Package: timedelay (via r-universe)

October 15, 2024

Version 1.0.11

Date 2020-05-18

Title Time Delay Estimation for Stochastic Time Series of Gravitationally Lensed Quasars

Author Hyungsuk Tak, Kaisey Mandel, David A. van Dyk, Vinay L. Kashyap, Xiao-Li Meng, and Zhirui Hu

Maintainer Hyungsuk Tak <hyungsuk.tak@gmail.com>

Depends R (>= 3.5.0)

Imports MASS (>= 7.3-51.3), mvtnorm(>= 1.0-11)

Description We provide a toolbox to estimate the time delay between the brightness time series of gravitationally lensed quasar images via Bayesian and profile likelihood approaches. The model is based on a state-space representation for irregularly observed time series data generated from a latent continuous-time Ornstein-Uhlenbeck process. Our Bayesian method adopts scientifically motivated hyper-prior distributions and a Metropolis-Hastings within Gibbs sampler, producing posterior samples of the model parameters that include the time delay. A profile likelihood of the time delay is a simple approximation to the marginal posterior distribution of the time delay. Both Bayesian and profile likelihood approaches complement each other, producing almost identical results; the Bayesian way is more principled but the profile likelihood is easier to implement. A new functionality is added in version 1.0.9 for estimating the time delay between doubly-lensed light curves observed in two bands. See also Tak et al. (2017) <doi:10.1214/17-AOAS1027>, Tak et al. (2018) <doi:10.1080/10618600.2017.1415911>, Hu and Tak (2020) <arXiv:2005.08049>.

License GPL-2

Encoding UTF-8

NeedsCompilation no

Date/Publication 2020-05-19 11:50:02 UTC

Contents

---

bayesian *Estimating the time delay via the Bayesian method*

Description

bayesian produces posterior samples of all the model parameters including the time delay. This function has options for three Markov chain Monte Carlo (MCMC) techniques; (i) ancillarity-sufficiency interweaving strategy (ASIS) and (ii) adaptive MCMC to improve the convergence rate, and (iii) repelling-attracting Metropolis for multimodality.

Usage

```
bayesian(data.lcA, data.lcB, data.flux, theta.ini,
  delta.ini, delta.uniform.range, delta.proposal.scale,
  tau.proposal.scale, tau.prior.shape = 1, tau.prior.scale = 1,
  sigma.prior.shape = 1, sigma.prior.scale = 1e-7,
  asis = TRUE, micro, multimodality = FALSE,
  adaptive.frequency, adaptive.delta =  TRUE, adaptive.delta.factor,
  adaptive.tau = TRUE, adaptive.tau.factor,
  sample.size, warmingup.size)
```

Arguments

data.lcA	A (n_1 by 3) matrix for image A (light curve A); the first column has n_1 observation times, the second column contains n_1 flux (or magnitude) values, the third column includes n_1 measurement errors.
data.lcB	A (n_2 by 3) matrix for image B (light curve B); the first column has n_2 observation times, the second column contains n_2 flux (or magnitude) values, the third column includes n_2 measurement errors.

