 0.467 | 0.402 | 0.406 | 0.380 | 0.413 |
| sipu/d31.data.gz | 0.788 | 0.783 | 0.638 | 0.565 | 0.648 | 0.531 | 0.567 |
| sipu/flame.data.gz | 0.488 | 0.476 | 0.465 | 0.465 | 0.465 | 0.465 | 0.465 |
| sipu/jain.data.gz | 0.300 | 0.277 | 0.277 | 0.266 | 0.266 | 0.266 | 0.266 |
| sipu/pathbased.data.gz | 0.460 | 0.353 | 0.330 | 0.335 | 0.334 | 0.336 | 0.336 |
| sipu/r15.data.gz | 0.890 | 0.694 | 0.695 | 0.686 | 0.648 | 0.739 | 0.502 |
| sipu/s1.data.gz | 0.902 | 0.899 | 0.845 | 0.707 | 0.780 | 0.771 | 0.661 |
| sipu/s2.data.gz | 0.839 | 0.848 | 0.862 | 0.805 | 0.645 | 0.695 | 0.646 |
| sipu/s3.data.gz | 0.669 | 0.731 | 0.676 | 0.578 | 0.527 | 0.534 | 0.499 |
| sipu/s4.data.gz | 0.637 | 0.638 | 0.608 | 0.508 | 0.422 | 0.440 | 0.431 |
| sipu/spiral.data.gz | -0.006 | -0.006 | -0.005 | -0.005 | -0.005 | -0.005 | -0.005 |
| sipu/unbalance.data.gz | 0.738 | 0.690 | 0.691 | 0.696 | 0.552 | 0.587 | 0.592 |
| wut/cross.data.gz | 0.193 | 0.310 | 0.377 | 0.421 | 0.457 | 0.484 | 0.493 |
| wut/smile.data.gz | 0.577 | 0.444 | 0.444 | 0.443 | 0.572 | 0.572 | 0.558 |
| wut/twosplashes.data.gz | 0.279 | 0.279 | 0.279 | 0.279 | 0.279 | 0.279 | 0.279 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.209 | 0.648 | 0.629 | 0.610 | 0.592 | 0.592 | 0.575 |
| wut/x3.data.gz | 0.845 | 0.867 | 0.891 | 0.891 | 0.891 | 0.907 | 0.907 |
| wut/z1.data.gz | 0.202 | 0.207 | 0.228 | 0.229 | 0.239 | 0.239 | 0.250 |
| wut/z2.data.gz | 0.442 | 0.420 | 0.286 | 0.412 | 0.285 | 0.414 | 0.414 |
| wut/z3.data.gz | 0.995 | 0.775 | 0.764 | 0.761 | 0.756 | 0.756 | 0.754 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|--------------------------|-------|-------|-------|-------|-------|-------|-------|
| fcps/atom.data.gz | 0.661 | 0.507 | 0.504 | 0.505 | 0.505 | 0.505 | 0.505 |
| fcps/chainlink.data.gz | 0.543 | 0.544 | 0.544 | 0.544 | 0.544 | 0.544 | 0.545 |
| fcps/engytime.data.gz | 0.891 | 0.894 | 0.895 | 0.897 | 0.899 | 0.899 | 0.899 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 0.705 | 0.844 | 0.700 |
| fcps/lsun.data.gz | 0.687 | 0.713 | 0.723 | 0.729 | 0.736 | 0.740 | 0.755 |
| fcps/target.data.gz | 0.613 | 0.425 | 0.374 | 0.534 | 0.324 | 0.303 | 0.439 |
| fcps/tetra.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 0.783 | 0.844 | 0.617 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/wingnut.data.gz | 0.885 | 0.875 | 0.868 | 0.868 | 0.868 | 0.865 | 0.865 |
| graves/dense.data.gz | 0.896 | 0.896 | 0.896 | 0.896 | 0.896 | 0.896 | 0.896 |
| graves/fuzzyx.data.gz | 0.662 | 0.667 | 0.704 | 0.774 | 0.787 | 0.796 | 0.803 |
| graves/line.data.gz | 0.598 | 0.601 | 0.601 | 0.601 | 0.601 | 0.601 | 0.601 |
| graves/parabolic.data.gz | 0.802 | 0.802 | 0.799 | 0.799 | 0.799 | 0.799 | 0.799 |
| graves/ring.data.gz | 0.499 | 0.499 | 0.499 | 0.499 | 0.499 | 0.499 | 0.499 |
| graves/zigzag.data.gz | 0.497 | 0.507 | 0.495 | 0.497 | 0.497 | 0.497 | 0.497 |
| other/iris.data.gz | 0.820 | 0.828 | 0.838 | 0.828 | 0.847 | 0.847 | 0.847 |
| other/iris5.data.gz | 0.732 | 0.614 | 0.616 | 0.617 | 0.617 | 0.620 | 0.620 |
| other/square.data.gz | 0.516 | 0.512 | 0.510 | 0.510 | 0.511 | 0.510 | 0.510 |
| sipu/a1.data.gz | 0.832 | 0.836 | 0.792 | 0.674 | 0.634 | 0.625 | 0.615 |
| sipu/a2.data.gz | 0.848 | 0.773 | 0.735 | 0.623 | 0.575 | 0.602 | 0.625 |
| sipu/a3.data.gz | 0.729 | 0.732 | 0.673 | 0.601 | 0.605 | 0.572 | 0.639 |
| sipu/aggregation.data.gz | 0.692 | 0.626 | 0.762 | 0.744 | 0.639 | 0.690 | 0.638 |
| sipu/compound.data.gz | 0.640 | 0.639 | 0.586 | 0.532 | 0.535 | 0.513 | 0.542 |
| sipu/d31.data.gz | 0.798 | 0.794 | 0.655 | 0.591 | 0.666 | 0.555 | 0.597 |
| sipu/flame.data.gz | 0.753 | 0.747 | 0.742 | 0.742 | 0.742 | 0.742 | 0.742 |
| sipu/jain.data.gz | 0.689 | 0.679 | 0.679 | 0.674 | 0.674 | 0.674 | 0.674 |
| sipu/pathbased.data.gz | 0.660 | 0.582 | 0.567 | 0.574 | 0.574 | 0.577 | 0.577 |
| sipu/r15.data.gz | 0.898 | 0.723 | 0.721 | 0.712 | 0.685 | 0.762 | 0.567 |
| sipu/s1.data.gz | 0.910 | 0.907 | 0.856 | 0.733 | 0.798 | 0.790 | 0.699 |
| sipu/s2.data.gz | 0.851 | 0.859 | 0.873 | 0.820 | 0.675 | 0.719 | 0.673 |
| sipu/s3.data.gz | 0.692 | 0.749 | 0.699 | 0.608 | 0.563 | 0.572 | 0.546 |
| sipu/s4.data.gz | 0.661 | 0.662 | 0.634 | 0.544 | 0.470 | 0.486 | 0.477 |
| sipu/spiral.data.gz | 0.327 | 0.327 | 0.328 | 0.328 | 0.328 | 0.328 | 0.328 |
| sipu/unbalance.data.gz | 0.814 | 0.779 | 0.780 | 0.783 | 0.677 | 0.705 | 0.707 |
| wut/cross.data.gz | 0.472 | 0.531 | 0.569 | 0.594 | 0.616 | 0.633 | 0.639 |
| wut/smile.data.gz | 0.692 | 0.587 | 0.587 | 0.587 | 0.688 | 0.688 | 0.677 |
| wut/twosplashes.data.gz | 0.639 | 0.639 | 0.639 | 0.639 | 0.639 | 0.639 | 0.639 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.551 | 0.769 | 0.755 | 0.743 | 0.730 | 0.730 | 0.719 |
| wut/x3.data.gz | 0.886 | 0.902 | 0.920 | 0.920 | 0.920 | 0.932 | 0.932 |
| wut/z1.data.gz | 0.465 | 0.469 | 0.483 | 0.484 | 0.491 | 0.491 | 0.499 |
| wut/z2.data.gz | 0.618 | 0.601 | 0.498 | 0.595 | 0.499 | 0.597 | 0.597 |
| wut/z3.data.gz | 0.997 | 0.840 | 0.832 | 0.830 | 0.826 | 0.826 | 0.825 |

