Package ‘bamp’

October 18, 2022

Type Package

Title Bayesian Age-Period-Cohort Modeling and Prediction

Version 2.1.3

Date 2022-10-17

Author Volker Schmid [aut, cre],
Florian Geressen [ctb],
Leonhard Held [ctb],
Evi Rainer [ctb]

Maintainer Volker Schmid <volker.schmid@lmu.de>

Description Bayesian Age-Period-Cohort Modeling and Prediction using efficient Markov Chain Monte Carlo Methods. This is the R version of the previous BAMP software as described in Volker Schmid and Leonhard Held (2007) <DOI:10.18637/jss.v021.i08> Bayesian Age-Period-Cohort Modeling and Prediction - BAMP, Journal of Statistical Software 21:8. This package includes checks of convergence using Gelman’s R.

License GPL-3

Encoding UTF-8

Depends R (>= 3.5.0)

Imports coda, graphics, parallel, stats, abind

LazyData true

RoxygenNote 7.2.1

VignetteBuilder knitr, R.rsp

Suggests knitr, rmarkdown, R.rsp

URL https://volkerschmid.github.io/bamp/

BugReports https://github.com/volkerschmid/bamp/issues

NeedsCompilation yes

Repository CRAN

Date/Publication 2022-10-18 08:02:47 UTC
**R topics documented:**

- `apc` .................................................. 2
- `apc.data` .......................................... 3
- `apcSimulate` ....................................... 3
- `bamp` ............................................... 4
- `checkConvergence` ............................... 6
- `coh` .................................................. 7
- `effects.apc` ....................................... 8
- `plot.apc` ........................................... 8
- `predict_apc` ...................................... 9
- `print.apc` ......................................... 10

---

**Index**

---

<table>
<thead>
<tr>
<th>apc</th>
<th>apc S3 class</th>
</tr>
</thead>
</table>

**Description**

Class for (Bayesian) age-period-cohort objects

**Usage**

`apc()`

**Details**

`bamp` will return an object of class `apc`. Available functions are

- `plot.apc` plots main effects
- `print.apc` print summary of model and effects
- `effects.apc` extract effects (mean, median and quantiles)

**Value**

`apc` class
apc.data  

**Example dataset for APC model**

---

**Description**

A dataset containing case counts and population numbers in eight age groups for ten years. Each age group consists of five years.

**Usage**

```r
data(apc)
```

**Format**

- **population**: matrix of population data
- **cases**: matrix of case counts
- **cov_p**: covariate for period
- **cov_c**: covariate for cohort

---

apcSimulate  

**Simulate from age-period-cohort model**

---

**Description**

This function simulates a data set of cases on the Lexis diagram from given age, period and cohort effects. Population numbers have to be given; can be one number for all age group/period combinations.

**Usage**

```r
apcSimulate(intercept, age, period, cohort, periods_per_agegroup, population)
```

**Arguments**

- **intercept**: Intercept
- **age**: Vector of effect for age groups
- **period**: Vector of effects for periods
- **cohort**: Vector of effect for cohorts
- **periods_per_agegroup**: Periods per age group
- **population**: Population number. Either a matrix or a scalar.
Value

List with number of cases (matrix) and population numbers (matrix).

See Also

vignette("simulation", package = "bamp")

Examples

age=sqrt(seq(5,0,length=10)); age<-1-age-mean(age)
period=15:1; period[8:15]<-8:15; period<-period/6; period<-period-mean(period)
periods_per_agegroup=5; number_of_cohorts <- periods_per_agegroup*(10-1)+15
cohort<-rep(0,60); cohort[1:10]<-10:1; cohort[41:60]<- -(1:20)/2; cohort<-cohort/10;
cohort<-cohort-mean(cohort)
simdata<-apcSimulate(-5, age, period, cohort, periods_per_agegroup, 1e6)
par(mfrow=c(3,1))
plot(age, type="l")
plot(period, type="l")
plot(cohort, type="l")
## Not run:
simmod <- bamp(cases = simdata$cases, population = simdata$population, age = "rw1",
period = "rw1", cohort = "rw1", periods_per_agegroup =periods_per_agegroup)
plot(simmod)
## End(Not run)
• models with additional age, period and/or cohort heterogeneity,
• additional covariates.

