About the method - cluster\_kmeans: partitions data into k groups by minimizing within-cluster variance. Sensitive to scale; normalization can improve results.

# Clustering - Kmeans  
  
# installation   
#install.packages("daltoolbox")  
  
# loading DAL  
library(daltoolbox)

Load sample data (iris).

# carregando conjunto de dados  
data(iris)

Configure K-means with k=3 (one cluster per species in iris).

# configuração do método de clusterização  
model <- cluster\_kmeans(k=3)

Fit the model and obtain cluster labels.

# ajuste do modelo e rotulagem  
model <- fit(model, iris[,1:4])  
clu <- cluster(model, iris[,1:4])  
table(clu)

## clu  
## 1 2 3   
## 50 62 38

External evaluation using true labels (Species).

# evaluate model using external metric  
eval <- evaluate(model, clu, iris$Species)  
eval

## $clusters\_entropy  
## # A tibble: 3 × 4  
## x ce qtd ceg  
## <fct> <dbl> <int> <dbl>  
## 1 1 0 50 0   
## 2 2 0.771 62 0.319   
## 3 3 0.297 38 0.0754  
##   
## $clustering\_entropy  
## [1] 0.3938863  
##   
## $data\_entropy  
## [1] 1.584963

Effect of normalization: compare results after min-max.

# Influence of normalization in clustering  
  
iris\_minmax <- transform(fit(minmax(), iris), iris)  
model <- fit(model, iris\_minmax[,1:4])  
clu <- cluster(model, iris\_minmax[,1:4])  
table(clu)

## clu  
## 1 2 3   
## 50 39 61

Re-evaluation with normalized data.

# evaluate model using external metric  
  
eval <- evaluate(model, clu, iris\_minmax$Species)  
eval

## $clusters\_entropy  
## # A tibble: 3 × 4  
## x ce qtd ceg  
## <fct> <dbl> <int> <dbl>  
## 1 1 0 50 0   
## 2 2 0.391 39 0.102  
## 3 3 0.777 61 0.316  
##   
## $clustering\_entropy  
## [1] 0.4177655  
##   
## $data\_entropy  
## [1] 1.584963