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"))
```

1 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:

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

1.1 Classification of clustering techniques

Most common clustering techniques may be classified as follows:

- **Representative Based Techniques**
  - k-means, k-medians, k-medoids, etc. Each cluster has a representative which is either the element of the data set or an element from the same space as all other elements of the dataset. Shape of the clusters is affected by the choice of distance function. Number of clusters is usually a hyperparameter.
- **Hierarchical Clustering Techniques**
  - Agglomerative and Divisive techniques. Not always relies on the distance function. Different levels of clustering granularity provide different provide different application specific insides.
- **Grid and Density Based Techniques**
  - Relies on the local density of the data points. Well suited for the clusters of irregular shapes.
- **Probabilistic Algorithms**
  - Examples: EM (Expectation-Maximization) and EM-like algorithms. Utilize probabilistic models for clustering

1.2 Hopkins Statistics

The Hopkins statistic (Lawson and Jurs 1990) is used to assess the clustering tendency of a data set by measuring the probability that a given data set is generated by a uniform data distribution

Let $\mathcal{D}$ be the data set to investigate and $\mathcal{R}$ is a representative sample of $\mathcal{D}$, of power r. $\mathcal{S}$ is a synthetic data set of r data points randomly generated from the same domain. Let $\alpha_1, \ldots \alpha_r$ be the distances of each point of R to the nearest neighbour in $\mathcal{D}$ and $\beta_1, \ldots \beta_r$ are the distances of each point of $\mathcal{S}$ to the nearest neighbour in $\mathcal{D}$. The Hopkins statistic is defined as follows:

$$H = \frac{\sum_{i=1}^{r} \beta_i}{\sum_{i=1}^{r} (\alpha_i + \beta_i)}$$

- **Hopkins Statistic**
  - Used to assess clustering tendency of a dataset. Measures likelihood of dataset having a cluster structure based on distances between data points.
  - Ranges between 0 and 1 .
- **Interpretation of Values**
  - Higher values of H indicate highly clustered data.

Example

```
# Load the Iris dataset
data(iris)

# Calculate the Hopkins statistic for the data columns.
hopkins_result <- hopkins_stat(iris[, 1:4])

# Print the calculated Hopkins statistic with a description
cat("Hopkins Statistic for the Iris dataset:", hopkins_result, "\n")
```

```
## Hopkins Statistic for the Iris dataset: 0.9979138
```

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**
    $i^* \leftarrow \text{argmin}_i\{||x_j - \mu_i^{t-1}||^2\}$;
    /* 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 $\sum_{i=1}^{k} ||\mu_i^t - \mu_i^{t-1}||^2 \leq \varepsilon$;