Outlier Detection: nagdmc_bacon

Purpose

nagdmc_bacon identifies data records that outlie a multivariate Normal distribution by using the Blocked Adaptive Computationally-efficient Outlier Nominator (BACON).

Declaration

```
#include <nagdmc.h>
void nagdmc_bacon(long rec1, long nvar, long nrec, long dblk, double data[],
                  long nxvar, long xvar[], double wt[], long c, double alpha,
                  int method, int opt, double mval, long maxit, double tol,
                  double indwt[], 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: 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** $- 1$.

6: **nxvar** – `long` *Input*

On entry: the number of variables in the analysis. If **nxvar** $= 0$, all variables in the data are used in the analysis.

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

7: **xvar**[**nxvar**] – `long` *Input*

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

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

8: **wt**[**dblk**] – `double` *Input*

On entry: **wt**[i] contains the weight on the ith data record, for $i = 0, 1, \ldots,$ **dblk** $- 1$. These weight values are used when computing the intial subset of data. If the weight on each record is equal, **wt** should be 0.

Constraint: if **wt** $\neq 0$, **wt**[i] ≥ 0.0, for $i = 0, 1, \ldots,$ **dblk** $- 1$; and the sum of weights must be greater than zero.

9: **c** – `long` *Input*

On entry: **c**$*p$ data records are included in intial good subset of data, for p variables in the analysis.

Constraint: $0 <$ **c** $<$ **nrec**$/p$.

Suggested value: **c** $= 3$.

10: **alpha** – `double` *Input*

On entry: the rejection level for the χ^2 distribution.

Constraint: $0.0 <$ **alpha** < 1.0.

Suggested value: **alpha** $= 1/$**nrec**.

11: **method** – `int` *Input*

On entry: the value of **method** determines how the initial subset of $\mathbf{c} * p$ data records is found. If **method** $= 0$, the Euclidean distance to the variable medians is used; otherwise **method** $= 1$ and the Mahalanobis distance from the sample mean is used. In both cases the $\mathbf{c} * p$ data records with the smallest distances are used for the initial subset of good data records.

Constraint: **method** $\in \{0, 1\}$.

12: **opt** – `int` *Input*

On entry: set **opt** $= 1$ if **data** contains missing values; otherwise **opt** must be set to zero. Setting **opt** $= 1$ will handle missing values in one of two ways, depending on the choice of method. If **method** $= 1$, the function imputes missing data values by using the EM algorithm; otherwise data records containing missing values will be excluded from the analysis.

Constraint: **opt** $\in \{0, 1\}$. if **opt** $= 0$, **mval** is not referenced; otherwise

13: **mval** – `double` *Input*

On entry: all values in **data** equal within machine precision to **mval** are considered missing from the analysis.

Suggested value: a value outside the interval $[a, b]$, where a and b are the minimum and maximum value in your data, respectively.

14: **maxit** – `long` *Input*

On entry: if **method** $= 1$ and **opt** $= 1$, the maximum number of iterations of the EM algorithm; otherwise **maxit** is not referenced.

Constraint: **maxit** ≥ 1.

15: **tol** – `double` *Input*

On entry: if **method** $= 1$ and **opt** $= 1$, the convergence tolerance of the EM algorithm; otherwise **tol** is not referenced.

Constraint: **tol** > 0.0.

16: **indwt**[**nrec**] – `double` *Output*

On exit: **indwt**$[i]$ contains the indicative weight on the ith data record in the analysis, for $i = 0, 1, \ldots,$ **nrec** $- 1$. The ith data record is an outlier if **indwt**$[i] = 0$, and an inlier if **indwt**$[i] > 0.0$. If **method** $= 1$ and **indwt**$[i] = -1$ the ith data record contains missing values and was not used in the analysis.

17: **info** – `int *` *Output*

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

0: the function successfully completed its task.

i; $i = 1, 2, 3, 4, 6, 7, \ldots, 12, \ldots, 14, 15$: the specification of the ith formal parameter was incorrect.