7.1.1.2 Moje zbiory

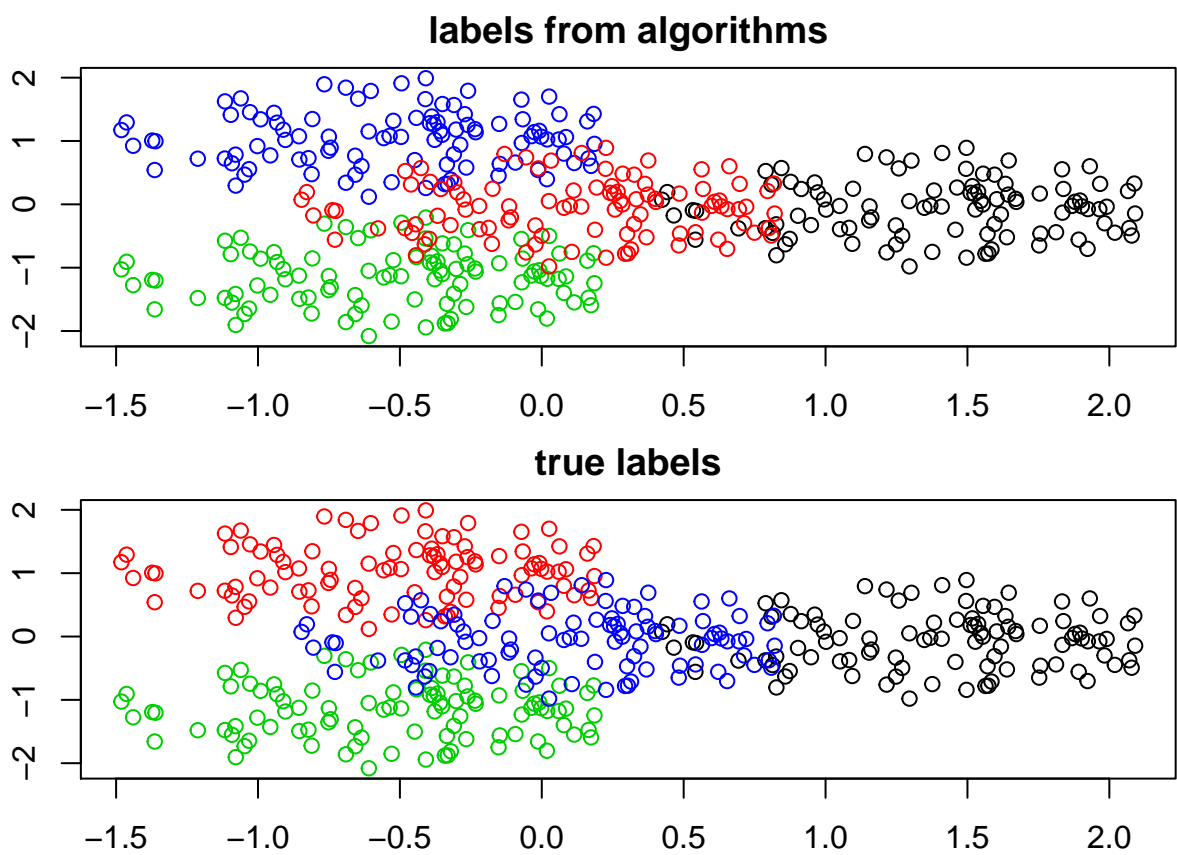
Wyniki działania na moich zbiorach danych przedstawiają się następująco:

