Package 'fussclust'

June 2, 2026

Title Fuzzy Unsupervised and Semi-Supervised Clustering

Version 0.1.0

Description Methods for distance-based fuzzy unsupervised and semi-supervised clustering, including fuzzy and possibilistic models based on alternating optimization (AO) algorithm. The package introduces a vectorized estimation framework for prototype-based fuzzy clustering algorithms, enabling modular algorithm design and extensibility. It also supports storage and retrieval of intermediate AO optimization results for downstream analysis and processing. For more details see Kmita et al. (2024) <doi:10.1109/TFUZZ.2024.3370768>.

License MIT + file LICENSE

LazyData true

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports rdist

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Kamil Kmita [aut, cre, cph] (ORCID: <https://orcid.org/0000-0001-8829-2420>)

Maintainer Kamil Kmita <kamil.kmita17@gmail.com>

Repository CRAN

Date/Publication 2026-06-02 08:30:02 UTC

Contents

`calculate_evidence` *Calculates data evidence matrix E from distances matrix D.*

Description

Calculates data evidence matrix E from distances matrix D.

Usage

```
calculate_evidence(D)
```

Arguments

`D`	Distances matrix of size N x c.

Value

Matrix of size N x c.

dheve — *Creates DHE (stands for "distances horizontally exploded") and DVE (stands for "distances vertically exploded") matrices.*

Description

Creates DHE (stands for "distances horizontally exploded") and DVE (stands for "distances vertically exploded") matrices.

Usage

```
dheve(A, vertical)
```

Arguments

A	Matrix of size N x c.
vertical	Boolean switch. If TRUE, create DVE (vertical explosion). If FALSE, create DHE (horizontal explosion).

Value

Matrix of size Nc x c

estimate_super_T — *Estimated T matrix with typicalities in semi-supervised case.*

Description

Estimated T matrix with typicalities in semi-supervised case.

Usage

```
estimate_super_T(D, superF, alpha, gammas, b = 1)
```

Arguments

D	Distances matrix of size N x c.
superF	Binary supervision matrix of size N x c.
alpha	Scaling factor, a floating point > 0 regulating the impact of partial supervision.
gammas	a c-vector of cluster-specific gamma hyperparameters.
b	a scalar weighting the contribution of possibilistic membership in SPFCM (semi-supervised possibilistic fuzzy c-means) model. It is set to 1 by default for other semi-supervised models.

estimate_T *Estimated T matrix with typicalities in unsupervised case.*

Description

Estimated T matrix with typicalities in unsupervised case.

Usage

```
estimate_T(D, gammas)
```

Arguments

D	Distances matrix of size N x c.
gammas	a c-vector of cluster-specific gamma hyperparameters.

estimate_U *Estimated U matrix with memberships in semi-supervised case.*

Description

Estimated U matrix with memberships in semi-supervised case.

Usage

```
estimate_U(D, superF, alpha)
```

Arguments

D	Distances matrix of size N x c.
superF	Binary supervision matrix of size N x c.
alpha	Scaling factor, a floating point > 0 regulating the impact of partial supervision.

`estimate_V` *Equation to calculate clusters' prototypes matrix $\hat{V}$.*

Description

Equation to calculate clusters' prototypes matrix $\hat{V}$.

Usage

```
estimate_V(Phi, X)
```

Arguments

`Phi`	Matrix with weights of size N x c.
`X`	Matrix with predictors of size N x p.

Value

Clusters' prototypes matrix of size c x p.

`FCM` *Fuzzy C-Means clustering model*

Description

Fits a Fuzzy C-Means (FCM) clustering model using the Alternating Optimization algorithm.

Usage

```
FCM(
  X,
  C,
  U = NULL,
  max_iter = 200,
  conv_criterion = 1e-04,
  function_dist = rdist::cdist,
  store_history = FALSE
)
```

Arguments

X	A numeric feature matrix.
C	Integer specifying the number of clusters.
U	Optional initial membership matrix. Primarily intended for reproducibility purposes. If NULL (default), the algorithm uses a random initialization.
max_iter	Maximum number of iterations. Defaults to 200.
conv_criterion	Convergence threshold used at the end of each iteration of the Alternating Optimization algorithm.
function_dist	Optional distance function. The function must accept two matrices, X and V, with the same number of columns, and return a matrix of size nrow(X) x nrow(V) containing distances between each row of X and each row of V. For the Euclidean distance, the returned distances should not be squared. Defaults to rdist::cdist().
store_history	Logical indicating whether optimization histories should be stored. If FALSE, the returned object will contain NULL history fields. Defaults to TRUE.

