z5_gis

Martyna Stajniak

2024-11-13

```
data<-read.csv("C:/Users/Martynaa/Desktop/5_semestr/GIS/z3/stajniak_output_0.txt",
               header=TRUE, sep=";", skip=1)

data$Dip_dir <- as.numeric(as.character(data$Dip_dir))
```

```
## Warning: pojawiły się wartości NA na skutek przekształcenia
```

```
data <- data[!is.na(data$Dip_dir), ]

write.table(data, "st_filtered_output2.txt", sep = ";", row.names = FALSE, quote = FALSE)
```

```
library(dplyr)
```

```
## Warning: pakiet 'dplyr' został zbudowany w wersji R 4.3.2
```

```
##
## Dołączanie pakietu: 'dplyr'
```

```
## Następujące obiekty zostały zakryte z 'package:stats':
##
##     filter, lag
```

```
## Następujące obiekty zostały zakryte z 'package:base':
##
##     intersect, setdiff, setequal, union
```

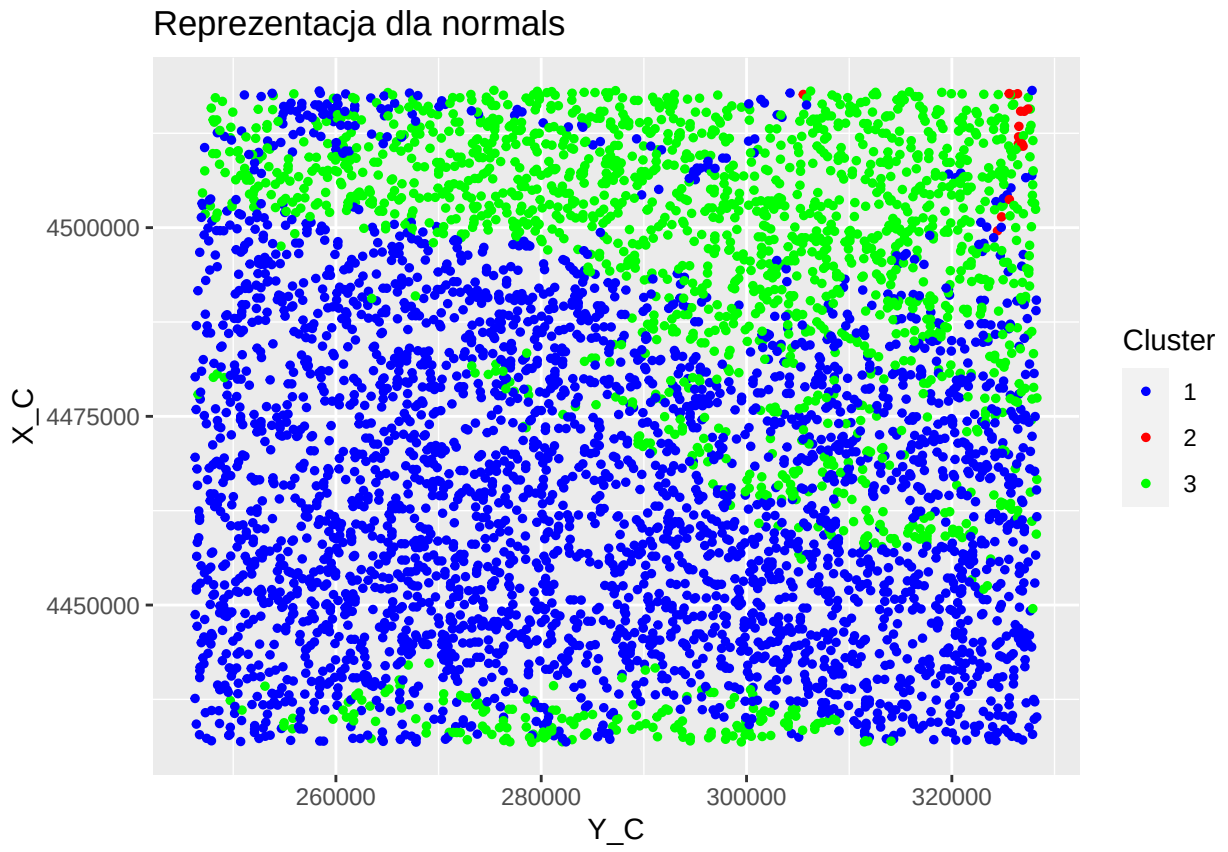
```
file = read.csv("C:/Users/Martynaa/Desktop/5_semestr/GIS/z3/st_filtered_output2.txt", sep=";")

df = filter( file, X_C >= min( file$X_C ) + 0.02*(max( file$X_C ) - min( file$X_C ) ))
df = filter( df, X_C <= max( file$X_C ) - 0.02*(max( file$X_C ) - min( file$X_C ) ))
df = filter( df, Y_C >= min( file$Y_C ) + 0.02*(max( file$Y_C ) - min( file$Y_C ) ))
df = filter( df, Y_C <= max( file$Y_C ) - 0.02*(max( file$Y_C ) - min( file$Y_C ) ))
write.table( x=df, file = "stajniak_output_0_filtred_cutted.txt", sep = ";", row.names = F )
```

```
library(ggplot2)
```

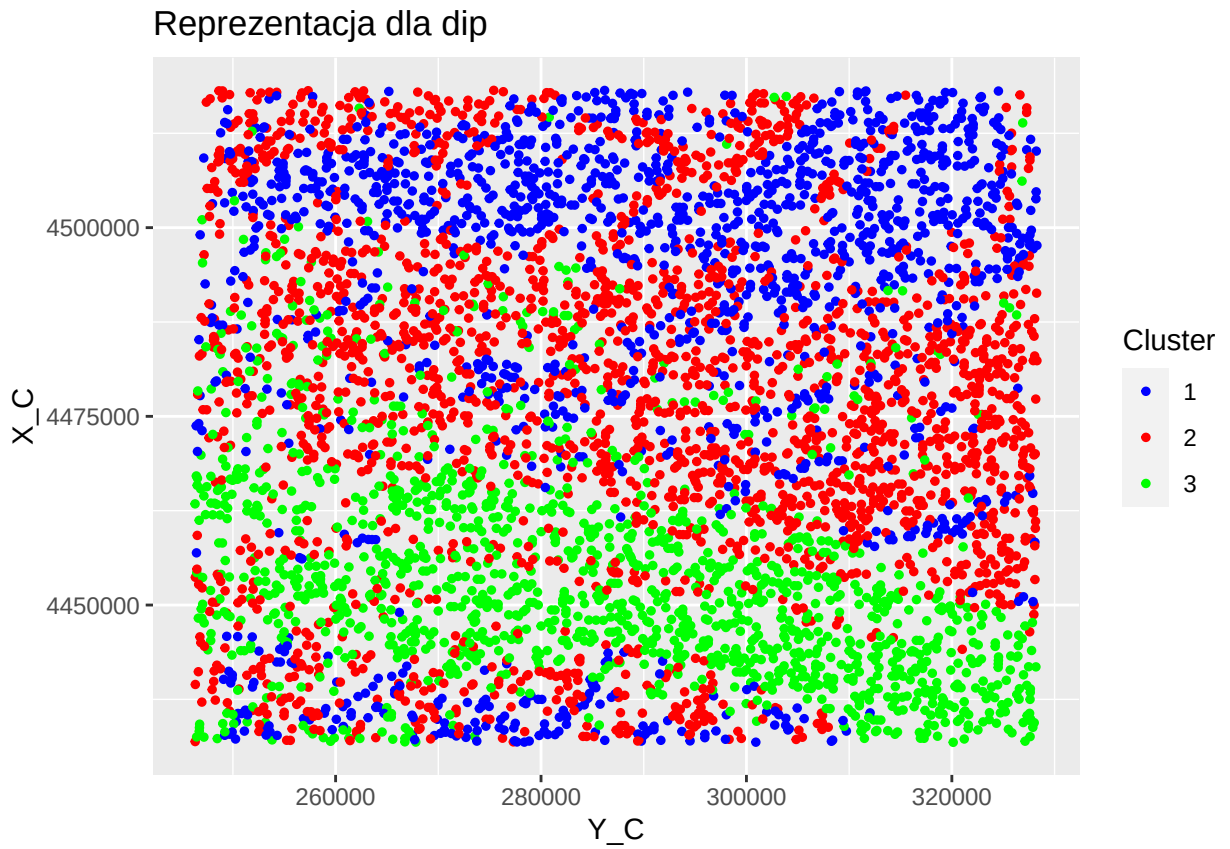
```
## Warning: pakiet 'ggplot2' został zbudowany w wersji R 4.3.2
```

```
library(dplyr)

file = read.csv("C:/Users/Martynaa/Desktop/5_semestr/GIS/z3/stajniak_output_0_filtred_cutted.txt",sep="


# ======== KMeans dla normals ============
grupowanie_normals <- kmeans(as.matrix(dplyr::select(file, c("X_N", "Y_N", "Z_N"))),
              centers = 3, nstart = 40, iter.max = 100000, algorithm = "Lloyd")
```

```r
# ======== KMeans dla dip ============
grupowanie_dip <- kmeans(as.matrix(dplyr::select(file, c("X_D", "Y_D", "Z_D"))),