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.507 | 0.518 | 0.518 | 0.518 | 0.518 | 0.518 | 0.521 |
| benchmark2.data | 0.998 | 0.765 | 0.838 | 0.769 | 0.767 | 0.710 | 0.684 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|
| benchmark3.data | 0.941 | 0.941 | 0.941 | 0.941 | 0.941 | 0.941 | 0.941 |
| benchmark4.data | 0.405 | 0.369 | 0.370 | 0.366 | 0.370 | 0.378 | 0.387 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.753 | 0.759 | 0.759 | 0.759 | 0.759 | 0.759 | 0.760 |
| benchmark2.data | 0.998 | 0.786 | 0.852 | 0.790 | 0.788 | 0.738 | 0.719 |
| benchmark3.data | 0.953 | 0.953 | 0.953 | 0.953 | 0.953 | 0.953 | 0.953 |
| benchmark4.data | 0.524 | 0.498 | 0.498 | 0.495 | 0.498 | 0.504 | 0.510 |

7.1.1.3 Przykładowy zbiór



7.1.2 Ze standaryzacją zmiennych

7.1.2.1 Zbiory destarczone

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|--------------------------|--------|--------|--------|--------|--------|--------|--------|
| fcps/atom.data.gz | 0.021 | 0.033 | 0.023 | 0.028 | 0.029 | 0.041 | 0.004 |
| fcps/chainlink.data.gz | 0.081 | 0.083 | 0.087 | 0.084 | 0.086 | 0.094 | 0.100 |
| fcps/engytime.data.gz | 0.821 | 0.821 | 0.819 | 0.819 | 0.820 | 0.822 | 0.822 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.361 |
| fcps/lsun.data.gz | 0.408 | 0.429 | 0.452 | 0.453 | 0.476 | 0.489 | 0.500 |
| fcps/target.data.gz | 0.383 | 0.098 | 0.385 | 0.167 | 0.001 | 0.117 | 0.008 |
| fcps/tetra.data.gz | 1.000 | 1.000 | 1.000 | 0.617 | 0.694 | 0.686 | 0.671 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/wingnut.data.gz | 0.547 | 0.524 | 0.519 | 0.519 | 0.519 | 0.519 | 0.519 |
| graves/dense.data.gz | 0.738 | 0.738 | 0.738 | 0.738 | 0.756 | 0.756 | 0.773 |
| graves/fuzzxyx.data.gz | 0.569 | 0.577 | 0.623 | 0.715 | 0.732 | 0.744 | 0.770 |
| graves/line.data.gz | -0.009 | 0.015 | 0.015 | 0.015 | 0.015 | 0.015 | 0.015 |
| graves/parabolic.data.gz | 0.556 | 0.565 | 0.565 | 0.571 | 0.571 | 0.571 | 0.571 |
| graves/ring.data.gz | -0.001 | -0.001 | -0.001 | -0.001 | -0.001 | -0.001 | -0.001 |
| graves/zigzag.data.gz | 0.182 | 0.193 | 0.195 | 0.187 | 0.187 | 0.184 | 0.180 |
| other/iris.data.gz | 0.630 | 0.630 | 0.652 | 0.652 | 0.664 | 0.664 | 0.664 |
| other/iris5.data.gz | 0.387 | 0.363 | 0.346 | 0.334 | 0.319 | 0.339 | 0.345 |
| other/square.data.gz | 0.022 | 0.022 | 0.020 | 0.020 | 0.019 | 0.018 | 0.019 |
| sipu/a1.data.gz | 0.791 | 0.687 | 0.723 | 0.701 | 0.560 | 0.568 | 0.508 |
| sipu/a2.data.gz | 0.769 | 0.724 | 0.655 | 0.525 | 0.622 | 0.635 | 0.533 |
| sipu/a3.data.gz | 0.802 | 0.653 | 0.616 | 0.568 | 0.562 | 0.551 | 0.543 |
| sipu/aggregation.data.gz | 0.731 | 0.691 | 0.575 | 0.585 | 0.423 | 0.553 | 0.533 |
| sipu/compound.data.gz | 0.510 | 0.396 | 0.467 | 0.403 | 0.423 | 0.421 | 0.437 |
| sipu/d31.data.gz | 0.805 | 0.677 | 0.648 | 0.659 | 0.594 | 0.542 | 0.570 |
| sipu/flame.data.gz | 0.500 | 0.476 | 0.476 | 0.476 | 0.465 | 0.465 | 0.465 |
| sipu/jain.data.gz | 0.537 | 0.522 | 0.515 | 0.515 | 0.515 | 0.515 | 0.515 |
| sipu/pathbased.data.gz | 0.484 | 0.481 | 0.369 | 0.361 | 0.363 | 0.361 | 0.366 |
| sipu/r15.data.gz | 0.881 | 0.828 | 0.559 | 0.649 | 0.743 | 0.687 | 0.486 |
| sipu/s1.data.gz | 0.895 | 0.912 | 0.837 | 0.690 | 0.688 | 0.818 | 0.836 |
| sipu/s2.data.gz | 0.937 | 0.860 | 0.851 | 0.785 | 0.737 | 0.820 | 0.765 |
| sipu/s3.data.gz | 0.682 | 0.667 | 0.578 | 0.661 | 0.581 | 0.586 | 0.632 |
| sipu/s4.data.gz | 0.604 | 0.637 | 0.527 | 0.481 | 0.522 | 0.433 | 0.387 |
| sipu/spiral.data.gz | -0.006 | -0.006 | -0.006 | -0.006 | -0.005 | -0.005 | -0.005 |
| sipu/unbalance.data.gz | 0.586 | 0.740 | 0.523 | 0.690 | 0.443 | 0.527 | 0.445 |
| wut/cross.data.gz | 0.191 | 0.305 | 0.373 | 0.420 | 0.457 | 0.471 | 0.491 |
| wut/smile.data.gz | 0.441 | 0.367 | 0.441 | 0.441 | 0.441 | 0.439 | 0.439 |
| wut/twosplashes.data.gz | 0.094 | 0.107 | 0.120 | 0.127 | 0.127 | 0.127 | 0.127 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.668 | 0.668 | 0.649 | 0.631 | 0.613 | 0.596 | 0.579 |
