Package: clap (via r-universe)

September 4, 2024

Type Package

Title Detecting Class Overlapping Regions in Multidimensional Data

Version 0.1.0

Description The issue of overlapping regions in multidimensional data arises when different classes or clusters share similar feature representations, making it challenging to delineate distinct boundaries between them accurately. This package provides methods for detecting and visualizing these overlapping regions using partitional clustering techniques based on nearest neighbor distances.

License GPL-3

Encoding UTF-8

Imports mclust, FNN, dplyr, stats, rlang

Suggests ggplot2

RoxygenNote 7.3.1

URL https://github.com/pridiltal/clap

BugReports https://github.com/pridiltal/clap/issues

Repository https://pridiltal.r-universe.dev

RemoteUrl https://github.com/pridiltal/clap

RemoteRef HEAD

RemoteSha a0e22afe5634a09ac4bf1ac4d381b2be8106d7ab

Contents

clap *clap: Detecting Class Overlapping Regions in Multidimensional Dat*

Description

The issue of overlapping regions in multidimensional data arises when different classes or clusters share similar feature representations, making it challenging to delineate distinct boundaries between them accurately. This package provides methods for detecting and visualizing these overlapping regions using partitional clustering techniques based on nearest neighbor distances.

Author(s)

Priyanga Dilini Talagala _PACKAGE

compute_cluster_composition
Compute cluster composition and filter based on percentage

Description

This function computes the cluster composition based on the input object of class 'clap' returned by perform_clustering function. It merges the data and cluster assignments, computes cluster composition statistics including counts, IDs, and percentages, and filters based on the specified percentage threshold.

Usage

```
compute_cluster_composition(x)
```

Arguments

x	An object of class 'clap' returned by perform_clustering function, containing members (list of clusters), cluster_df (data frame of cluster assignments), and the original dataset.

Value

filtered data frame summarizing cluster composition with class 'clap'.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) {
  # Generate dummy data
  class1 <- matrix(rnorm(100, mean = 0, sd = 1), ncol = 2) +
    matrix(rep(c(1, 1), each = 50), ncol = 2)
  class2 <- matrix(rnorm(100, mean = 0, sd = 1), ncol = 2) +
    matrix(rep(c(-1, -1), each = 50), ncol = 2)
  datanew <- rbind(class1, class2)
  training <- data.frame(datanew, class = factor(c(rep(1, 50), rep(2, 50))))

  # Plot the dummy data to visualize overlaps
  p <- ggplot2::ggplot(training, ggplot2::aes(x = X1, y = X2, color = class)) +
    ggplot2::geom_point() +
    ggplot2::labs(title = "Dummy Data with Overlapping Classes")
  print(p)

  # Perform clustering
  cluster_result <- perform_clustering(training, class_column = class)
  # Compute cluster composition
  composition <- compute_cluster_composition(cluster_result)
}
```

---

extract_ids_vector *Extract and convert IDs to numeric vector*

Description

This function extracts IDs from a data frame containing filtered composition data and converts them into a numeric vector.

Usage

```
extract_ids_vector(composition)
```

Arguments

composition	An object of class 'clap' returned by 'compute_cluster_composition' function, containing cluster composition data including IDs.

Value

A numeric vector of IDs.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) {
  # Generate dummy data
  class1 <- matrix(rnorm(100, mean = 0, sd = 1), ncol = 2) +
    matrix(rep(c(1, 1), each = 50), ncol = 2)
```

```
  class2 <- matrix(rnorm(100, mean = 0, sd = 1), ncol = 2) +
    matrix(rep(c(-1, -1), each = 50), ncol = 2)
  datanew <- rbind(class1, class2)
  training <- data.frame(datanew, class = factor(c(rep(1, 50), rep(2, 50))))

  # Plot the dummy data to visualize overlaps
  p <- ggplot2::ggplot(training, ggplot2::aes(x = X1, y = X2, color = class)) +
    ggplot2::geom_point() +
    ggplot2::labs(title = "Dummy Data with Overlapping Classes")
  print(p)

  # Perform clustering
  cluster_result <- perform_clustering(training, class_column = class)
  # Compute cluster composition
  composition <- compute_cluster_composition(cluster_result)
  # Extract IDs to numeric vector
  ids_vector <- extract_ids_vector(composition)
  # Subset data based on extracted IDs
  overlapdata <- training[ids_vector, ]
  # Plot overlapping data points
  p2 <- p + ggplot2::geom_point(data = overlapdata, ggplot2::aes(X1, X2), colour = "black")
  print(p2)
}
```

`perform_clustering` *Perform clustering based on nearest neighbor distances*

Description

Perform clustering based on nearest neighbor distances

Usage

```
perform_clustering(data, class_column = NULL)
```

Arguments

`data`	A numeric matrix or data frame of data points.
`class_column`	A character string or unquoted name specifying the name of the column containing class labels.

Details

This function first removes the specified class column from the data, calculates the nearest neighbor distances, and then performs clustering using a radius based on the maximum nearest neighbor distance.

Value

An object of class 'clap' containing:

members A list of clusters with their respective data point IDs.

cluster_df A data frame with cluster assignments for each data point.

data The original dataset.

Examples

```
if (requireNamespace("ggplot2", quietly = TRUE)) {
  # Generate dummy data
  class1 <- matrix(rnorm(100, mean = 0, sd = 1), ncol = 2) +
    matrix(rep(c(1, 1), each = 50), ncol = 2)
  class2 <- matrix(rnorm(100, mean = 0, sd = 1), ncol = 2) +
    matrix(rep(c(-1, -1), each = 50), ncol = 2)
  datanew <- rbind(class1, class2)
  training <- data.frame(datanew, class = factor(c(rep(1, 50), rep(2, 50))))

  # Plot the dummy data to visualize overlaps
  p <- ggplot2::ggplot(training, ggplot2::aes(x = X1, y = X2, color = class)) +
    ggplot2::geom_point() +
    ggplot2::labs(title = "Dummy Data with Overlapping Classes")
  print(p)

  # Perform clustering
  cluster_result <- perform_clustering(training, class_column = class)
}
```

Index