Machine Learning - 1100-ML0ENG (Ćwiczenia informatyczne Z-23/24)

Partitioning Clustering

In any clustering method, we expect to produce maximally compact and maximally disjoint clusters.

Swiss dataset

R: Swiss Fertility and Socioeconomic Indicators (1888) Data (ethz.ch)

```
data(swiss)
summary(swiss)
df.scaled <- scale(swiss)
install.packages("cluster")
library(cluster)
```

Tendency of the dataset to clustering

```
install.packages("factoextra")
library(factoextra)
fviz_dist(dist(swiss))+
 labs(title = "Swiss")
```

A heatmap is a data visualisation technique that shows the magnitude of a phenomenon as a colour in two dimensions. There are two fundamentally different categories of heatmaps: the cluster heatmap and the spatial heatmap. In a clustered heatmap, magnitudes are arranged in a matrix of fixed-size cells whose rows and columns are discrete phenomena and categories, and the sorting of rows and columns is deliberate and somewhat arbitrary, in order to suggest clusters or to represent them as discovered through analysis.

```
install.packages("hopkins")
library(hopkins)
```

```
set.seed(123)
hopkins(swiss)
```

Number of clusters

Elbow method

#elbow method - basic
#type = b points connected by lines

```
k.max <- 15
wss <- sapply(1:k.max, function(k){kmeans(df.scaled , k, nstart = 50)$tot.withinss})
plot(1:k.max, wss, type = "b", pch = 19,
 xlab = "number of clusters - k",
 ylab = "Sum of squares of distances within a cluster")
```

```
# elbow method factoextra package
fviz_nbclust(df.scaled , kmeans, method = "wss") +
  labs(subtitle = "Ebow2")
```

Silhouette method

The silhouette method (silhouette), is a method that, for each observation, indicates how close an observation in a group is to all others in another, closest group. **The silhouette measure compares the average distance to elements in the same cluster with the average**

distance to elements in another cluster. The higher the value of this parameter, the better the clustering. The silhouette index works well in the k-means method, among others, where it is used to determine the optimal number of clusters and to assess clustering.

Each observation **x** is assigned a number $s_x = \frac{b(x) - a(x)}{max\{a(x), b(x)\}}$

where $a(x)$ is the average distance between x and all other points in the cluster, and $b(x)$ is the minimum of the average distances between x and points in other clusters.

If the value for an observation is close to 1, it means that the observation has been well assigned to the group in which it is located.
If it is close to -1, it means that the observation fits into a neighbouring group. Values close to zero indicate that the observation lies on the border between two clusters.
When determining the number of clusters, we look for the maximum.

```
fviz_nbclust(df.scaled, kmeans, method = "silhouette")+
 labs(subtitle = "Silhouette method - swiss dataset")
```

Gap statistics method

```
fviz_nbclust(swiss, kmeans, nstart = 25, method = "gap_stat", nboot = 50)+
 labs(subtitle = "Gap - swiss dataset")
```

NbClust function

```
library(NbClust)
#min.nc, max.nc - minimum and maximum number of clusters
liczba<- NbClust(df.scaled, distance="euclidean", min.nc=2, max.nc=10,
                 method="kmeans", index="all")
```

The k-means algorithm in R

In the **kmeans() function**, we specify **nstart = 25**. This means that R will sample 25 different random starting values for the means, and then choose the best result corresponding to the one with the least variation within the cluster.

The default value of **nstart=1**. It is recommended to determine the clusters in k-means with a large value of nstart, e.g. 25 or 50, to obtain more stable results.

```
set.seed(123) #random initialization trap
gr.km <- kmeans(df.scaled, centers = 3, nstart=25)
gr.km$cluster
gr.km$centers
gr.km$size
gr.km$withinss # shows how close the objects in the clusters are
```

We obtained the following clusters

```
swiss2<-cbind(swiss,gr.km$cluster)
aggregate(swiss2, by=list(cluster = gr.km$cluster), mean)
```

or equivalent

```
swiss%>%
 mutate(clusters=gr.km$cluster)%>%
 group_by(clusters)%>%
 summarise_all("mean")
```

We obtained 3 clusters.