Usage

bamp(
  cases,
  population,
  age,
  period,
  cohort,
  overdisp = FALSE,
  period_covariate = NULL,
  cohort_covariate = NULL,
  periods_per_agegroup,
  mcmc.options = list(number_of_iterations = 1e+05, burn_in = 50000, step = 50, tuning = 500),
  hyperpar = list(age = c(1, 0.5), period = c(1, 5e-04), cohort = c(1, 5e-04), overdisp = c(1, 0.05)),
  dic = TRUE,
  parallel = TRUE,
  verbose = FALSE
)

Arguments

cases number of cases
population population number
age prior for age groups ("rw1", "rw2", "rw1+het", "rw2+het", "")
period prior for periods ("rw1", "rw2", "rw1+het", "rw2+het", "")
cohort prior for cohorts ("rw1", "rw2", "rw1+het", "rw2+het", "")
overdisp logical, add overdispersion to model
period_covariate covariate for period
cohort_covariate covariate for cohort
periods_per_agegroup periods per age group
mcmc.options list of options for MCMC.
  • number_of_iterations: number of iterations per chain.
  • burn_in: number of iterations used as burnin at the beginning of the algorithm, these iterations will be removed.
  • step: Step size, for example default is 50, so only every 50th iterations will be stored.
checkConvergence

- tuning: number of iterations for automatic tuning. Depending on the model, the MCMC algorithm will tune certain parameters for more efficient MCMC chains. After tuning, the algorithm is restarted.

hyperpar list of hyper parameters. The hyper prior for the precision (inverse variance) in the random walk priors is a Gamma distribution with parameters $a$ and $b$; expected value is $a/b$, variance is $a/b^2$. Weak hyper parameters are suggested, defaults are $a = 1, b = 0.5$ for age, $a = 1, b = 0.0005$ for period and cohort effects and $a = 1, b = 0.05$ for overdispersion (if added). It is recommended to choose the hyper priors depending on the model, in particular on the order of the random walk.

dic logical. If true. DIC will be computed

parallel logical, should computation be done in parallel. This uses the parallel package, which does not allow parallel computing under Windows.

verbose verbose mode

Details

This functions returns an apc object. Only samples from the posterior are computed, point estimates and credible intervals will be computed in effects.apc, print.apc and plot.apc. predict_apc can be used for for prediction of the future rates and number of cases and for a retrospective prediction for model checking.

See Also

vignette("modeling", package = "bamp")

Examples

```r
## Not run:
data(apc)
model <- bamp(cases, population, age="rw1", period="rw1", cohort="rw1", periods_per_agegroup = 5)
## End(Not run)
```

checkConvergence Check apc object, whether MCMC has converged

Description

This functions uses Gelman and Rubin’s R to check convergence for all main parameters. All parameters should have $R<1.1$. bamp runs at least four MCMC chains by default (more if parallel is more than four).

Usage

```r
checkConvergence(x, info = FALSE, level = 2, auto = FALSE)
```
coh

**Arguments**

- **x**  
  An apc object
- **info**  
  logical; print more information
- **level**  
  level of check; 1 uses point point estimation, 2 uses upper C.I.
- **auto**  
  logical; should be TRUE if called automatically from `bamp` #'

**Value**

logical; TRUE if check is fine.

**Examples**

```r
## Not run:
data(apc)
model <- bamp(cases, population, age="rw1", period="rw1", cohort="rw1", periods_per_agegroup = 5)
checkConvergence(model)
## End(Not run)
```

---

**Description**

Compute cohort index from age and period index

**Usage**

`coh(agegroup, period, noa, periods_per_agegroup)`

**Arguments**

- **agegroup**  
  age group index
- **period**  
  period index
- **noa**  
  number of age groups in total
- **periods_per_agegroup**  
  periods per age group

**Value**

cohort index

**Examples**

```r
# last agegroup in first period equals first cohort
coh(10, 1, 10, 5)

# first agegroup in last period equals last cohort
coh(1, 8, 10, 5)
```
effects.apc

