Package ‘stochasticGEM’

April 19, 2009

Version 0.0-1

Date 2007-02-25

Title R Package for Fitting Stochastic General Epidemic Models

Author Eugene Zwane <e.zwane@gmail.com>.

Maintainer Eugene Zwane <e.zwane@gmail.com>

Depends R (>= 2.4.0), coda

Description stochasticGEM is a publicly available package that implements Bayesian inference for partially observed stochastic epidemics. The general epidemic model is used for estimating the parameters governing the infectious and incubation period length, and the parameters governing susceptibility. In real-life epidemics the infection process is unobserved, and the data consists of the times individuals are detected, usually via appearance of symptoms. The stochasticGEM package fits several variants of the general epidemic model, namely the stochastic SIR with Markovian and nonMarkovian infectious periods. The estimation is based on Markov chain Monte Carlo algorithm.

LazyLoad yes

LazyData yes

License GPL (>= 2)

Repository CRAN

Date/Publication 2007-04-26 17:55:36

R topics documented:

`fLatentSEIR.MH` *Fit a partially observed SEIR general epidemic model with fixed latency duration.*

Description

`fLatentSEIR.MH` is used to fit the SEIR general epidemic model with fixed latency duration via Markov chain Monte Carlo. In this compartmental model only the removal times are observed. The computation is based on the algorithm that is developed by O'Neill & Becker (2001).

Usage

```
fLatentSEIR.MH(N, infectionTimes = NULL, latencyTimes = NULL, removalTimes,
    START = NULL, priorValues = NULL, fixedLatencyDuration = NULL, minInfPeriod = 1
    bayesReps = 10000, burnIn = 0, bayesThin = 1, updateIniInfGibbs=TRUE,
    verbose = FALSE, missingInfectionTimes = TRUE,
    infectious.density = "exponential")
```

Arguments

`N` initial susceptible individuals

`infectionTimes` removal times

`latencyTimes` end of latency times

`removalTimes` removal times

`START` A `vector` with 2 or 3 elements, the infection rate and the removal rate for exponential distributed infectious periods, or the infection rate and the scale and shape parameter of the infectious period respectively for gamma/Weibull distributed infectious periods. Defaults to NULL.

`priorValues` A `list` with elements `infectionRate` and `removalRate` for an exponential density or `infectionRate`, `infectiousScale` and `infectiousShape` for other densities, being vectors of length 2 containing the gamma prior coefficients for the corresponding parameters. A prior, `theta`, for the initial infection time is needed for the `exponential` (a scalar) and `gamma` (a vector of `length` 2). Priors are necessary, for instance when using a gamma density the first element of `theta` should be at least 1.

`fixedLatencyDuration` The 'assumed' fixed value of the latency duration. The default is NULL

`minInfPeriod` The minimum possible infectious period. The default is 1

`bayesReps` A positive integer denoting the number of MCMC draws. The default is 10000

`burnIn` A positive integer denoting the burn-in interval for the Markov chain, i.e., the number of initial draws that should not be stored. The default is 0.

`bayesThin` A positive integer denoting the thinning interval for the Markov chain, i.e., the interval between successive values of the Markov chain. The default is 1.

`updateIniInfGibbs`
: Update the initial infection time. The default is TRUE

`verbose`
: Used to check activity of MCMC sampler. A dot is printed at every `bayesReps`/100 iteration.

`missingInfectionTimes`
: Are missing values updated or fixed. By default the infection times are updated.

`infectious.density`
: type of density. can be either `exponential`, `gamma` and `weibull`

Details

If certain elements of the starting values are missing an attempt is made to get suitable starting values. If 'infection.density' is `exponential` the `remRateSEIR` is returned instead of `infScaleSEIR` and `infShapeSEIR`.

Value

a `list` of components containing the folowing elements:

`logLikelihood`
: A Markov chain Monte Carlo object of the 'psuedo' log likelihood for each completed data set. The function `mcmc` in the `coda` library is used to create the object.

`fixedLatencyDuration`
: The 'assumed' fixed value of the latency duration.

`infectionTimes`
: Posterior mean of the infection times.

`removalTimes`
: Removal times.

`infRateSEIR`
: A Markov chain Monte Carlo object of the Gibbs draws for the infection rate. The function `mcmc` in the `coda` library is used to create the object.

`remRateSEIR`
: A Markov chain Monte Carlo object of the Gibbs draws for the removal rate. The function `mcmc` in the `coda` library is used to create the object.

