

Package ‘selectiongain’

April 19, 2009

Type Package

Title gain of a model selection

Version 1.0

Date 2008-11-24

Author Xuefei Mi, Xuefei Mi, Friedrich Utz

Maintainer Xuefei Mi <mi@pz.uni-hohenheim.de>

Depends R (>= 1.8.0), mvtnorm

Description This package calculate the gain from a model selection. It uses part of Tallis’ algorithm (1961). The original algorithm is to calculate the moment generating function of the truncated Mvtnorm distribution. There will be a Mvtmoment package for the whole Tallis’ algorithm.

License GPL-2

Repository CRAN

Date/Publication 2008-11-24 19:54:19

R topics documented:

selectiongain-package	2
calculatefromalpha	2
multistageselection	4

Index	7
--------------	----------

selectiongain-package
gain of selection

Description

This package calculate the gain from a selection. It uses part of Tallis' algorithm (1961).

The original algorithm is to calculate the moment generating function of the truncated Mvtnorm distribution.

There will be a Mvtmoment package for the whole Tallis' algorithm.

Details

Package: selectiongain
Type: Package
Version: 1.0
Date: 2008-11-24
License: GPL

the most important function is "multistageselection"

Author(s)

Xuefei Mi, Friedrich Utz Maintainer:Xuefei Mi <mi@pz.uni-hohenheim.de>

calculatefromalpha *calculate the quantile for given alpha*

Description

this function can be considered as similar as qmvnorm. but for qmvnorm, a uniform alpha is given and the uniform quantiles are calculated for this alpha. in our function a vector of alpha is given, and the quantiles are calculate one by one to satisfy thesis given alpha.

e.g. alpha2= pmvnorm(quantile1,quantile2) now given alpha2, quantile1. we want to know quantile2. this is kind of Recursion Procedure.

Usage

```
calculatefromalpha(alpha, dim, corr)
```

Arguments

alpha	Probability vector for random variable X. $\alpha(i) = pmvnorm(quantile(1), \dots, quantile(i))$ in plants breeding, it is also called selection fraction. here we have $\alpha(\hat{i}) = N_{(i+1)}/N(i)$, which is the elimination rate from generation i to generation $i+1$.
dim	dimension of the random variable X.
corr	correlation matrix

Details

more details

Value

the out come is a vector for the quantiles

Note

further notes

Author(s)

Xuefei Mi

References

Tallis 1961, Utz1969

See Also

selectiongain()

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##-- or do help(data=index) for the standard data sets.

## The function is currently defined as
function(alpha,dim,corr)
{
  qnormbar<-function(kwanted,koutput,corr,howmanyk,alpha)
  {
    corr2dim=corr[1:howmanyk,1:howmanyk]

    tempofk1=koutput[1:c(howmanyk-1)]

    alphatotal=1
    for (i in 1:howmanyk)
```

```

    {
      alphanormal=alphanormal*alpha[i]
    }

    pmvnorm(lower = c(tempofk1,kwanted), upper = rep(Inf,howmanyk), mean = rep(0,howmanyk),
            corr = corr2dim) - alphanormal
  }

  koutput=0
  for (i in 1: dim )
  {
    if (i == 1)
    {
      koutput[1]= qnorm(alpha[1],lower.tail = FALSE)[[1]][1]
    }else
    {
      koutput[i]= uniroot(qnormbar, interval=c(-100,100), koutput=koutput, corr, i, alpha) [[1]]
    }
  }

  koutput
}

```

multistageselection

multistageselection

Description

it is a function which calculates multi-stage selection gain. the gain can be calculated from a momentum function of multi-variate normal distribution.

Usage

```
multistageselection(k, corr, alphaofx, sumdimofxandy)
```

Arguments

k	the lower bound of the integral, the first element of k is the lower bound of the response variable, Y, which is usually set as -Inf
corr	correlation matrix
alphaofx	selected fraction
sumdimofxandy	dimension of k, which is dimension of x plus 1(y has one dimension).

Details

Tallis, 1961; Utz, 1984

Value

the value returned, is the the gain of the selection.

Note

further notes

Author(s)

Xuefei, Mi

References

Tallis, 1965; Utz, 1984

See Also

link

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##-- or do help(data=index) for the standard data sets.

## The function is currently defined as
function(k,corr,alphaofx,sumdimofxandy)
{
  dim=sumdimofxandy
  A=array(0,c(dim,dim))
  for (i in 1 : dim)
  {
    for (j in 1 : dim)
    {
      if(i!=j)
      {
        A[i,j]= (k[j]-corr[i,j]*k[i])/ (1-corr[i,j]^2)^0.5
      }
    }
  }
  part.corr=array(1,c(dim,dim,dim))
  for (i in 1 : dim)
```

```

{
  for (j in 1 : dim)
  {
    for (q in 1 : dim)
    {
      if(i!=j && q!=j && i!=q)
        { part.corr[i,j,q]= (corr[i,j]-corr[i,q]*corr[j,q])/ ((1-corr[i,q]^2)^0.5 * (1-corr[j,q]^2)^0.5) }
    }
  }
}

```

```
j3q<-function (q,A,part.corr,dim)
```

```

{
  lower=A[q,-q]
  corr= part.corr[-q,-q,q]
  output=pmvnorm(lower = lower, upper = rep(Inf,c(dim-1)), mean = rep(0, length(lower)),
  corr = corr, sigma = NULL, algorithm = GenzBretz())
  output
}

```

```
calculatx1<-function(A,part.corr,dim,corr,k,alpha3)
```

```

{
  output=0
  i=1
  for (i in 1 : dim)
  {
    output= output+ corr[1,i]*dnorm(k[i])*j3q(i,A,part.corr,dim)
  }
  output=output/alpha3
  output
}

```

```
calculatx1(A=A,part.corr=part.corr,dim=dim,corr=corr,k=k,alpha3=alphaofx)
```

```

}

```

Index

*Topic **package**

 selectiongain-package, 1

calculatefromalpha, 2

multistageselection, 4

selectiongain

 (selectiongain-package), 1

selectiongain-package, 1