

# Package ‘PAMA’

April 9, 2020

**Title** Rank Aggregation with Partition Mallows Model

**Version** 0.1.1

## Description

Rank aggregation aims to achieve a better ranking list given multiple observations. 'PAMA' implements Partition-Mallows model for rank aggregation. Both Bayesian inference and Maximum likelihood estimation (MLE) are provided. It can handle partial list as well. When covariate information is available, this package can make inference by incorporating the covariate information. More information can be found in the paper "Integrated Partition-Mallows Model and Its Inference for Rank Aggregation". The paper is not yet published.

**Depends** R (>= 3.1.0), PerMallows, mc2d, stats

## Imports

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.0.1

**Suggests** knitr, rmarkdown

**NeedsCompilation** no

**Author** Wanchuang Zhu [cre, aut]

**Maintainer** Wanchuang Zhu <andy.chou.sub@gmail.com>

**Repository** CRAN

**Date/Publication** 2020-04-09 08:50:15 UTC

## R topics documented:

NBANFL	2
PAMA.B	2
PAMA.Cov	3
PAMA.F	4
PAMA.PL	5

<b>Index</b>	<b>6</b>
--------------	----------

---

`NBANFL`*Dataset of NBA and NFL.*

---

**Description**

Dataset of NBA and NFL.

**Usage**

```
NBANFL()
```

**Value**

List. It contains Bayesian posterior samples of all the parameters and log-likelihood.

1. NBA: full ranking lists of NBA power ranking.
2. NBAPL: The partial lists of NBA power ranking
3. NFLdata: all the ranking lists of NFL data
4. NFLcov: covariates of NFL players

**Examples**

```
NBANFL()
```

---

`PAMA.B`*This function implements Bayesian inference of PAMA model.*

---

**Description**

This function implements Bayesian inference of PAMA model.

**Usage**

```
PAMA.B(datfile, nRe, iter = 1000)
```

**Arguments**

<code>datfile</code>	A matrix or dataframe. This is the data where our algorithm will work on. Each column denotes a ranker's ranking. The data should be in entity-based format.
<code>nRe</code>	A number. Number of relevant entities
<code>iter</code>	A number. Number of iterations of MCMC

**Value**

List. It contains Bayesian posterior samples of all the parameters and log-likelihood.

1. I.mat: posterior samples of I
2. phi.mat: posterior samples of phi
3. smlgamma.mat: posterior samples of gamma
4. l.mat: posterior samples of log-likelihood

**Examples**

```
dat=t(PerMallows::rmm(10,1:20,0.5))
PAMA.B(dat,10,iter=10)
PAMA.B(dat,10,iter=1000)
```

---

PAMA.Cov	<i>This function implements Bayesian inference of PAMA model with covariates.</i>
----------	---

---

**Description**

This function implements Bayesian inference of PAMA model with covariates.

**Usage**

```
PAMA.Cov(datfile, Covdatfile, nRe, iter)
```

**Arguments**

datfile	A matrix or dataframe. This is the data where our algorithm will work on. Each column denotes a ranker's ranking. The data should be in entity-based format.
Covdatfile	A matrix or dataframe. Each column denotes a covariate.
nRe	A number. Number of relevant entities
iter	A number. Number of iterations of MCMC. Defaulted as 1000.

**Details**

The covariates are incorporated in the PAMA framework as indicators of groupmember. That is covariates are associated to group members via a logistic regression.

**Value**

List. It contains Bayesian posterior samples of all the parameters and log-likelihood.

1. I.mat: posterior samples of I
2. phi.mat: posterior samples of phi
3. smlgamma.mat: posterior samples of gamma
4. l.mat: posterior samples of log-likelihood.
5. theta.mat: posterior samples of coefficients of covariates.

**Examples**

```
a=NBANFL()
PAMA.Cov(t(a$NFLdata),a$NFLcov,nRe=10,iter=10)
PAMA.Cov(t(a$NFLdata),a$NFLcov,nRe=10,iter=1000)
```

---

PAMA.F	<i>This function implements Maximum Likelihood estimation of PAMA model.</i>
--------	--

---

**Description**

This function implements Maximum Likelihood estimation of PAMA model.

**Usage**

```
PAMA.F(datfile, nRe, threshold, iter = 1000)
```

**Arguments**

datfile	A matrix or dataframe. This is the data where our algorithm will work on. Each row denotes a ranker's ranking. The data should be in entity-based format.
nRe	A number. Number of relevant entities.
threshold	A number(positive). The stopping threshold in determining convergence of MLE. if the two consecutive iterations of log-likelihood is smaller than threshold, then the convergence achieves.
iter	A number. Numner of iterations of MCMC.

**Value**

List. It contains MLE of all the parameters and log-likelihood.

1. I.mat: samples of I
2. phi.mat: samples of phi.
3. smlgamma.mat: samples of gamma
4. l.mat: samples of log-likelihood

**Examples**

```
a=NBANFL()
PAMA.F(a$NBA,nRe=10,threshold=0.1,iter=100)
```

---

PAMA.PL	<i>This function implements Bayesian inference of PAMA model with partial lists.</i>
---------	--

---

### Description

This function implements Bayesian inference of PAMA model with partial lists.

### Usage

```
PAMA.PL(datfile, PLdatfile, nRe, iter)
```

### Arguments

datfile	A matrix or dataframe. This is the data where our algorithm will work on. Each column denotes a ranker's ranking. The data should be in entity-based format.
PLdatfile	A matrix or dataframe. It contains all the partial lists. Each column denotes a partial list.
nRe	A number. Number of relevant entities.
iter	A number. Number of iterations of MCMC. Defaulted as 1000.

### Details

The partial lists are handle by Data Augmentation strategy.

### Value

List. It contains Bayesian posterior samples of all the parameters and log-likelihood.

1. I.mat: posterior samples of I
2. phi.mat: posterior samples of phi
3. smlgamma.mat: posterior samples of gamma
4. l.mat: posterior samples of log-likelihood.

### Examples

```
a=NBANFL()  
PAMA.PL(a$NBA,a$NBAPL,nRe=10,iter=1)  
PAMA.PL(a$NBA,a$NBAPL,nRe=10,iter=100)
```

# Index

NBANFL, [2](#)

PAMA . B, [2](#)

PAMA . Cov, [3](#)

PAMA . F, [4](#)

PAMA . PL, [5](#)