`infScaleSEIR`
: A Markov chain Monte Carlo object of the Gibbs draws for the scale parameter of the gamma/Weibull distribution. The function `mcmc` in the `coda` library is used to create the object.

`infShapeSEIR`
: A Markov chain Monte Carlo object of the Metropolis-Hastings draws for the shape parameter of the gamma/Weibull distribution. The function `mcmc` in the `coda` library is used to create the object.

`acceptRate`
: Number of accepted draws for the infection times.

`bayesReps`
: The number of MCMC draws

`burnIn`
: The burn-in interval for the Markov chain.

`bayesThin`
: The thinning interval for the Markov chain.

`bayesOut`
: Number of saved iterations.

`infectiousPeriod`
: A Markov chain Monte Carlo object of the infectious period. The function `mcmc` in the `coda` library is used to create the object.

`reproductionNumber`
: A Markov chain Monte Carlo object of the reproduction number. The function `mcmc` in the `coda` library is used to create the object.

`initialSusceptible`
: Initial susceptible individuals.

`initialInfective`
: Initial infective individuals.

Author(s)

Eugene Zwane ⟨e.zwane@gmail.com⟩

References

O'Neill, P.D. & Roberts, G.O. (1999). 'Bayesian inference for partially observed stochastic epidemics' *J.R. Statist. Soc. A.* **162**, 121-129.

O'Neill, P.D. & Becker, N.G. (2001). 'Inference for an epidemic when susceptibility varies' *Biostatistics* **2**, 99-108.

O'Neill, P.D. (2002). 'A tutorial introduction to Bayesian inference for stochastic epidemic models using Markov chain Monte Carlo methods' *Mathematical Biosciences* **180**, 103-114.

See Also

`getValidSEIR`, `SEIR.MH`, `SIR.MH`, `SIR.MLE`

Examples

```
data(smallpox)
priors <- list(infectionRate = c(1.0,0.01), removalRate = c(1.0,0.01), theta = c(0,0))
temp <- fLatentSEIR.MH(N=119,
    removalTimes=smallpox, priorValues=priors,
    fixedLatencyDuration = 10, bayesReps=1000,
    burnIn=500,bayesThin=1)
summary(temp$infRateSEIR)
summary(temp$remRateSEIR)
summary(temp$infectiousPeriod)
summary(temp$reproductionNumber)
#
#
priors <- list(infectionRate = c(1.00,0.01),
    infectiousScale = c(1.0,0.01), infectiousShape = c(1.0,0.01),
    theta = c(1.0,0.01))
temp <- fLatentSEIR.MH(N=119,
    removalTimes=smallpox, priorValues=priors,
    fixedLatencyDuration = 10, bayesReps=1000,
    burnIn=500,bayesThin=1,infectious.density = "gamma")
summary(temp$infRateSEIR)
summary(temp$infScaleSEIR)
summary(temp$infShapeSEIR)
summary(temp$infectiousPeriod)
summary(temp$reproductionNumber)
```

```
#
#
priors <- list(infectionRate = c(1.00,0.01),
    infectiousScale = c(1.0,0.01), infectiousShape = c(1.0,0.01))
temp <- fLatentSEIR.MH(N=119,
    removalTimes=smallpox, priorValues=priors,
    fixedLatencyDuration = 10, bayesReps=1000,
    burnIn=500,bayesThin=1,infectious.density = "weibull")
summary(temp$infRateSEIR)
summary(temp$infScaleSEIR)
summary(temp$infShapeSEIR)
summary(temp$infectiousPeriod)
summary(temp$reproductionNumber)
```

`getValidSEIR` *Fit a fully observed standard SIR general epidemic model*

Description

`getValidSEIR` is used to fit the standard SIR general epidemic model. This is an internal function which can also be called directly by the user.

Usage

```
getValidSEIR(N,removalTimes,fLatent=NULL)
```

Arguments

`N`	initial susceptible individuals
`removalTimes`	removal times
`fLatent`	some assumed value of the latency duration

Details

This is a simple and likely to be in-efficient way of getting valid SEIR starting infection and 'end of latency' times. It uses the maximum interremoval time and the assumed latency duration.

Author(s)

Eugene Zwane ⟨e.zwane@gmail.com⟩

See Also

`SEIR.MH`, `fLatentSEIR.MH`

Examples

```
data(smallpox)
getValidSEIR(119, smallpox, 10)
```

respiratory — *Removal times by age for a respiratory epidemic that occurred on the island of Tristan da Cunha, 1967*

Description

Removal times and removals by 'age'.