data.flux	"True" if data are recorded on flux scale or "FALSE" if data are on magnitude scale. Simply speaking, if your observed time series can take on negative values, then data.flux = FALSE.
theta.ini	Initial values of the OU parameters, (μ, σ, τ).
delta.ini	Initial value of the time delay, Δ.
delta.uniform.range	
	The range of the Uniform prior distribution for the time delay.
delta.proposal.scale	
	The proposal scale of the Metropolis step for the time delay.
tau.proposal.scale	
	The proposal scale of the Metropolis-Hastings step for τ.
tau.prior.shape	
	The shape parameter of the Inverse-Gamma hyper-prior distribution for τ. Default is 1.
tau.prior.scale	
	The scale parameter of the Inverse-Gamma hyper-prior distribution for τ. Default is 1.
sigma.prior.shape	
	The shape parameter of the Inverse-Gamma hyper-prior distribution for σ^2. Default is 1.
sigma.prior.scale	
	The scale parameter of the Inverse-Gamma hyper-prior distribution for σ^2. Default is 1e-7.
asis	(Optional) "TRUE" if we use the ancillarity-sufficiency interweaving strategy (ASIS) for c (always recommended). Default is "TRUE".
micro	It determines the order of a polynomial regression model that accounts for the difference between microlensing trends. Default is 3. When zero is assigned, the Bayesian model fits a curve-shifted model.
multimodality	(Optional) "TRUE" if we use a repelling-attracting Metropolis algorithm for sampling Delta. When it is "TRUE" it is recommended that adaptive.delta = FALSE so that the adaptive MCMC is not used in the presence of multimodality. Default is "FALSE".
adaptive.frequency	
	(If "adaptive.delta = TRUE" or "adaptive.tau = TRUE") The adaptive MCMC is applied for every specified frequency. If it is specified as 500, the adaptive MCMC is applied to every 500th iterstion.
adaptive.delta	(Optional) "TRUE" if we use the adaptive MCMC for the time delay. Default is "TRUE".
adaptive.delta.factor	
	(If "adaptive.delta = TRUE") The factor, $\exp(\pm\text{adaptive.delta.factor})$, multiplied by the proposal scale of the time delay for adaptive MCMC.
adaptive.tau	(Optional) "TRUE" if we use the adaptive MCMC for tau. Default is "TRUE".
adaptive.tau.factor	
	(If "adaptive.tau = TRUE") The factor, $\exp(\pm\text{adaptive.tau.factor})$, multiplied by the proposal scale of tau for adaptive MCMC.
sample.size	The number of the posterior samples of each model parameter.
warmingup.size	The number of burn-in posterior samples.

Details

The function `bayesian` produces posterior samples of the model parameters one of which is the time delay. Please note that when astronomical time series data are loaded on R by `read.table`, `read.csv`, etc., some decimal places of the the observation times are automatically rounded because R's default is to load seven effective digits. For example, R will load the observation time 51075.412789 as 51075.41. This default will produce many ties in observation times even though there is actually no tie in observation times. To prevent this, please type `"options(digits = 11)"` before loading the data if the observation times are in seven effective digits.

Value

The outcome of `bayesian` comprises of:

`delta`	Posterior samples of the time delay
`X`	Posterior samples of the latent magnitudes
`beta`	Posterior samples of the polynomial regression coefficients, beta
`mu`	Posterior samples of the mean parameter of the O-U process, mu
`sigma`	Posterior samples of the short term variability of the O-U process, sigma
`tau`	Posterior samples of the mean reversion time of the O-U process, tau
`tau.accept.rate`	
	The acceptance rate of the MCMC for tau
`delta.accept.rate`	
	The acceptance rate of the MCMC for the time delay

Author(s)

Hyungsuk Tak

References

Hyungsuk Tak, Kaisey Mandel, David A. van Dyk, Vinay L. Kashyap, Xiao-Li Meng, and Aneta Siemiginowska (2017). "Bayesian Estimates of Astronomical Time Delays between Gravitationally Lensed Stochastic Light Curves," The Annals of Applied Statistics, 11 (3), 1309-1348. Hyungsuk Tak, Xiao-Li Meng, and David A. van Dyk (2018), "A Repelling-Attracting Metropolis Algorithm for Multimodality", Journal of Computational and Graphical Statistics, 27 (3), 479-490. Zhirui Hu and Hyungsuk Tak (2020+), "Modeling Stochastic Variability in Multi-Band Time Series Data," arXiv:2005.08049.

