Package ‘mvnmle’

May 19, 2018

Version 0.1-11.1

Title ML Estimation for Multivariate Normal Data with Missing Values

Author Kevin Gross, with help from Douglas Bates

Maintainer ORPHANED

Depends R (>= 1.2.0)

Description Finds the maximum likelihood estimate of the mean vector and variance-covariance matrix for multivariate normal data with missing values.

License GPL (>= 2)

Repository CRAN

Date/Publication 2018-05-19 16:32:22 UTC

X-CRAN-Original-Maintainer Kevin Gross <kevin_gross@ncsu.edu>

X-CRAN-Comment Orphaned on 2012-12-21 as the maintainer failed to respond to requests for an update for R >= 2.14.0

NeedsCompilation yes

R topics documented:

- apple .................................................. 2
- getclf ............................................... 2
- getstartvals ....................................... 3
- make.del .......................................... 4
- missvals .......................................... 5
- mlest ............................................. 6
- mysort ............................................ 7

Index 8
Description

The apple data frame provides the number of apples (in 100s) on 18 different apple trees. For 12 trees, the percentage of apples with worms (x 100) is also given.

Format

This data frame contains the following columns:

- size hundreds of apples on the tree.
- worms percentage (x100) of apples harboring worms.

Details

These data constitute Table 6.1 in Little and Rubin (1987), adapted from Table 6.9.1 of Snedecor and Cochran (1967).

Source


Examples

```r
library(mvnmle)
data(apple)
mlest(apple)
```

getclf

Create likelihood function for multivariate data with missing values.

Description

getclf returns a function proportional to twice the negative log likelihood function for multivariate normal data with missing values. This is a private function used in mlest.

Usage

```r
getclf(data, freq)
```
getstartvals

Arguments

data  A data frame sorted so that records with identical patterns of missingness are grouped together.

freq  An integer vector specifying the number of records in each block of data with identical patterns of missingness.

Details

The argument of the returned function is the vector of parameters. The parameterization is: mean vector first, followed by the log of the diagonal elements of the inverse of the Cholesky factor, and then the elements of the inverse of the Cholesky factor above the main diagonal. These off-diagonal elements are ordered by column (left to right), and then by row within column (top to bottom).

Value

A function proportional to twice the negative log likelihood of the parameters given the data.

References


See Also

mlest

getstartvals(x, eps=0.001)

Arguments

x  Multivariate data, potentially with missing values.

eps  All eigenvalues of the variance-covariance matrix less than eps times the smallest positive eigenvalue are set to eps times the smallest positive eigenvalue.
make.del

Details
Starting values for the mean vector are simply sample means. Starting values for the variance-covariance matrix are derived from the sample variance-covariance matrix, after setting eigenvalues less than \( \text{eps} \) times the smallest positive eigenvalue equal to \( \text{eps} \) times the smallest positive eigenvalue to enforce positive definiteness.

Value
A numeric vector, containing the mean vector first, followed by the log of the diagonal elements of the inverse of the Cholesky factor of the adjusted sample variance-covariance matrix, and then the elements of the inverse of the Cholesky factor above the main diagonal. These off-diagonal elements are ordered by column (left to right), and then by row within column (top to bottom).

See Also
mlest

Description
make.del takes a parameter vector of length \( k \times (k + 1)/2 \) and returns the upper triangular \( k \times k \) matrix \( \Delta \). make.del is a private function intended for use inside mlest.

Usage
make.del(pars)

Arguments
pars A length \( k \times (k + 1)/2 \) numerical vector giving the elements of \( \Delta \).

Details
The first \( k \) elements of pars are the log of the diagonal elements of \( \Delta \). The next \( k \times (k - 1)/2 \) elements are the elements above the main diagonal of \( \Delta \), ordered by column (left to right), and then by row within column (top to bottom). That is to say, if \( \Delta_{ij} \) is the element in the \( i \)th row and \( j \)th column of \( \Delta \), then the order of the parameters is \( \Delta_{11}, \Delta_{22}, \ldots, \Delta_{kk}, \Delta_{12}, \Delta_{13}, \Delta_{23}, \Delta_{14}, \ldots, \Delta_{(k-1)k} \).