Usage

```
data(respiratory)
```

Format

A `list` containing 2 elements, a `vector` with removal data (respiratoryRemovals) and a matrix with removals by 'age' and day of removal (removals.by.day).

References

Becker, N.G. & Hopper, J.L. (1983). 'Assessing heterogeneity of disease spread through a community' *American J. Epidemiology* **117**, 362-374.

SEIR.MH — *Fit a partially observed SEIR general epidemic model.*

Description

`SEIR.MH` is used to fit the SEIR general epidemic model via Markov chain Monte Carlo. In this compartmental model only the removal times are observed. The computation is based on the algorithm that is developed by O'Neill & Becker (2001).

Usage

```
SEIR.MH(N, infectionTimes = NULL, InfPeriod = c(1,20),
    latencyTimes = NULL, LatPeriod = c(1,20), removalTimes,  START = NULL,
    priorValues = NULL, bayesReps = 10000, burnIn = 0, bayesThin = 1,
    verbose = FALSE, missingInfectionTimes = TRUE)
```

Arguments

`N`	initial susceptible individuals
`infectionTimes`	removal times
`InfPeriod`	A range of giving the possible infectious periods
`latencyTimes`	end of latency times

`LatPeriod` A range of giving the possible latency durations

`removalTimes` removal times

`START` A `vector` with 2 elements, the infection and removal rate respectively. Defaults to NULL.

`priorValues` A `list` with elements`infectionRate`, `removalRate` and `theta`, the first 2 being vectors of length 2 containing the gamma prior coefficients for the corresponding parameters, and `theta` being a scalar for the exponential prior of the infection time of the initial infective. Defaults to NULL.

`bayesReps` A positive integer denoting the number of MCMC draws. The default is 10000

`burnIn` A positive integer denoting the burn-in interval for the Markov chain, i.e., the number of initial draws that should not be stored. The default is 0.

`bayesThin` A positive integer denoting the thinning interval for the Markov chain, i.e., the interval between successive values of the Markov chain. The default is 1.

`verbose` Used to check activity of MCMC sampler. A dot is printed at every `bayesReps`/100 iteration.

`missingInfectionTimes`
Are missing values updated or fixed. By default the infection times are updated.

Details

If certain elements of the starting values are missing an attempt is made to get suitable starting values.

Value

a `list` of components containing the folowing elements:

`logLikelihood`
A Markov chain Monte Carlo object of the 'psuedo' log likelihood for each completed data set. The function `mcmc` in the `coda` library is used to create the object.

`infectionTimes`
Posterior mean of the infection times.

`removalTimes` Removal times.

`infRateSEIR` A Markov chain Monte Carlo object of the Gibbs draws for the infection rate. The function `mcmc` in the `coda` library is used to create the object.

`latRateSEIR` A Markov chain Monte Carlo object of the Gibbs draws for the latency rate. The function `mcmc` in the `coda` library is used to create the object.

`remRateSEIR` A Markov chain Monte Carlo object of the Gibbs draws for the removal rate. The function `mcmc` in the `coda` library is used to create the object.

`acceptRate` Number of accepted draws for the infection times.

`bayesReps` The number of MCMC draws

`burnIn` The burn-in interval for the Markov chain.

`bayesThin` The thinning interval for the Markov chain.

`bayesOut` Number of saved iterations.

`infectiousPeriod`
: A Markov chain Monte Carlo object of the infectious period. The function `mcmc` in the `coda` library is used to create the object.

`reproductionNumber`
: A Markov chain Monte Carlo object of the reproduction number. The function `mcmc` in the `coda` library is used to create the object.

`initialSusceptible`
: Initial susceptible individuals.

`initialInfective`
: Initial infective individuals.

Author(s)

Eugene Zwane ⟨e.zwane@gmail.com⟩

References

Gibson, G.J. & Renshaw, E. (1998). 'Estimating the parameters in stochastic epidemic models using Markov chain models' *IMA Journal of Mathematics Applied in Medicine & Biology* **16**, 19-40.

Hoehle, M., Jorgensen, E. & O'Neill, P.D. (2005). 'Inference in disease transmission experiments by using stochastic epidemic models' *Appl. Statist.* **54**, 349-366.

O'Neill, P.D. & Roberts, G.O. (1999). 'Bayesian inference for partially observed stochastic epidemics' *J.R. Statist. Soc. A.* **162**, 121-129.

O'Neill, P.D. & Becker, N.G. (2001). 'Inference for an epidemic when susceptibility varies' *Biostatistics* **2**, 99-108.

