

Package: metabup (via r-universe)

August 22, 2024

Type Package

Title Bayesian Meta-Analysis Using Basic Uncertain Pooling

Version 0.1.3

Description Contains functions that allow Bayesian meta-analysis (1) with binomial data, counts(y) and total counts (n) or, (2) with user-supplied point estimates and associated variances. Case (1) provides an analysis based on the logit transformation of the sample proportion. This methodology is also appropriate for combining data from sample surveys and related sources. The functions can calculate the corresponding similarity matrix. More details can be found in Cahoy and Sedransk (2023), Cahoy and Sedransk (2022) <[doi:10.1007/s42519-018-0027-2](https://doi.org/10.1007/s42519-018-0027-2)>, Evans and Sedransk (2001) <[doi:10.1093/biomet/88.3.643](https://doi.org/10.1093/biomet/88.3.643)>, and Malec and Sedransk (1992) <[doi:10.1093/biomet/79.3.593](https://doi.org/10.1093/biomet/79.3.593)>.

License GPL (>= 3)

Encoding UTF-8

Imports stats, ggplot2, partitions

LazyData true

RoxygenNote 7.2.1

Depends R (>= 2.10)

Repository <https://dcahoy.r-universe.dev>

RemoteUrl <https://github.com/dcahoy/metabup>

RemoteRef HEAD

RemoteSha b1072a380f8ca9eef5212ce52955fe683aab0d2d

Contents

metabup	2
Index	5

metabup

*Bayesian meta-analysis using basic uncertain pooling.***Description**

Bayesian meta-analysis analysis (1) with binomial data, counts(y) and total counts (n) or, (2) with user-supplied estimates and associated variances. Case (1) provides an analysis based on the logit transformation of the sample proportion. This methodology is also appropriate for combining data from sample surveys and related sources.

Usage

```
metabup(y, nv, type = NULL, d2max = NULL, ngrid2 = NULL, N = NULL)
```

Arguments

y	vector of counts or effect estimates.
nv	vector of total counts n (if y 's are counts) or variances associated with the estimates (if y 's are estimates).
$type$	assumes a single value: 1 (counts and total counts as inputs) or 2 (estimates and variances as inputs). Default is type 2.
$d2max$	maximum value of the prior variance δ^2 to be used in the grid sampling. Default is <code>.Machine\$double.xmin</code> .
$ngrid2$	number of grid points for the prior variance. Default is 1000 if $d2max > .Machine$double.xmin$.
N	sample size to be drawn from the partition- δ^2 grid. Default is 10000.

Value

list consisting of the sample and posterior effect estimates and standard deviations, the partitions with the largest posterior probabilities, and the similarity matrix.

References

Cahoy and Sedransk (2023). *Combining data from surveys and related sources*. *Surv. Methodol.*, To appear.

Cahoy and Sedransk (2022). *Bayesian inference for asymptomatic COVID-19 infection rates*. *Stat Med*, 41(16):3131-3148, <doi:10.1002/sim.9408>

Evans and Sedransk (2001). *Combining data from experiments that may be similar*. *Biometrika*, 88(3):643-656, <doi:10.1093/biomet/88.3.643>

Malec and Sedransk (1992). *Bayesian methodology for combining the results from different experiments when the specifications for pooling are uncertain*. *Biometrika*, 79(3):593-601, <doi:10.1093/biomet/79.3.593>

Examples

```

y=c(4, 18, 40, 130)
nv=c(13, 83, 60,166)
require(partitions)
out=metabup(y, nv, type=1)

#estimates
out[1]
#the partitions with the largest posterior p(g|y)
out[2:3]

#plotting similarity matrix
require(ggplot2)
L=length(y)
indmat=matrix(unlist(out[4]),ncol=L, byrow=FALSE)
psingle<-rep(0,L)
sim_mat<-indmat +t(indmat)
diag(sim_mat)<-psingle #rep(1,L)
xy=expand.grid(x=1:L, y=1:L)
index=as.vector( sim_mat)
mat_data = cbind(xy,index)
brlab<-round(seq(0, max(sim_mat)+0.05, length.out=4),2)
p <- ggplot(data = mat_data) + # Set data
geom_tile(aes(x = x, y = y, fill =index)) +
scale_fill_gradientn(colours=rev(heat.colors(5)),na.value = "transparent",
breaks=brlab,labels=brlab,
limits=c(0,max(sim_mat)+0.075))+
scale_x_continuous(name="Study Number", breaks=1:L, limits=c(0.5,+L+0.5)) +
scale_y_continuous(name="", breaks=1:L, limits=c(0.5,L+0.5))
p + theme(axis.title = element_text() )

ph=c(31, 21.7, 66.7,78.3)/100
y=log(ph/(1-ph))
n=c(13, 83, 60,166)
nv=1/(ph*(1-ph)*n) #variance
require(partitions)
out=metabup(y, nv)

#estimates
out[1]
#the partitions with the largest posterior p(g|y)
out[2:3]

#plotting similarity matrix
require(ggplot2)
L=length(y)
indmat=matrix(unlist(out[4]),ncol=L, byrow=FALSE)
psingle<-rep(0,L)
sim_mat<-indmat +t(indmat)

```

```
diag(sim_mat)<-psingle #rep(1,L)
xy=expand.grid(x=1:L, y=1:L)
index=as.vector( sim_mat)
mat_data = cbind(xy,index)
brlab<-round(seq(0, max(sim_mat)+0.05, length.out=4),2)
p <- ggplot(data = mat_data) + # Set data
geom_tile(aes(x = x, y = y, fill =index)) +
scale_fill_gradientn(colours=rev(heat.colors(5)),na.value = "transparent",
breaks=brlab,labels=brlab,
limits=c(0,max(sim_mat)+0.075))+
scale_x_continuous(name="Study Number", breaks=1:L, limits=c(0.5,+L+0.5)) +
scale_y_continuous(name="", breaks=1:L, limits=c(0.5,L+0.5))
p + theme(axis.title = element_text() )
```

Index

metabup, [2](#)