Effects from Fitted APC Model

Description

Effects from Fitted APC Model

Usage

## S3 method for class 'apc'
effects(object, mean = FALSE, quantiles = 0.5, update = FALSE, ...)

Arguments

- **object**: an apc object
- **mean**: logical. If TRUE, mean effects are computed
- **quantiles**: Scalar or vector of quantiles to compute (only if mean=FALSE)
- **update**: logical. If TRUE, the apc object including the effects is returned
- **...**: Additional arguments will be ignored

Value

List of age, period, cohort effects or apc object including effects (if update=TRUE)

Examples

## Not run:
data(apc)
model <- bamp(cases, population, age="rw1", period="rw1", cohort="rw1", periods_per_agegroup = 5)
effects(model)
## End(Not run)

plot.apc

Plot apc object

Description

Plot apc object

Usage

## S3 method for class 'apc'
plot(x, quantiles = c(0.05, 0.5, 0.95), ...)
predict_apc

Arguments

- `x`: apc object
- `quantiles`: quantiles to plot. Default: `c(0.05, 0.5, 0.95)` is median and 90% credible interval.
- `...`: Additional arguments will be ignored

Details

Plot of age, period and cohort effects from apc objects. If covariates have been used for period/cohort, a second plot with covariate, absolute effect and relative effect is created. Absolute effect is relative effect times covariate.

Value

- `plot`

Examples

```r
## Not run:
data(apc)
model <- bamp(cases, population, age="rw1", period="rw1", cohort="rw1", periods_per_agegroup = 5)
plot(model)
## End(Not run)
```

predict_apc  Prediction for age-period-cohort models

Description

Prediction of rates and, if possible, cases from the Bayesian age-period-cohort model using the prior assumptions (random walks) of the model and the estimated variance of the random walk. For example, random walk of first order (rw1) for period effect predicts constant effects for future periods plus noise.

Usage

```r
predict_apc(
    object,
    periods = 0,
    population = NULL,
    quantiles = c(0.05, 0.5, 0.95),
    update = FALSE
)
```
Arguments

- **object**: apc object
- **periods**: number of periods to predict
- **population**: matrix of (predicted) population, if NULL, population data from original bamp call will be used
- **quantiles**: vector of quantiles to compute
- **update**: boolean. If TRUE, object will be returned with results added to the object

Details

This function will return predicted rates for future periods. For this, future period and cohort effects will be predicted. Further age group effects will not be predicted. The rates are random samples from the predictive distribution; number of samples is equal to number of MCMC iterations. Quantiles will be provided for convenience, but all samples are available. If population numbers are given, number of cases will also be predicted. Number of cases will not only be predicted for future periods, but also for the time periods where data are available; this can be used for model assessment.

Value

A list with quantiles of predicted probabilities (pr), predicted cases (cases) and predicted cases per period (cases_period) and a list samples with MCMC samples of pr, cases and cases_period. If `update=TRUE`, the apc object will be returned with this list (predicted) added.

See Also

vignette("prediction", package = "bamp")

Examples

```r
## Not run:
data(apc)
model <- bamp(cases, population, age="rw1", period="rw1", cohort="rw1", periods_per_agegroup = 5)
pred <- predict_apc(model, periods=1)
plot(pred$pr[2,11,], main="Predicted rate per agegroup", ylab="p")

## End(Not run)
```

---

print.apc  
*Print apc objects*

**Description**

Print apc objects
**print.apc**

**Usage**

```r
## S3 method for class 'apc'
print(x, ...)
```

**Arguments**

- `x` apc object
- `...` additional arguments will be ignored

**Value**

print

**Examples**

```r
## Not run:
data(apc)
model <- bamp(cases, population, age="rw1", period="rw1", cohort="rw1", periods_per_agegroup = 5)
print(model)

## End(Not run)
```
Index

* datasets
  apc.data, 3

apc, 2, 6
apc.data, 3
apcSimulate, 3

bamp, 2, 4, 6, 7

cases (apc.data), 3
ccheckConvergence, 6
coh, 7
cov_c (apc.data), 3
cov_p (apc.data), 3

effects.apc, 2, 6, 8

plot.apc, 2, 6, 8
population (apc.data), 3
predict_apc, 6, 9
print (print.apc), 10
print.apc, 2, 6, 10