O'Neill, P.D. (2002). 'A tutorial introduction to Bayesian inference for stochastic epidemic models using Markov chain Monte Carlo methods' *Mathematical Biosciences* **180**, 103-114.

See Also

`fLatentSEIR.MH`, `getValidSEIR`, `SIR.MH`, `SIR.MLE`

Examples

```
data(smallpox)
priors <- list(infectionRate = c(1.00,0.01),
        latencyRate = c(1.00,0.01),
        removalRate = c(1.00,0.01),
        theta = c(0.01,0.01))
validStartTimes <- getValidSEIR(119,smallpox,13)
temp <- SEIR.MH(N=119,
        infectionTimes=validStartTimes$infectionTimes,
        latencyTimes=validStartTimes$latencyTimes,
        removalTimes=smallpox, priorValues=priors,
        bayesReps=1000,burnIn=500,bayesThin=1)
summary(temp$infRateSEIR)
summary(temp$remRateSEIR)
summary(temp$infectiousPeriod)
```

```
summary(temp$reproductionNumber)
```

SIR.MH *Fit a partially observed Markovian/nonMarkovian SIR general epidemic model*

Description

SIR.MH is used to fit the Markovian/nonMarkovian SIR general epidemic model via Markov chain Monte Carlo. In this compartmental model only the removal times are observed.

Usage

```
SIR.MH(N, infectionTimes = NULL, removalTimes, START = NULL,
    priorValues = NULL, bayesReps = 10000, burnIn = 0, bayesThin = 1,
    ARS = TRUE, verbose = FALSE, missingInfectionTimes = TRUE,
    infectious.density = "exponential")
```

Arguments

N	initial susceptible individuals
infectionTimes	starting values for the infection times
removalTimes	removal times
START	A vector with 2 or 3 elements, the infection rate and the removal rate for exponential distributed infectious periods, or the infection rate and the scale and shape parameter of the infectious period respectively for gamma/Weibull distributed infectious periods. Defaults to NULL.
priorValues	A list with elements infectionRate and removalRate for an exponential density or infectionRate, infectiousScale and infectiousShape for other densities, being vectors of length 2 containing the gamma prior coefficients for the corresponding parameters. A prior, theta, for the initial infection time is needed for the exponential (a scalar) and gamma (a vector of length 2). Priors are necessary, for instance when using a gamma density the first element of theta should be at least 1.
bayesReps	A positive integer denoting the number of MCMC draws. The default is 10000
burnIn	A positive integer denoting the burn-in interval for the Markov chain, i.e., the number of initial draws that should not be stored. The default is 0.
bayesThin	A positive integer denoting the thinning interval for the Markov chain, i.e., the interval between successive values of the Markov chain. The default is 1.
ARS	Use adaptive rejection sampling or slice sampling to update shape parameter(s). Defaults to adaptive rejection sampling.
verbose	Used to check activity of MCMC sampler. A dot is printed at every bayesReps/100 iteration.

missingInfectionTimes
: Are missing values updated or fixed. By default the infection times are updated.

infectious.density
: type of density. can be either exponential, gamma and weibull

Details

If certain elements of the starting values are missing an attempt is made to get suitable starting values. In this attempt the infectiousShape is assumed to be 1. If 'infection.density' is exponential the remRateSIR is returned instead of infScaleSIR and infShapeSIR.

Value

a list of components containing the folowing elements:

logLikelihood
: A Markov chain Monte Carlo object of the 'psuedo' log likelihood for each completed data set. The function mcmc in the coda library is used to create the object.

infectionTimes
: Posterior mean of the infection times.

removalTimes
: Removal times.

infRateSIR
: A Markov chain Monte Carlo object of the Gibbs draws for the infection rate. The function mcmc in the coda library is used to create the object.

remRateSIR
: A Markov chain Monte Carlo object of the Gibbs draws for the removal rate for exponentially distributed infectious periods. The function mcmc in the coda library is used to create the object.

infScaleSIR
: A Markov chain Monte Carlo object of the Gibbs draws for the scale parameter of the gamma/Weibull distribution. The function mcmc in the coda library is used to create the object.

infShapeSIR
: A Markov chain Monte Carlo object of the Metropolis-Hastings draws for the shape parameter of the gamma/Weibull distribution. The function mcmc in the coda library is used to create the object.

acceptRate
: A vector of length 2 with the number of accepted draws for the infection times and the shape parameter of the infectious duration. The first elements corresponds to the infection times and the second to the shape parameter.

bayesReps
: The number of MCMC draws

burnIn
: The burn-in interval for the Markov chain.

bayesThin
: The thinning interval for the Markov chain.

bayesOut
: Number of saved iterations.

initialSusceptible
: Initial susceptible individuals.

initialInfective
: Initial infective individuals.

infectiousPeriod
: A Markov chain Monte Carlo object of the infectious period. The function mcmc in the coda library is used to create the object.