19: an error occured computing deviates of the χ^2 distrbution.

25: an error occured in the EM function.

30: the inverse of the variance-covariance matrix could not be computed; depending on your data, a higher value for **c** may be required.

42: all data records are outliers; the value of **alpha** is too high.

99: the function failed to allocate enough memory.

100: an internal error occurred during the execution of the function.

Notation

nrec	the number of data records, n.
data	the data set, X.
nxvar	determines the number of variables, p.
wt	if supplied, the weights w_i, for $i = 1, 2, \ldots, n$.
c	the value of c.
alpha	the value of rejection level α.
indwt	the indicative weights v_i, for $i = 1, 2, \ldots, n$.

Description

Let X be a set of n data records on p variables where the ith data record in X is denoted by $x_i \in \mathbb{R}^{1\times p}$, for $i = 1, 2, \ldots, n$. The method begins by searching for the cp data records closest to the centre of X, for a given scalar value c. Two distance measures are available. Firstly, the Mahalanobis distances from the sample mean $\bar{x} \in \mathbb{R}^{1\times p}$, given by:

$$(x_i - \bar{x})S^{-1}(x_i - \bar{x})^T, \quad i = 1, 2, \ldots, n,$$

where S is the sample variance-covariance. Secondly, the Euclidean squared distances from the medians of variables $m \in \mathbb{R}^{1\times p}$, given by:

$$(x_i - m)(x_i - m)^T, \quad i = 1, 2, \ldots, n.$$

The cp data records with the smallest distances constitute an initial good subset of data records, G.

BACON uses the following (asymptotic) property of multivariate Normal data in order to re-estimate the members of G. In general, let z be drawn at random from a p-dimensional multivariate Normal distribution with mean μ and variance-covariance Σ. Then the contour of constant probability density for a given value of a scalar e is defined by the z values that satisfy:

$$(z - \mu)\Sigma^{-1}(z - \mu)^T = e.$$

Setting e equal to the $(1 - \alpha)$th percentile of the χ^2 distribution with p degrees of freedom defines the contour containing $(1 - \alpha) \times 100\%$ of the probability, i.e., the ellipsoid containing z values that satisfy:

$$(z - \mu)\Sigma^{-1}(z - \mu)^T \leq \chi^2_{p,(1-\alpha)},$$

has probability $1 - \alpha$.

At the jth iteration BACON calculates the (weighted) sample mean $\bar{x}_G$ and variance-covariance S_G for data records in G and uses these estimates to calculate the n Mahalanobis distances:

$$d_i = (x_i - \bar{x}_G)S_G^{-1}(x_i - \bar{x}_G)^T, \quad i = 1, 2, \ldots, n.$$

BACON then considers each data record as a potential member of a new good subset H, initially set equal to the null set. Specifically, if $d_i < u\chi^2_{p,(1-\alpha)}$, the ith data record is included in H for a given rejection level α, where u is a small sample correction factor given by $u = s + t$ with:

$$s = 1 + \frac{p+1}{n-p} + \frac{1}{n-h-p},$$

$$t = \max\left[0, (h-r)/(h+r)\right],$$

and $h = \lfloor (n + p + 1)/2 \rfloor$.

If $|H| \neq |G|$, G is set equal to H and H to null, and the next iteration begins; otherwise the method halts and defines a set of indicative weights as follows:

$$v_i = \begin{cases} y, & x_i \in G \\ 0, & x_i \notin G \end{cases}, \quad i = 1, 2, \ldots, n,$$

where if user-supplied weights are available, $y = w_i$; otherwise $y = 1$. All data records not in G, i.e., those with $v_i = 0$, are defined to be outliers.

If the set of data X contains values missing at random, BACON handles these values in one of two ways. If the intial subset is determined by using Mahalanobis distances from the mean, missing values are imputed by the EM algorithm; otherwise the missing values are ignored when computing the Euclidean distances to the medians.

References and Further Reading

Billor N. Hadi A. and Velleman P. (2000) BACON: blocked adaptive computationally efficient outlier nominators *Comp. Stats. and Data Analysis* **34** 279–298.

See Also

nagdmc_impute_em	the EM data imputation function.
bacon_ex.c	the example calling program.