Value

An object of class fcm containing:

U An $N \times C$ membership matrix.

V A $C \times p$ matrix of cluster prototypes.

function_dist The distance function used by the model.

counter Number of iterations performed until convergence.

U_history If store_history = TRUE, a list of length counter containing membership matrices estimated at each iteration; otherwise NULL.

V_history If store_history = TRUE, a list of length counter containing prototype matrices estimated at each iteration; otherwise NULL.

Phi_history If store_history = TRUE, a list of length counter containing phi-weight matrices estimated at each iteration; otherwise NULL.

References

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Springer US. https://doi.org/10.1007/978-1-4757-0450-1

Examples

```
X <- matrix(rnorm(100), ncol = 2)

model_fcm <- fussclust::FCM(
  X = X,
  C = 2
)

print(model_fcm$V)
```

gamma_fcm — *Aggregates elements of DHE and DVE matrices in a step to build evidence matrix E.*

Description

Aggregates elements of DHE and DVE matrices in a step to build evidence matrix E.

Usage

```
gamma_fcm(dhe, dve)
```

Arguments

dhe	DHE matrix of size Nc x c.
dve	DVE matrix of size Nc x c.

Value

Matrix of size Nc x 1.

init_gamma — *Initialization procedure to calculate values of gamma hyperparameters.*

Description

Initialization procedure to calculate values of gamma hyperparameters.

Usage

```
init_gamma(.model, .X)
```

Arguments

.model	estimated model of class fcm
.X	features matrix of size N x c

PCM *Possibilistic C-Means clustering model*

Description

Fits a Possibilistic C-Means (PCM) clustering model using the Alternating Optimization algorithm.

Usage

```
PCM(
  X,
  C,
  U = NULL,
  gammas = NULL,
  initFCM = NULL,
  max_iter = 200,
  conv_criterion = 1e-04,
  function_dist = rdist::cdist,
  store_history = FALSE
)
```

Arguments

X	A numeric feature matrix.
C	Integer specifying the number of clusters.
U	Optional initial membership matrix. Primarily intended for reproducibility purposes. If NULL (default), the algorithm uses a random initialization.
gammas	Optional vector of cluster-specific gamma hyperparameters. If NULL (default), the initialization strategy depends on the value of initFCM. If initFCM is NULL, a vector of ones is used. Otherwise, a Fuzzy C-Means model is first fitted, and the init_gamma() function is used to estimate the cluster-specific gamma hyperparameters.
initFCM	Optional fitted Fuzzy C-Means model used to initialize cluster-specific gamma hyperparameters via weighted averaging. If NULL (default), no preliminary Fuzzy C-Means initialization is used. If provided, this argument is effective only when gammas is NULL.
max_iter	Maximum number of iterations. Defaults to 200.
conv_criterion	Convergence threshold used at the end of each iteration of the Alternating Optimization algorithm.
function_dist	Optional distance function. The function must accept two matrices, X and V, with the same number of columns, and return a matrix of size nrow(X) x nrow(V) containing distances between each row of X and each row of V. For the Euclidean distance, the returned distances should not be squared. Defaults to rdist::cdist().
store_history	Logical indicating whether optimization histories should be stored. If FALSE, the returned object will contain NULL history fields. Defaults to TRUE.

Value

An object of class `pcm` containing:

U An $N \times C$ membership matrix.

V A $C \times p$ matrix of cluster prototypes.

function_dist The distance function used by the model.

counter Number of iterations performed until convergence.

gammas Vector of cluster-specific gamma hyperparameters.

U_history If `store_history = TRUE`, a list of length `counter` containing membership matrices estimated at each iteration; otherwise `NULL`.

V_history If `store_history = TRUE`, a list of length `counter` containing prototype matrices estimated at each iteration; otherwise `NULL`.

Phi_history If `store_history = TRUE`, a list of length `counter` containing phi-weight matrices estimated at each iteration; otherwise `NULL`.

References

Krishnapuram, R., & Keller, J. (1993). *A possibilistic approach to clustering*. IEEE Transactions on Fuzzy Systems, 1(2), 98–110. https://doi.org/10.1109/91.227387

Examples

```
X <- matrix(rnorm(100), ncol = 2)

model_pcm <- fussclust::PCM(
  X = X,
  C = 2,
  initFCM = TRUE
)

print(model_pcm$V)
```

`predict.ssfcm` *Predict method for* `ssfcm` *objects*

Description

Predicts cluster memberships for new observations using a fitted Semi-Supervised Fuzzy C-Means model.