`reproductionNumber`
A Markov chain Monte Carlo object of the reproduction number. The function `mcmc` in the `coda` library is used to create the object.

Author(s)

Eugene Zwane ⟨e.zwane@gmail.com⟩

References

O'Neill, P.D. & Roberts, G.O. (1999). 'Bayesian inference for partially observed stochastic epidemics' *J.R. Statist. Soc. A.* **162**, 121-129.

O'Neill, P.D. & Becker, N.G. (2001). 'Inference for an epidemic when susceptibility varies' *Biostatistics* **2**, 99-108.

Streftaris, G. & Gibson, G.J. (2004). 'Bayesian inference for stochastic epidemics in closed populations' *Statistical Modelling* **4**, 63-75.

See Also

`fLatentSEIR.MH`, `SEIR.MH`, `SIR.MLE`

Examples

```
data(smallpox)
priors <- list(infectionRate = c(0.0,0.0), removalRate = c(0.0,0.0), theta = 0)
temp <- SIR.MH(N=119,removalTimes=smallpox, priorValues=priors,
    bayesReps=1000,burnIn=500,bayesThin=1,infectious.density = "exponential")
summary(temp$infRateSIR)
summary(temp$remRateSIR)
summary(temp$infectiousPeriod)
summary(temp$reproductionNumber)
#
#
priors <- list(infectionRate = c(0.0,0.0),
    infectiousScale = c(0.0,0.0), infectiousShape = c(0.0,0.0),
    theta = c(1,0))
temp <- SIR.MH(N=119,removalTimes=smallpox, priorValues=priors,
    bayesReps=1000,burnIn=500,bayesThin=1,infectious.density = "gamma")
summary(temp$infRateSIR)
summary(temp$infScaleSIR)
summary(temp$infShapeSIR)
summary(temp$infectiousPeriod)
summary(temp$reproductionNumber)
#
#
priors <- list(infectionRate = c(0.0,0.0),
    infectiousScale = c(0.0,0.0), infectiousShape = c(0.0,0.0))
temp <- SIR.MH(N=119,removalTimes=smallpox, priorValues=priors,
    bayesReps=1000,burnIn=500,bayesThin=1,infectious.density = "weibull")
summary(temp$infRateSIR)
summary(temp$infScaleSIR)
```

```
summary(temp$infShapeSIR)
summary(temp$infectiousPeriod)
summary(temp$reproductionNumber)
```

SIR.MLE *Fit a fully observed standard SIR general epidemic model*

Description

SIR.MLE is used to fit the standard SIR general epidemic model. This is an internal function which can also be called directly by the user. It requires that the epidemic is fully observed, implying that the infection times are also available.

Usage

```
SIR.MLE(parameters, N, infectionTimes, removalTimes,
    infectious.density = "exponential")
```

Arguments

parameters	starting values for parameters. a vector of length 2. needed by optim.
N	initial susceptible individuals
infectionTimes	infection times
removalTimes	removal times
infectious.density	type of density. can be either exponential, gamma and weibull

Value

a list of components returned by optim.

Author(s)

Eugene Zwane ⟨e.zwane@gmail.com⟩

References

O'Neill, P.D. & Roberts, G.O. (1999). 'Bayesian inference for partially observed stochastic epidemics' *J.R. Statist. Soc. A*. **162**, 121-129.

O'Neill, P.D. & Becker, N.G. (2001). 'Inference for an epidemic when susceptibility varies' *Biostatistics* **2**, 99-108.

Streftaris, G. & Gibson, G.J. (2004). 'Bayesian inference for stochastic epidemics in closed populations' *Statistical Modelling* **4**, 63-75.

See Also

SIR.MH

Examples

```
data(smallpox)
SIR.MLE(c(1/1000,1/10), 119, smallpox-16, smallpox,
    infectious.density = "exponential")
```

---

smallpox — *Removal times for smallpox data*

Description

Removal times associated with an outbreak of smallpox in a closed community of 120 individuals in Abakaliki, Nigeria.

Usage

```
data(smallpox)
```

Format

A `vector` containing 30 observations

References

Bailey, N.T.J. (1975). *The Mathematical Theory of Infectious Diseases and its Applications*, 2nd edn. London: Griffin

Index