| wut/x3.data.gz | 0.845 | 0.867 | 0.891 | 0.891 | 0.891 | 0.891 | 0.891 |
| wut/z1.data.gz | 0.157 | 0.097 | 0.091 | 0.080 | 0.068 | 0.040 | -0.009 |
| wut/z2.data.gz | 0.430 | 0.413 | 0.295 | 0.398 | 0.287 | 0.405 | 0.285 |
| wut/z3.data.gz | 0.995 | 0.769 | 0.754 | 0.752 | 0.676 | 0.736 | 0.732 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|------------------------|-------|-------|-------|-------|-------|-------|-------|
| fcps/atom.data.gz | 0.512 | 0.520 | 0.513 | 0.518 | 0.516 | 0.525 | 0.502 |
| fcps/chainlink.data.gz | 0.540 | 0.541 | 0.543 | 0.542 | 0.542 | 0.547 | 0.550 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|--------------------------|-------|-------|-------|-------|-------|-------|-------|
| fcps/engytime.data.gz | 0.911 | 0.911 | 0.910 | 0.910 | 0.910 | 0.911 | 0.911 |
| fcps/hepta.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 0.541 |
| fcps/lsun.data.gz | 0.625 | 0.638 | 0.653 | 0.653 | 0.667 | 0.675 | 0.681 |
| fcps/target.data.gz | 0.613 | 0.371 | 0.615 | 0.432 | 0.286 | 0.396 | 0.310 |
| fcps/tetra.data.gz | 1.000 | 1.000 | 1.000 | 0.719 | 0.773 | 0.768 | 0.759 |
| fcps/twodiamonds.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| fcps/wingnut.data.gz | 0.773 | 0.762 | 0.759 | 0.759 | 0.759 | 0.759 | 0.759 |
| graves/dense.data.gz | 0.870 | 0.870 | 0.870 | 0.870 | 0.878 | 0.878 | 0.887 |
| graves/fuzzysx.data.gz | 0.662 | 0.667 | 0.702 | 0.774 | 0.787 | 0.796 | 0.817 |
| graves/line.data.gz | 0.598 | 0.601 | 0.601 | 0.601 | 0.601 | 0.601 | 0.601 |
| graves/parabolic.data.gz | 0.778 | 0.782 | 0.782 | 0.785 | 0.785 | 0.785 | 0.785 |
| graves/ring.data.gz | 0.499 | 0.499 | 0.499 | 0.499 | 0.499 | 0.499 | 0.499 |
| graves/zigzag.data.gz | 0.508 | 0.512 | 0.510 | 0.506 | 0.506 | 0.504 | 0.503 |
| other/iris.data.gz | 0.752 | 0.752 | 0.767 | 0.767 | 0.775 | 0.775 | 0.775 |
| other/iris5.data.gz | 0.665 | 0.613 | 0.602 | 0.594 | 0.585 | 0.598 | 0.602 |
| other/square.data.gz | 0.512 | 0.511 | 0.511 | 0.510 | 0.510 | 0.509 | 0.510 |
| sipu/a1.data.gz | 0.803 | 0.709 | 0.739 | 0.720 | 0.586 | 0.595 | 0.544 |
| sipu/a2.data.gz | 0.782 | 0.735 | 0.671 | 0.551 | 0.639 | 0.655 | 0.562 |
| sipu/a3.data.gz | 0.807 | 0.665 | 0.631 | 0.585 | 0.582 | 0.577 | 0.567 |
| sipu/aggregation.data.gz | 0.787 | 0.757 | 0.661 | 0.669 | 0.534 | 0.643 | 0.626 |
| sipu/compound.data.gz | 0.621 | 0.527 | 0.586 | 0.533 | 0.549 | 0.548 | 0.561 |
| sipu/d31.data.gz | 0.813 | 0.691 | 0.662 | 0.676 | 0.617 | 0.576 | 0.593 |
| sipu/flame.data.gz | 0.759 | 0.747 | 0.747 | 0.747 | 0.742 | 0.742 | 0.742 |
| sipu/jain.data.gz | 0.802 | 0.794 | 0.790 | 0.790 | 0.790 | 0.790 | 0.790 |
| sipu/pathbased.data.gz | 0.672 | 0.668 | 0.586 | 0.582 | 0.587 | 0.587 | 0.593 |
| sipu/r15.data.gz | 0.890 | 0.844 | 0.608 | 0.680 | 0.765 | 0.714 | 0.552 |
| sipu/s1.data.gz | 0.904 | 0.919 | 0.853 | 0.718 | 0.716 | 0.832 | 0.849 |
| sipu/s2.data.gz | 0.941 | 0.870 | 0.862 | 0.800 | 0.757 | 0.833 | 0.783 |
| sipu/s3.data.gz | 0.704 | 0.690 | 0.610 | 0.685 | 0.612 | 0.615 | 0.660 |
| sipu/s4.data.gz | 0.632 | 0.661 | 0.559 | 0.520 | 0.559 | 0.480 | 0.435 |
| sipu/spiral.data.gz | 0.327 | 0.327 | 0.327 | 0.328 | 0.328 | 0.328 | 0.328 |
| sipu/unbalance.data.gz | 0.704 | 0.816 | 0.655 | 0.779 | 0.584 | 0.658 | 0.585 |
| wut/cross.data.gz | 0.471 | 0.528 | 0.566 | 0.593 | 0.616 | 0.625 | 0.638 |
| wut/smile.data.gz | 0.585 | 0.528 | 0.585 | 0.585 | 0.585 | 0.583 | 0.583 |
| wut/twosplashes.data.gz | 0.546 | 0.552 | 0.559 | 0.563 | 0.563 | 0.563 | 0.563 |
| wut/x1.data.gz | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| wut/x2.data.gz | 0.782 | 0.782 | 0.769 | 0.756 | 0.744 | 0.732 | 0.721 |
| wut/x3.data.gz | 0.886 | 0.902 | 0.920 | 0.920 | 0.920 | 0.920 | 0.920 |
| wut/z1.data.gz | 0.440 | 0.397 | 0.394 | 0.386 | 0.379 | 0.363 | 0.346 |
| wut/z2.data.gz | 0.609 | 0.595 | 0.504 | 0.584 | 0.500 | 0.590 | 0.501 |
| wut/z3.data.gz | 0.997 | 0.835 | 0.825 | 0.823 | 0.768 | 0.811 | 0.808 |