Usage

```
## S3 method for class 'ssfcm'
predict(object, X, ...)
```

Arguments

object	An object of class ssfcm.
X	A numeric matrix of new observations with p columns.
...	Additional arguments. Currently ignored.

Value

A matrix of size $N \times C$ containing predicted cluster memberships, where C is the number of clusters.

Examples

```
X <- matrix(rnorm(100), ncol = 2)

superF <- matrix(0, nrow = nrow(X), ncol = 2)

superF[1:10, 1] <- 1
superF[11:20, 2] <- 1

model_ssfcm <- SSFCM(
  X = X,
  C = 2,
  superF = superF,
  alpha = 1
)

predict(model_ssfcm, matrix(rnorm(2), ncol = 2))
```

---

predict.sspcm *Predict method for* sspcm *objects*

Description

Predicts cluster memberships for new observations using a fitted Semi-Supervised Possibilistic C-Means model.

Usage

```
## S3 method for class 'sspcm'
predict(object, X, ...)
```

Arguments

object	An object of class sspcm.
X	A numeric matrix of new observations with p columns.
...	Additional arguments. Currently ignored.

Value

A matrix of size $N \times C$ containing predicted cluster memberships, where C is the number of clusters.

Examples

```
X <- matrix(rnorm(100), ncol = 2)

superF <- matrix(0, nrow = nrow(X), ncol = 2)

superF[1:10, 1] <- 1
superF[11:20, 2] <- 1

model_sspcm <- SSPCM(
  X = X,
  C = 2,
  superF = superF,
  initFCM = TRUE,
  alpha = 1
)

predict(model_sspcm, matrix(rnorm(2), ncol = 2))
```

SSFCM *Semi-Supervised Fuzzy C-Means clustering model*

Description

Fits a Semi-Supervised Fuzzy C-Means (SSFCM) clustering model using the Alternating Optimization algorithm.

Usage

```
SSFCM(
  X,
  C,
  U = NULL,
  max_iter = 200,
  conv_criterion = 1e-04,
  function_dist = rdist::cdist,
  store_history = FALSE,
  alpha = NULL,
  superF = NULL
)
```

Arguments

X	A numeric feature matrix.
C	Integer specifying the number of clusters.
U	Optional initial membership matrix. Primarily intended for reproducibility purposes. If NULL (default), the algorithm uses a random initialization.
max_iter	Maximum number of iterations. Defaults to 200.
conv_criterion	Convergence threshold used at the end of each iteration of the Alternating Optimization algorithm.
function_dist	Optional distance function. The function must accept two matrices, X and V, with the same number of columns, and return a matrix of size nrow(X) x nrow(V) containing distances between each row of X and each row of V. For the Euclidean distance, the returned distances should not be squared. Defaults to rdist::cdist().
store_history	Logical indicating whether optimization histories should be stored. If FALSE, the returned object will contain NULL history fields. Defaults to TRUE.
alpha	Positive scaling factor regulating the impact of partial supervision.
superF	Binary supervision matrix of the same dimensions as U, indicating the available partial supervision information.

Value

An object of class sspcm containing:

U An $N \times C$ memberships matrix.

V A $C \times p$ matrix of cluster prototypes.

function_dist The distance function used by the model.

counter Number of iterations performed until convergence.

alpha Value of scaling factor.

U_history If store_history = TRUE, a list of length counter containing membership matrices estimated at each iteration; otherwise NULL.

V_history If store_history = TRUE, a list of length counter containing prototype matrices estimated at each iteration; otherwise NULL.

Phi_history If store_history = TRUE, a list of length counter containing phi-weight matrices estimated at each iteration; otherwise NULL.

References

Kmita, K., Kaczmarek-Majer, K., & Hryniewicz, O. (2024). *Explainable Impact of Partial Supervision in Semi-Supervised Fuzzy Clustering*. IEEE Transactions on Fuzzy Systems, 1–10. https://doi.org/10.1109/TFUZZ.2024

Examples

```
X <- matrix(rnorm(100), ncol = 2)

superF <- matrix(0, nrow = nrow(X), ncol = 2)

superF[1:10, 1] <- 1
superF[11:20, 2] <- 1

model_ssfcm <- SSFCM(
  X = X,
  C = 2,
  superF = superF,
  alpha = 1
)

print(model_ssfcm$V)
```

SSPCM *Semi-Supervised Possibilistic C-Means clustering model*

Description

Fits a Semi-Supervised Possibilistic C-Means (SSPCM) clustering model using the Alternating Optimization algorithm.