Examples

```
# Loading datasets
data(simple)

# A typical quasar data set
head(simple)

# Subset (data for image A) of the typical quasar data set
lcA <- simple[, 1 : 3]
```

```
# Another subset (data for image B) of the typical quasar data set
# The observation times for image B are not necessarily the same as those for image A
lcB <- simple[, c(1, 4, 5)]

# The two subsets do not need to have the same number of observations
# For example, here we add one more observation time for image B
lcB <- rbind(lcB, c(290, 1.86, 0.006))

dim(lcA)
dim(lcB)

###############################################
# Time delay estimation via Bayesian approach #
###############################################

# Cubic microlensing model (m = 3)

output = bayesian(data.lcA = lcA, data.lcB = lcB,
                  data.flux = FALSE, theta.ini = c(0, 0.01, 200),
                  delta.ini = 50, delta.uniform.range = c(0, 100),
                  delta.proposal.scale = 1,
                  tau.proposal.scale = 3,
                  tau.prior.shape = 1, tau.prior.scale = 1,
                  sigma.prior.shape = 1, sigma.prior.scale = 2 / 10^7,
                  asis = TRUE, micro = 3,
                  sample.size = 10, warmingup.size = 10)
names(output)

# hist(output$delta, 20)
# plot(output$delta, type = "l")
# acf(output$delta)
# output$delta.accept
# output$tau.accept

# If multimodality exists, then please add two arguments:
# multimodality = TRUE, adaptive.delta = FALSE
# This is to prevent the Markov chain from adapting to a local mode.
# If we know the distance between the most distant modes,
# try the smallest value of delta.proposal.scale that still makes
# the Markov chain jump between those modes frequently.
# For example, when the distance is d, then try
# delta.proposal.scale = d * 0.7, decreasing or increasing 0.7.

# Graphical model checking
# beta.hat <- colMeans(output$beta)
# delta.hat <- mean(output$delta)
# time.x <- lcB[, 1] - delta.hat
# time.covariate <- cbind(rep(1, length(time.x)), time.x, time.x^2, time.x^3)
##### This time.covariate is when micro = 3, a third-order polynomial regression.
##### With micro = 0, "time.covariate <- rep(1, length(time.x))" is used.
# predicted <- time.covariate %*% beta.hat
# plot(lcA[, 1], lcA[, 2], col = 2)
```

```
##### Adjust the range of the x-axis by the argument of plot.
##### For example "xlim = c(-1, 2)"
# points(lcB[, 1] - delta.hat, lcB[, 2] - predicted, col = 4, pch = 1)
# for (i in 1 : length(output$delta)) {
#   temp.time <- c(lcA[, 1], lcB[, 1] - output$delta[i])
#   points(sort(temp.time), output$X[i, ],
#          col = "gray", cex = 0.5, pch = 1, lwd = 0.01)
# }
# points(lcA[, 1], lcA[, 2], pch = 0, col = 2, lwd = 1)
# points(lcB[, 1] - delta.hat, lcB[, 2] - predicted,
#        col = 4, pch = 1, lwd = 1)
```

---

bayesian.multiband   *Estimating the time delay between doubly-lensed multi-band light curves in a Bayesian way*

Description

bayesian.multiband produces posterior samples of all the model parameters including the time delay between doubly-lensed multi-band light curves.

Usage

```
bayesian.multiband(data.band1, data.band2, n.bands = 2,
  theta.ini = c(0.01, 0.01, 100, 100, 0.5),
  delta.ini, delta.uniform.range = c(-500, 500), delta.proposal.scale = 1,
  tau.proposal.scale = 1, tau.prior.shape = 1, tau.prior.scale = 1,
  sigma2.proposal.scale = 0.5, sigma2.prior.shape = 1, sigma2.prior.scale = 1e-7,
  rho.proposal.scale = 0.1, beta.prior.diag = 10 * c(0.1, 0.01, 1e-3, 1e-5)^2,
  micro = 3, timesc = 100, adaptive.frequency = 100,
  adaptive.delta.factor = 0.1, adaptive.tau.factor = 0.1,
  adaptive.sigma2.factor = 0.1, adaptive.rho.factor = 0.1,
  sample.size, warmingup.size)
```

