Cluster Analysis: nagdmc_wcss

Purpose

nagdmc_wcss computes the within-cluster sum of squares.

Declaration

```
#include <nagdmc.h>
void nagdmc_wcss(long rec1, long nvar, long nrec, long dblk, double data[],
                 void (*dfun)(long, long, double [], char *, int *), char *comm,
                 long chunksize, long nxvar, long xvar[], long iwts, long k,
                 double c[], long ic[], double css[], long nic[],
                 double *tss, int *info);
```

Parameters

1: **rec1** – `long` *Input*

On entry: the index in the data of the first data record used in the analysis.

Constraint: **rec1** ≥ 0.

2: **nvar** – `long` *Input*

On entry: the number of variables in the data.

Constraint: **nvar** ≥ 1.

3: **nrec** – `long` *Input*

On entry: the number of consecutive records, beginning at **rec1**, used in the analysis.

Constraint: **nrec** > 1.

4: **dblk** – `long` *Input*

On entry: the total number of records in the data block.

Constraint: **dblk** $\geq$ **rec1** + **nrec**.

5: **data**[**dblk** $*$ **nvar**] – `double` *Input*

On entry: the data values for the jth variable (for $j = 0, 1, \ldots,$ **nvar**-1) are stored in **data**[$i*$**nvar**$+j$], for $i = 0, 1, \ldots,$ **dblk** $- 1$. When the data function is used, **data** is not referenced.

6: **dfun** – function supplied by user *External Procedure*

On entry: the pointer to a data function supplied by the user.

Constraint: if **dfun** is a valid pointer, **data** must be 0.

The specification of **dfun** is:

```
void dfun(long irec, long chunksize, double x[], char *comm, int *ierr)
```

1: **irec** – `long` *Input*

On entry: the index in the data of the first record returned.

2: **chunksize** – `long` *Input*

On entry: the number of consecutive records returned.

3: **x**[**chunksize**$*$**nvar**] – `double` *Output*

On exit: data values for the jth variable (for $j = 0, 1, \ldots,$ **nvar** $- 1$) must be returned in **x**[$i *$ **nvar** $+ j$], for $i = 0, 1, \ldots,$ **chunksize** $- 1$.

4: **comm** – `char *` *Input*

On entry: a communication parameter allowing additional information to be passed to **dfun**. This parameter is passed 'as is' through the calling function.

5: **ierr** – `int *` *Output*

On exit: if the value pointed to by **ierr** on return is greater than 100, the NAG DMC function will terminate immediately and **info** will point to this value.

7: **comm** – `char *` *Input*

On entry: a communication parameter allowing additional information to be passed to **dfun**. This parameter is passed 'as is' through the calling function.

8: **chunksize** – `long` *Input*

On entry: if the data function is used, the function inputs no more than **chunksize** data records at a time; otherwise **chunksize** is not referenced.

Constraint: if **dfun** $\neq 0$, **chunksize** ≥ 1.

9: **nxvar** – `long` *Input*

On entry: the number of variables in the analysis. If **nxvar** $= 0$, all variables in the data, excluding **iwts** (if **iwts** ≥ 0), are used in the analysis.

Constraint: $0 \leq$ **nxvar** $\leq$ **nvar**.

10: **xvar[nxvar]** – `long` *Input*

On entry: the indices indicating the position in **data** in which the variables are stored. If **nxvar** $= 0$ then **xvar** must be 0, and the indices of variables are given by $j = 0, 1, \ldots,$ **nvar** $- 1$ and $j \neq$ **iwts**.

Constraints: if **nxvar** > 0, $0 \leq$ **xvar**$[i] <$ **nvar**, for $i = 0, 1, \ldots,$ **nxvar** $- 1$; otherwise **xvar** must be 0.

11: **iwts** – `long` *Input*

On entry: the index in **data** in which the weights are stored. If **iwts** $= -1$, no weights are used.

Constraints: $-1 \leq$ **iwts** $<$ **nvar**; if **nxvar** > 0, **iwts** $\neq$ **xvar**$[i]$, for $i = 0, 1, \ldots,$ **nxvar** $- 1$.

12: **k** – `long` *Input*

On entry: the number of groups in the clustering.

Constraint: $0 <$ **k** $<$ **nrec**.

13: **c[k∗nvar]** – `double` *Input*

On entry: **c**$[i *$ **k** $+ j]$ contains the mean value of the jth variable for the ith group, for $j = 0, 1, \ldots,$ **nvar** $- 1$; for $i = 0, 1, \ldots,$ **k** $- 1$.

14: **ic[nrec]** – `long` *Input*

On entry: the allocation of data records to groups in the clustering.

Constraints: $0 \leq$ **ic**$[i - 1] <$ **k**, for $i = 1, 2, \ldots,$ **nrec**

15: **css[k]** – `double` *Output*

On exit: **css**$[i]$ contains the within-cluster sum of squares for the ith group, for $i = 0, 1, \ldots,$ **k** $- 1$.

16: **nic[k]** – `long` *Output*

On exit: **nic**$[i]$ contains the number of data records in the ith group, for $i = 0, 1, \ldots,$ **k** $- 1$.

17: **tss** – `double *` *Output*

On exit: the total within-cluster sum of squares for the clustering.

18: **info** – `int *` *Output*

On exit: **info** gives information on the success of the function call:

0: the function successfully completed its task.

i; $i = 1, 2, \ldots, 4, 6, 8, 9, \ldots, 12, 14$: the specification of the ith formal parameter was incorrect.

99: the function failed to allocate enough memory.

> 100: an error occurred in a function specified by the user.

Notation

nrec	the number of data record in the clustering, n.
nvar	the number of variables in the clustering, p.
iwts	if required, the index in **data** defining the weights w_i, for $i = 1, 2, \ldots, n$.
k	the number of groups in the clustering, k.
c	the group means for the clustering, $\bar{x}_{lj}$, for $j = 1, 2, \ldots, p$; $l = 1, 2, \ldots, k$.
ic	the allocation of data records to groups S_l, for $l = 1, 2, \ldots, k$.
css	the within-cluster sum of squares values for the k groups.
tss	the total within-cluster sum of squares, v.

Description

Let X be a set of n data records x_i on p variables, for $i = 1, 2, \ldots, n$. One measure of the quality of a clustering is the within-cluster sum of squares which measures the variance-covariance within a clustering and for the lth of k clusters is defined by:

$$u_l = \sum_{i \in S_l} \sum_{j=1}^{p} w_i (x_{ij} - \bar{x}_{lj})^2, \quad l = 1, 2, \ldots, k,$$

where x_{ij} is the value of variable j for the ith data record; w_i is the weight on the ith data record; S_l is the set of data records belonging to the lth cluster; and $\bar{x}_{lj}$ is the mean for the variable j over cluster l.

The total within-cluster sum of squares is given by:

$$v = \sum_{l=1}^{k} u_l,$$

where, according to this criterion, lower values of v represent higher quality clusterings than higher values.

References and Further Reading

None.

See Also

kmeans_ex.c an example calling program.