Usage

```
SSPCM(
  X,
  C,
  U = NULL,
  gammas = NULL,
  initFCM = NULL,
  max_iter = 200,
  conv_criterion = 1e-04,
  function_dist = rdist::cdist,
  store_history = FALSE,
  alpha = NULL,
  superF = NULL
)
```

Arguments

X	A numeric feature matrix.
C	Integer specifying the number of clusters.
U	Optional initial membership matrix. Primarily intended for reproducibility purposes. If NULL (default), the algorithm uses a random initialization.

gammas	Optional vector of cluster-specific gamma hyperparameters. If NULL (default), the initialization strategy depends on the value of initFCM. If initFCM is NULL, a vector of ones is used. Otherwise, a Fuzzy C-Means model is first fitted, and the init_gamma() function is used to estimate the cluster-specific gamma hyperparameters.
initFCM	Optional fitted Fuzzy C-Means model used to initialize cluster-specific gamma hyperparameters via weighted averaging. If NULL (default), no preliminary Fuzzy C-Means initialization is used. If provided, this argument is effective only when gammas is NULL.
max_iter	Maximum number of iterations. Defaults to 200.
conv_criterion	Convergence threshold used at the end of each iteration of the Alternating Optimization algorithm.
function_dist	Optional distance function. The function must accept two matrices, X and V, with the same number of columns, and return a matrix of size nrow(X) x nrow(V) containing distances between each row of X and each row of V. For the Euclidean distance, the returned distances should not be squared. Defaults to rdist::cdist().
store_history	Logical indicating whether optimization histories should be stored. If FALSE, the returned object will contain NULL history fields. Defaults to TRUE.
alpha	Positive scaling factor regulating the impact of partial supervision.
superF	Binary supervision matrix of the same dimensions as U, indicating the available partial supervision information.

Value

An object of class sspcm containing:

U An $N \times C$ typicalities matrix.

V A $C \times p$ matrix of cluster prototypes.

function_dist The distance function used by the model.

counter Number of iterations performed until convergence.

gammas Vector of cluster-specific gamma hyperparameters.

alpha Value of scaling factor.

U_history If store_history = TRUE, a list of length counter containing membership matrices estimated at each iteration; otherwise NULL.

V_history If store_history = TRUE, a list of length counter containing prototype matrices estimated at each iteration; otherwise NULL.

Phi_history If store_history = TRUE, a list of length counter containing phi-weight matrices estimated at each iteration; otherwise NULL.

References

Kmita, K., Kaczmarek-Majer, K., & Hryniewicz, O. (2024). *Explainable Impact of Partial Supervision in Semi-Supervised Fuzzy Clustering*. IEEE Transactions on Fuzzy Systems, 1–10. https://doi.org/10.1109/TFUZZ.2024

Examples

```
X <- matrix(rnorm(100), ncol = 2)

superF <- matrix(0, nrow = nrow(X), ncol = 2)

superF[1:10, 1] <- 1
superF[11:20, 2] <- 1

model_sspcm <- SSPCM(
  X = X,
  C = 2,
  superF = superF,
  alpha = 1
)

print(model_sspcm$V)
```

`superFstruct_underimpact` *Binary supervision structure to reconstruct the issue of underimpact of partial supervision.*

Description

This dataset provides a concrete superivison structure: - 'superF' matrix of size 150 x 3 with partial supervision, - 'ind' vector with indices of unsupervised observations, - 'tind' vector with indicies of observations selected to be in the test dataset, - 'tclass' vector with class membership of the observations selected to be in the test dataset.

This supervision structure is meant to reproduce a particular realization of phenomenon of underimpact of partial supervision specific to the iris dataset.

Usage

```
data(superFstruct_underimpact)
```

Format

A list with: a matrix of size 150 x 3, and three vectors.

U_underimpact *Initialization matrix to analyze underimpact in iris data.*

Description

This dataset provides a concrete initialization of membership matrix specific to the iris data that exhibits the phenomenon of underimpact of partial supervision in semi-supervised fuzzy clustering.

Usage

```
data(U_underimpact)
```

Format

A matrix of size 150 x 3.

xi_fcm *Rearranges elements of input matrix from a block matrix with vertical blocks (column vectors) to a block matrix with horizontal blocks (row vectors).*

Description

Rearranges elements of input matrix from a block matrix with vertical blocks (column vectors) to a block matrix with horizontal blocks (row vectors).

Usage

```
xi_fcm(A, c)
```

Arguments

A	Matrix of size Nc x 1.
c	Number of columns in the wanted matrix. Associated with the number of clusters.

Value

Matrix of size N x c.

Index