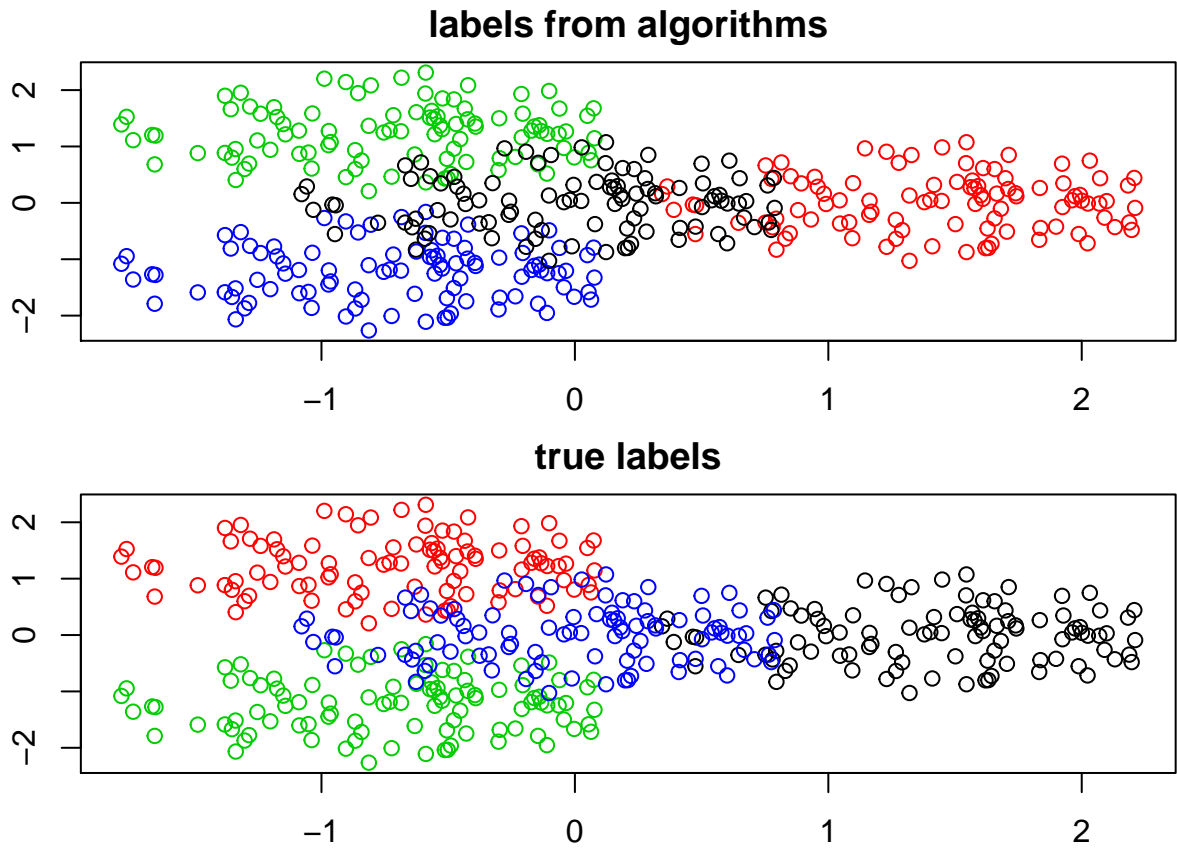
7.1.2.2 Moje zbiory

Wyniki działania na moich zbiorach danych przedstawiają się następująco:

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.965 | 0.965 | 0.965 | 0.965 | 0.965 | 0.965 | 0.965 |
| benchmark2.data | 0.866 | 0.866 | 0.998 | 0.793 | 0.658 | 0.709 | 0.804 |
| benchmark3.data | 0.922 | 0.927 | 0.927 | 0.927 | 0.927 | 0.927 | 0.927 |
| benchmark4.data | 0.226 | 0.222 | 0.227 | 0.232 | 0.010 | 0.091 | 0.005 |

| set | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------------|-------|-------|-------|-------|-------|-------|-------|
| benchmark1.data | 0.983 | 0.983 | 0.983 | 0.983 | 0.983 | 0.983 | 0.983 |
| benchmark2.data | 0.878 | 0.878 | 0.998 | 0.813 | 0.698 | 0.740 | 0.823 |
| benchmark3.data | 0.937 | 0.941 | 0.941 | 0.941 | 0.941 | 0.941 | 0.941 |
| benchmark4.data | 0.381 | 0.379 | 0.383 | 0.386 | 0.223 | 0.281 | 0.225 |