Value
An upper triangular \( k \times k \) matrix.

References
**missvals**

See Also

`mlest`

---

**missvals**  
*A multivariate data set with missing values.*

### Description

The `missvals` data frame has 13 rows and 5 columns. These are data from Draper and Smith (1968), and are included to demonstrate ML estimation of mean and variance-covariance parameters of multivariate normal data when some observations are missing.

### Format

This data frame contains the following columns:

- `x1,x2,x3,x4,x5` numeric vectors

### Details

These data constitute Table 6.4 in Little and Rubin (1987). They are analyzed both in Rubin (1976) and Little and Rubin (1987).

### Source


### Examples

```r
library(mvnmle)
data(missvals)
mlest(missvals, iterlim=400)
```
ML Estimation of Multivariate Normal Data

Description

Finds the maximum likelihood estimates of the mean vector and variance-covariance matrix for multivariate normal data with (potentially) missing values.

Usage

mlest(data, ...)

Arguments

data A data frame or matrix containing multivariate normal data. Each row should correspond to an observation, and each column to a component of the multivariate vector. Missing values should be coded by 'NA'.

... Optional arguments to be passed to the nlm optimization routine.

Details

The estimate of the variance-covariance matrix returned by mlest is necessarily positive semi-definite. Internally, nlm is used to minimize the negative log-likelihood, so optional arguments may be passed to nlm which modify the details of the minimization algorithm, such as iterlim. The likelihood is specified in terms of the inverse of the Cholesky factor of the variance-covariance matrix (see Pinheiro and Bates 2000).

mlest cannot handle data matrices with more than 50 variables. Each variable must also be observed at least once.

Value

muhat MLE of the mean vector.
sigmahat MLE of the variance-covariance matrix.
value The objective function that is minimized by nlm. Is is proportional to twice the negative log-likelihood.
gradienyear The curvature of the likelihood surface at the MLE, in the parameterization used internally by the optimization algorithm. This parameterization is: mean vector first, followed by the log of the diagonal elements of the inverse of the Cholesky factor, and then the elements of the inverse of the Cholesky factor above the main diagonal. These off-diagonal elements are ordered by column (left to right), and then by row within column (top to bottom).
stop.code The stop code returned by nlm.
iterations The number of iterations used by nlm.
**mysort**

**References**


**See Also**

nlm

**Examples**

```r
library(mvnmle)

data(apple)
mlest(apple)

data(missvals)
mlest(missvals, iterlim=400)
```

---

**mysort**

Sort a multivariate data matrix according to patterns of missingness.

**Description**

`mysort` sorts a multivariate data matrix so that records with identical patterns of missingness are adjacent to one another. `mysort` is a private function used inside of `mlest`.

**Usage**

```r
mysort(x)
```

**Arguments**

- **x**
  
  A multivariate data matrix. Rows correspond to individual records and columns correspond to components of the multivariate vector.

**Value**

- **sorted.data**
  
  A matrix of the same size as `x` but with the rows re-arranged so that records with identical patterns of missingness are adjacent to one another.

- **freq**
  
  An integer vector giving the number of records in each block of rows with a unique pattern of missingness. The first element in `freq` counts the number of rows in the top block of `sorted.data`, and so on.

**See Also**

mlest
Index

*Topic algebra
  make.del, 4
*Topic datasets
  apple, 2
  missvals, 5
*Topic multivariate
  getclf, 2
  getstartvals, 3
  mlest, 6
  mysort, 7

apple, 2

getclf, 2
getstartvals, 3

make.del, 4
missvals, 5
mlest, 3–5, 6, 7
mysort, 7

nlm, 7