Week-2

Rajesh Kalakoti

2023-08-03

- Packages
  - devtools
  - tidyverse
  - here

```
library(here)
project_path <- here()
source(here("R","utils.R"))
source(here("R","distance_functions.R"))
```

```
data(iris)
iris_data <- iris[, 1:4]  # Selecting the first 4 columns (features)
# Calculate the Hopkins statistic using the hopkins_stat function
?hopkins_stat
```

```
## No documentation for 'hopkins_stat' in specified packages and libraries:
## you could try '??hopkins_stat'
```

```
hopkins_value <- hopkins_stat(iris_data)
print(hopkins_value)
```

```
## [1] 0.9964967
```

Clustering

Given a clustering $C = \{C_1, C_2, \dots, C_k\}$, we need some scoring function that evaluates its quality or goodness. This sum of squared errors scoring function is defined as:

$$W(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{i:C(i)=k}\|x_i - \bar{x}_k\|^2$$

The goal is to find the clustering that minimizes:

$$C^* = \arg\min_{C}\{W(c)\}$$

K-means employs a greedy iterative approach to find a clustering that minimizes loss function.

Algorithm 1: K-means Algorithm

Data: D, k, ε

K-means(D, k, ε):

$t \leftarrow 0$;

Randomly initialize k centroids: $\mu_1^t, \mu_2^t, \dots, \mu_n^t \in \mathbb{R}^d$;

repeat

  $t \leftarrow t + 1$;

  $C_i \leftarrow \emptyset$ for all $i = 1, \dots, k$

  /* Cluster assignment step */

  for $x_j \in D$ **do**

    $i^* \leftarrow \operatorname{argmin}_i \{||x_j - \mu_i^{t-1}||^2\}$;

    /* assign x_j to closest centroid */

    $C_{i^*} \leftarrow C_{i^*} \cup \{x_j\}$;

  end

  for $i = 1, .., k$ **do**

    $\mu_i^t \leftarrow \frac{1}{|C_i|} \sum_{x_j \in C_i} X_j$

  end

until $\sum_{i=1}^{k} ||\mu_i^t - \mu_i^{t-1}||^2 \leq \varepsilon$;