7.1.2.3 Przykładowy zbiór



8 Podsumowanie

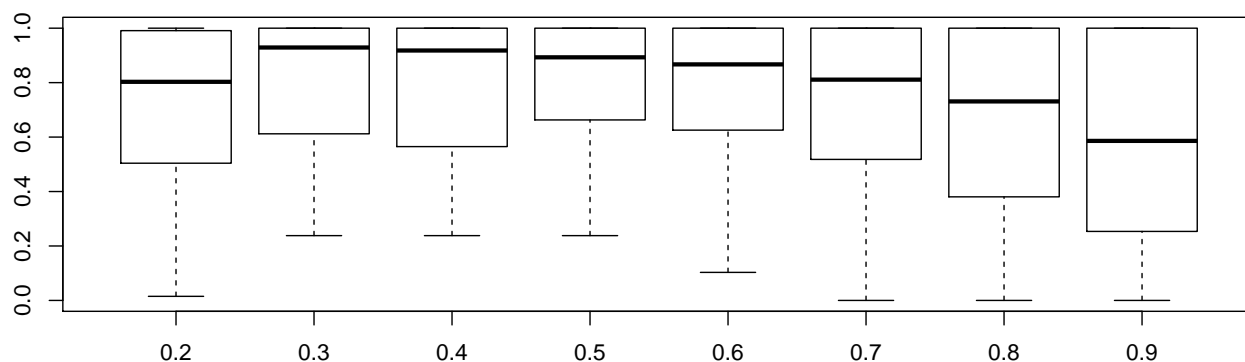
8.1 *genie*

Na podstawie tabel możemy zauważyć, iż najlepszym algorytmem jest algorytm *genie*. Właściwie dla wszystkich zbiorów algorytm znalazł prawie, że idealne rozwiązanie. Jednak nie widzimy znaczącej różnicy pomiędzy danymi ustandaryzowanymi a danymi bez standaryzacji. Wg mnie wynika to głównie z tego, iż dane tu są już po części standaryzowane. Pewne różnice mogą być widoczne na etapie czasu wykonania danych funkcji.

Na podstawie wykresów widzimy, iż najlepszym parametrem dla tego algorytmu jest wartość 0.3. Dla parametrów *thresholdGini* z zakresu od 0.7 do 0.8 wartość błędu jest większa niż dla innych wartości.

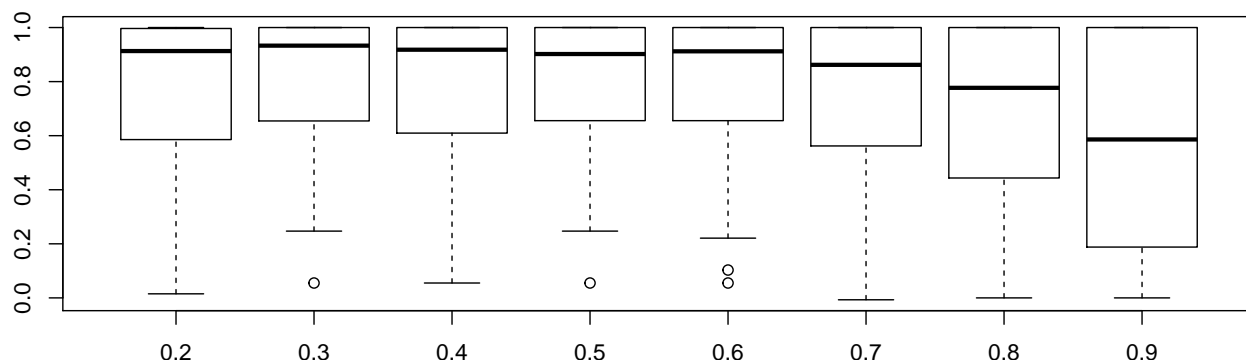
8.1.1 Bez standaryzacji zmiennych

Poniżej jest przedstawiony *barplot* dla algorytmu *genie* (indeks *AM*) w zależności od parametru *thresholdGini*:



Najlepszym parametrem *thresholdGini* jest wartość 0.3 (swoją drogą jest to wartość defaultowa).

8.1.2 Ze standaryzacją zmiennych



Najlepszym parametrem *thresholdGieni* są wartości od 0.3 do 0.8. Wartości te dają podobne wyniki.

8.2 *hclust*

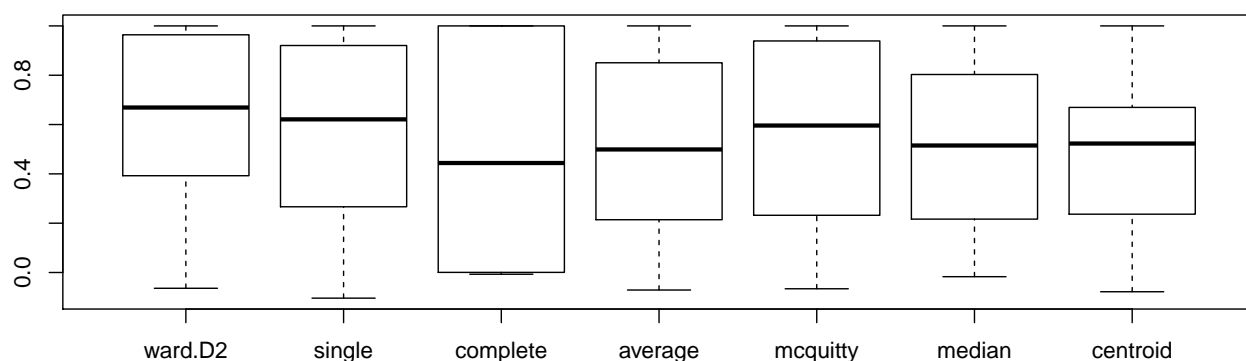
Na podstawie tabel możemy zauważyć, iż najlepszą metodą dla algorytmów hierarchicznych z pakietu *hclust* jest metoda *ward.D2*, gdyż średni błąd wg wykresu jest najmniejszy oraz odchylenie standardowe nie jest zbyt duże.

Jednak nie widzimy znaczącej różnicy pomiędzy danymi ustandaryzowanymi a danymi bez standaryzacji. Wg mnie wynika to głównie z tego, iż dane tu są już po części standaryzowane. Pewne różnice mogą być widoczne na etapie czasu wykonania danych funkcji.

Ponadto na podstawie wykresu możemy zauważyć, iż metoda *complete* radzi sobie najgorzej.