              centers = 3, nstart = 40, iter.max = 100000, algorithm = "Lloyd")

# ======== Przypisanie klastrów ============
file1 <- file
file2 <- file

file1$clustering <- grupowanie_normals$cluster
file2$clustering <- grupowanie_dip$cluster

# ======== Próbkowanie dla wizualizacji ============
set.seed(36)
file1 <- sample_n(file1, 5000)
file2 <- sample_n(file2, 5000)

# ======== Wyciąganie danych dla normals ============
df1 <- dplyr::filter(file1, clustering == 1)[c("Dip_dir", "Dip_ang")]
df2 <- dplyr::filter(file1, clustering == 2)[c("Dip_dir", "Dip_ang")]
df3 <- dplyr::filter(file1, clustering == 3)[c("Dip_dir", "Dip_ang")]

# ======== Wyciąganie danych dla dip ============
df4 <- dplyr::filter(file2, clustering == 1)[c("Dip_dir", "Dip_ang")]
df5 <- dplyr::filter(file2, clustering == 2)[c("Dip_dir", "Dip_ang")]
df6 <- dplyr::filter(file2, clustering == 3)[c("Dip_dir", "Dip_ang")]

# ======== Eksport wyników dla normals ============
write.table(x = df1, file = "C:/Users/Martynaa/Desktop/5_semestr/GIS/projekt/finalny/norm_1.txt",
            sep = ",", row.names = F, col.names = F)
write.table(x = df2, file = "C:/Users/Martynaa/Desktop/5_semestr/GIS/projekt/finalny/norm_2.txt",
            sep = ",", row.names = F, col.names = F)
write.table(x = df3, file = "C:/Users/Martynaa/Desktop/5_semestr/GIS/projekt/finalny/norm_3.txt",
            sep = ",", row.names = F, col.names = F)

# ======== Eksport wyników dla dip ============
write.table(x = df4, file = "C:/Users/Martynaa/Desktop/5_semestr/GIS/projekt/finalny/dip_1.txt",
            sep = ",", row.names = F, col.names = F)
write.table(x = df5, file = "C:/Users/Martynaa/Desktop/5_semestr/GIS/projekt/finalny/dip_2.txt",
            sep = ",", row.names = F, col.names = F)
write.table(x = df6, file = "C:/Users/Martynaa/Desktop/5_semestr/GIS/projekt/finalny/dip_3.txt",
            sep = ",", row.names = F, col.names = F)

# ======== Wykres dla normals ============
palkmeans3 <- c( "blue", "red","green")