- Let's start with the variable Catholics We see that the first group is mainly Catholics, the second group is Protestants, while in the third group we have a quarter of Catholics, the rest are Protestants.
- Variable Education. The third group has the highest level of education, while the first and second groups have low levels.

- Variable Examination. The lowest level of the military exam is in group one, in group two we have a medium level, while population three had the highest level of preparation for the military exam.
- Variable Agriculture. In group one we have the highest percentage of men working in agriculture, in group two we have half and half, while in group three the majority of residents lived and worked in the city.
- Variable Infant.Mortality. It has comparable levels in all groups and does not generally differentiate between groups.
- Fertility variable. Group one has the highest common standardised measure of fertility. group two is lower than group one but still high, group three has the lowest value.

Visualize Clustering Results

cluster package

```
clusplot(df.scaled, gr.km$cluster, color=TRUE, shade=TRUE,
 labels=2, lines=0)
```

factoextra package

Provides ggplot2-based visualization of partitioning methods including kmeans. Observations are represented by points in the plot, using principal components if ncol(data) > 2.

```
fviz_cluster(gr.km, data = df.scaled,geom = c("point", "text"), main = "Clusters in
Swiss")
```

```
#repel to avoid overplotting text labels,
```

```
fviz_cluster(gr.km, data = df.scaled,geom = c("point", "text"),repel = TRUE,
ellipse.type = "confidence",
 ellipse.level = 0.99, main = "Clusters in Swiss")
```

```
fviz_cluster(gr.km, data = df.scaled, choose.vars = c("Fertility", "Catholic"),
 main = "Clusters in Swiss",
 ggtheme = theme_classic())
```

```
fviz_cluster(gr.km, data = df1, choose.vars = c("Fertility", "Catholic"),
 stand = TRUE, geom = c("point", "text"),
 repel = TRUE, ellipse.type = "confidence",
 ellipse.level = 0.95, main = "Clusters in Swiss",
 ggtheme = theme_classic())
```

```
fviz_cluster(gr.km, data = df.scaled, choose.vars = c("Examination", "Catholic"),
 main = "Clusters in Swiss",
 ggtheme = theme_classic())
```

Validation of the groups obtained

After the clustering has been performed, the groups obtained should be evaluated. The most commonly used methods take into account the distances of objects from the centres (or centroids) of clusters in relation to the distances between clusters. WCSS errors and distances between clusters and the silhuuette measure can be compared.

Silhouette method

cluster package

```
kms = silhouette(gr.km$cluster,dist(df.scaled))
summary(kms)
plot(kms)
```

Eclust function

For validation, it is convenient to use the eclust() function, which allows the use of various clustering algorithms.

eclust function

```
gr.eclust<- eclust(swiss, "kmeans", k = 3, stand = TRUE, nstart = 25, graph = FALSE)
```

```
gr.eclust$cluster
gr.eclust$centers
gr.eclust$nbclust
```

```
#chart for clusters
fviz_cluster(gr.eclust, data = df.scaled,geom = c("point", "text"),repel = TRUE,
ellipse.type = "confidence", ellipse.level = 0.95, main = "eclust validation")
```

Silhouette method with eclust

```
fviz_silhouette(gr.eclust)
```

```
# Observations in the wrong group
silinfo <- gr.eclust$silinfo
silinfo
```

```
sil <- gr.eclust$silinfo$widths
negative <- which(sil$sil_width < 0)
sil[negative, , drop = FALSE]
rn<-rownames(sil[negative, , drop = FALSE] )
swiss[rn,]
```

Last modified: czwartek, 23 listopada 2023, 1:00

Accessibility settings

Przetwarzanie danych osobowych

Platformą administruje Komisja ds. Doskonalenia Dydaktyki wraz z Centrum Informatyki Uniwersytetu Łódzkiego Więcej

Informacje na temat logowania

Na platformie jest wykorzystywana metoda logowania za pośrednictwem Centralnego Systemu Logowania.

Studentów i pracowników Uniwersytetu Łódzkiego obowiązuje nazwa użytkownika i hasło wykorzystywane podczas logowania się do systemu USOSweb.

Deklaracja dostępności