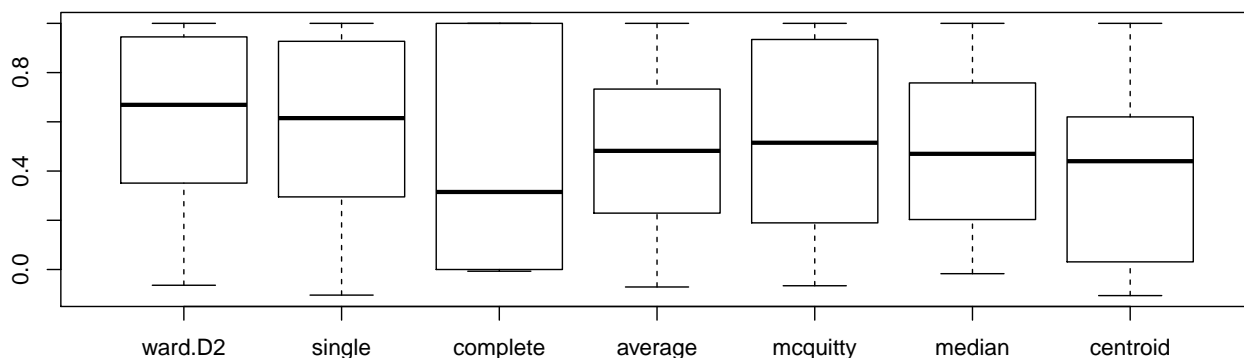
8.2.1 Bez standaryzacji zmiennych

Poniżej jest przedstawiony *barplot* dla algorytmu *hclust* (indeks *AM*) w zależności od metody clusteringu:



Najlepszą metodą jest metoda *ward.D2*.

8.2.2 Ze standaryzacją zmiennych



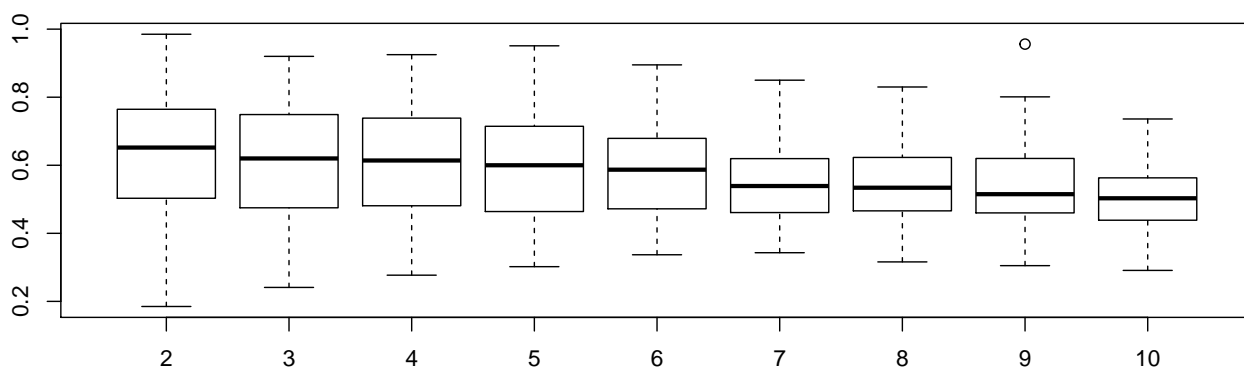
Najlepszą metodą jest metoda *ward.D2*.

8.3 *spectral_clustering*

Algorytm zaimplementowany przeze mnie daje satysfakcjonujące wyniki. Widać drobną poprawę, gdy wykonaliśmy standaryzację zmiennych.

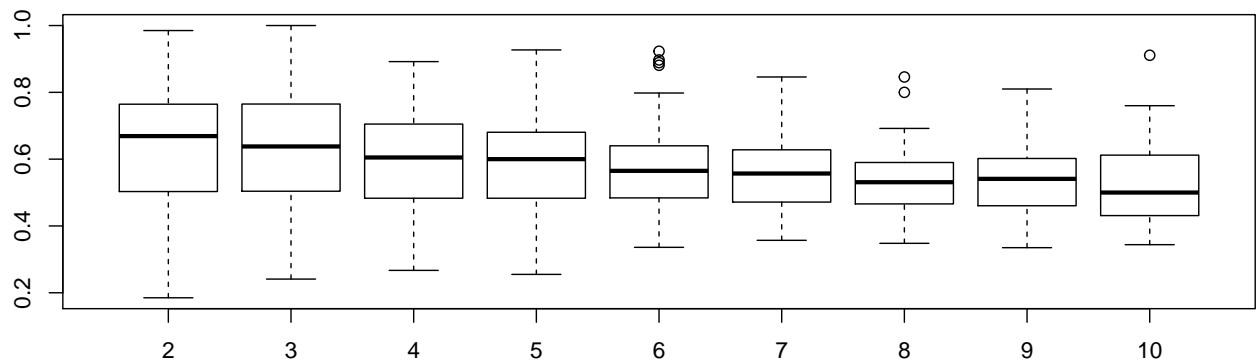
8.3.1 Bez standaryzacją zmiennych

Poniżej jest przedstawiony *barplot* dla algorytmu *spectral_clustering* (indeks *FM*) w zależności od parametru *M*:



Najlepszym parametrem jest $M = 3$.

8.3.2 Ze standaryzacją zmiennych



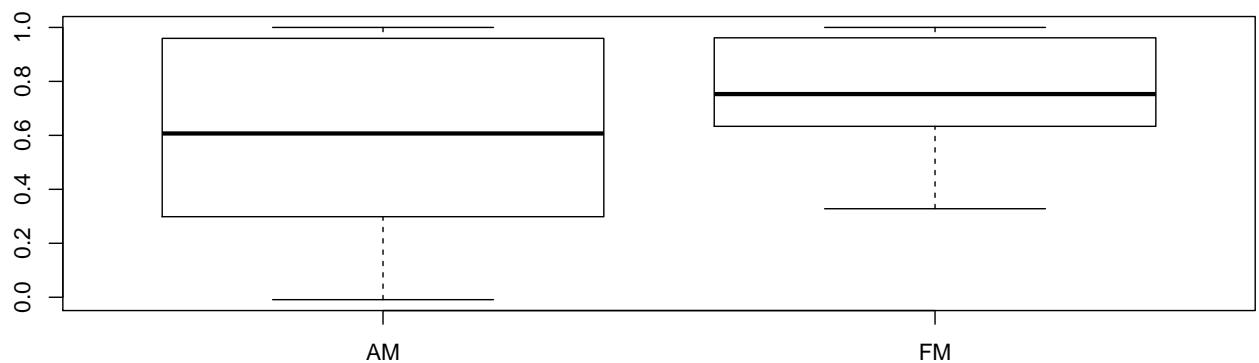
Najlepszym parametrem jest $M = 2$.