ggplot(file1, aes(Y_C, X_C, colour = factor(clustering))) +
  geom_point(size = 1) +
  scale_color_manual("Cluster", values = palkmeans3) +
  labs(title = "Reprezentacja dla normals")
```




```
# ======== Wykres dla dip ============
ggplot(file2, aes(Y_C, X_C, colour = factor(clustering))) +
  geom_point(size = 1) +
  scale_color_manual("Cluster", values = palkmeans3) +
  labs(title = "Reprezentacja dla dip")
```




```r
# ======== Obliczanie centrów dla normals ============
norm_vec <- function(x) {
  sqrt(sum(x^2))
}

norm_vec2 <- function(x) {
  sqrt(sum(x^2))
}


normalized_centers <- t(apply(grupowanie_normals$centers, 1, function(row) row / norm_vec(row)))
Dip_ang <- acos(normalized_centers[, 3]) * (180 / pi)
Dip_dir <- atan2(normalized_centers[, 2], normalized_centers[, 1]) * (180 / pi)
Dip_dir <- ifelse(Dip_dir < 0, Dip_dir + 360, Dip_dir)

centers_data <- data.frame(
  Dip_ang = Dip_ang,
  Dip_dir = Dip_dir
)
write.table(centers_data, file = "C:/Users/Martynaa/Desktop/5_semestr/GIS/projekt/finalny/ClusterCenters
            sep = ",", row.names = FALSE, col.names = TRUE)

# ======== Obliczanie centrów dla dip ============
normalized_centers2 <- t(apply(grupowanie_dip$centers, 1, function(row) row / norm_vec2(row)))
Dip_ang2 <- acos(normalized_centers2[, 3]) * (180 / pi)
Dip_dir2 <- atan2(normalized_centers2[, 2], normalized_centers2[, 1]) * (180 / pi)
```

```r
Dip_dir2 <- ifelse(Dip_dir2 < 0, Dip_dir2 + 360, Dip_dir2)

centers_data2 <- data.frame(
  Dip_ang = Dip_ang2-90,
  Dip_dir = Dip_dir2
)
centers_data
```

```
##    Dip_ang  Dip_dir
## 1 0.246228 303.3932
## 2 5.191589 256.0471
## 3 0.700390 212.2405
```

```r
centers_data2
```

```
##     Dip_ang  Dip_dir
## 1 0.7804583 194.2332
## 2 0.5240318 256.3585
## 3 0.4896375 339.6984
```

```r
write.table(centers_data2, file = "C:/Users/Martynaa/Desktop/5_semestr/GIS/projekt/finalny/ClusterCente
            sep = ",", row.names = FALSE, col.names = TRUE)
```