8.4 HCPC

Przy algorytmie *HCPC* nie badałem żadnego współczynnika. Algorytm przy ustawieniach defaultowych wypadł gorzej niż *genie*. Algorytm daje podobne rezultaty co algorytm zaimplementowany przeze mnie oraz *ward.D2*.

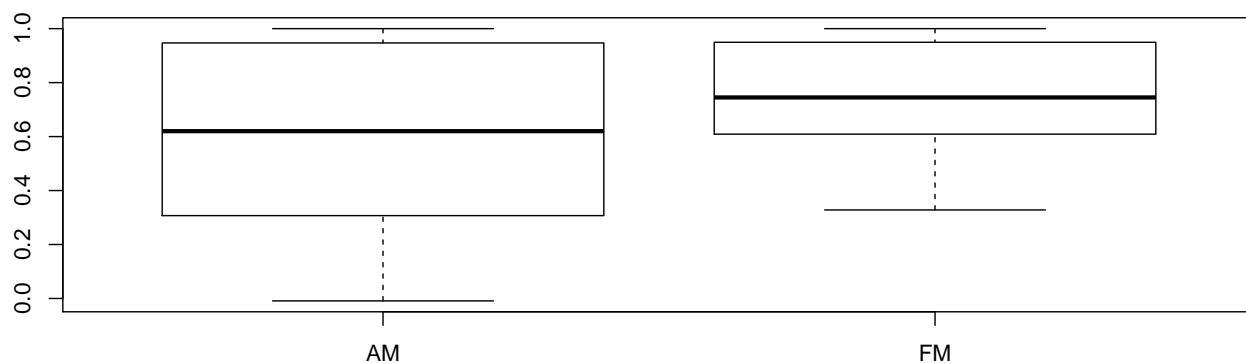
8.4.1 Bez standaryzacji zmiennych

Poniżej jest przedstawiony *barplot* dla algorytmu *HCPC* bez standaryzacji zmiennych:



8.4.2 Ze standaryzacją zmiennych

Poniżej jest przedstawiony *barplot* dla algorytmu *HCPC* ze standaryzacją zmiennych:

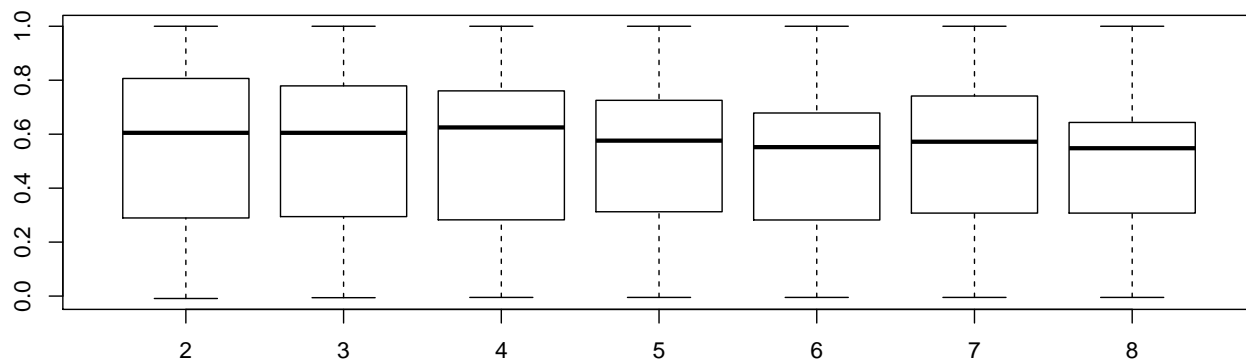


8.5 *cmeans*

Algorytm *cmeans* nie wypada najlepiej na tle innych.

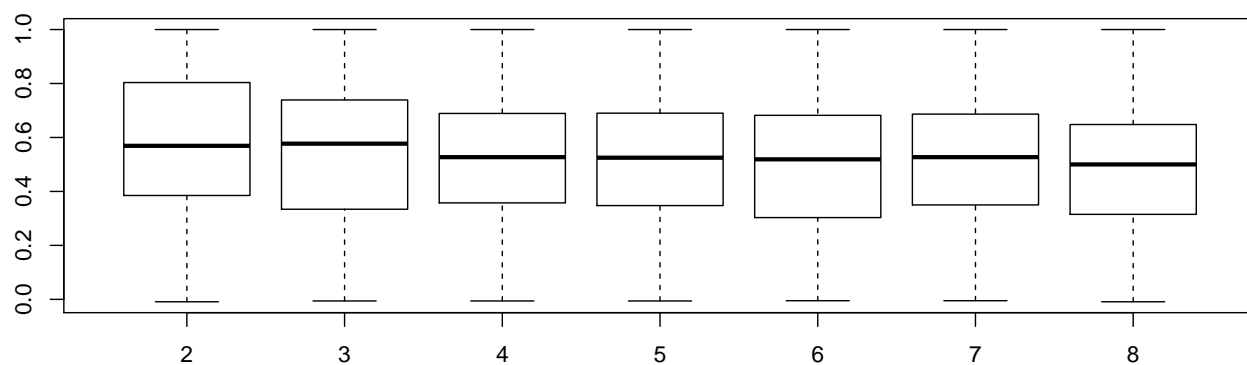
8.5.1 Bez standaryzacji zmiennych

Poniżej jest przedstawiony *barplot* dla algorytmu *cmeans*:



Najlepszym parametrem dla tego algorytmu bez standaryzacji jest wartość *rate.par* = 4, gdyż dla tego parametru średnia wartość jest największa.

8.5.2 Ze standaryzacją zmiennych



Najlepszym parametrem dla tego algorytmu ze standaryzacją jest wartość $rate.par = 3$.