Arguments

data.band1   A (n_1 by 5) matrix composed of the data for doubly-lensed images oberved in band 1; the first column contains n_1 observation times, the second column contains n_1 magnitudes for image A, the third column has n_1 measurement error standard deviations for image A, the fourth column include n_1 magnitudes for image B, and the fifth column contains n_1 measurement error standard deviations for image B.

data.band2   A (n_2 by 5) matrix composed of the data for doubly-lensed images oberved in band 2; the first column contains n_2 observation times, the second column contains n_2 magnitudes for image A, the third column has n_2 measurement error standard deviations for image A, the fourth column include n_2 magnitudes for

image B, and the fifth column contains n_2 measurement error standard deviations for image B.

n.bands   The number of bands used to obtain the data. For now, only two bands are allowed. The authors plan to modify the code to allow more bands in the near future.

theta.ini   A vector for initial values of the OU processes for two bands, i.e., $(\sigma_1, \sigma_2, \tau_1, \tau_2, \rho)$, where σ_1 is the short-term variability (standard deviation) for band 1, σ_2 is the short-term variability (standard deviation) for band 2, τ_1 is the timescale for band 1, and τ_2 is the timescale for band 2, and ρ is the cross-correlation parameter between two bands. Defaults are (0.01, 0.01, 100, 100, 0.5).

delta.ini   Initial values of the time delay.

delta.uniform.range
                 The range of the Uniform prior distribution for the time delay. Default range is set to $(-500, 500)$.

delta.proposal.scale
                 The proposal scale of the Metropolis step for the time delay. Default is 1.

tau.proposal.scale
                 The proposal scale of the Metropolis-Hastings step for τ_j $(j = 1, 2)$ Default is 1.

tau.prior.shape
                 The shape parameter of the Inverse-Gamma hyper-prior distribution for τ_j. Default is 1.

tau.prior.scale
                 The scale parameter of the Inverse-Gamma hyper-prior distribution for τ_j. Default is 1.

sigma2.proposal.scale
                 The proposal scale of the Metropolis-Hastings step for σ_j^2 $(j = 1, 2)$. Default is 0.5.

sigma2.prior.shape
                 The shape parameter of the Inverse-Gamma hyper-prior distribution for σ_j^2. Default is 1.

sigma2.prior.scale
                 The scale parameter of the Inverse-Gamma hyper-prior distribution for σ_j^2. If no prior information is available, we recommend using 1e-7 (default).

rho.proposal.scale
                 The proposal scale of the Metropolis-Hastings step for ρ. Default is 0.1.

beta.prior.diag
                 The diagonal elements of the covariance matrix in the multivariate Gaussian prior for β (polynomial regression coefficients for microlensing adjustment). If such information is not available, these are set to `10 * c(0.1)^2` if `micro = 0`, `10 * c(0.1, 0.01)^2` if `micro = 1`, `10 * c(0.1, 0.01, 1e-3)^2` if `micro = 2`, and `10 * c(0.1, 0.01, 1e-3, 1e-5)^2` if `micro = 3`.

micro   A non-negative integer less than or equal to 3. It determines the order of a polynomial regression model that accounts for the long-term trend of microlensing effect. Default is 3.

timesc   It scales the observation time for fitting polynomial of microlensing, i.e., time /
                    timesc, so that the coefficients are not too small. Default is 100.

adaptive.frequency

                    The adaptive MCMC is applied for every specified frequency. If it is specified
                    as 500, the adaptive MCMC is applied to every 500th iterstion. Default is 100.

adaptive.delta.factor

                    The factor, exp($\pm$adaptive.delta.factor), multiplied by the proposal scale of the
                    time delay for adaptive MCMC. Default is 0.1.

adaptive.tau.factor

                    The factor, exp($\pm$adaptive.tau.factor), multiplied by the proposal scale of τ_j for
                    adaptive MCMC. Default is 0.1.

adaptive.sigma2.factor

                    The factor, exp($\pm$adaptive.tau.factor), multiplied by the proposal scale of σ_j^2 for
                    adaptive MCMC. Default is 0.1.

adaptive.rho.factor

                    The factor, exp($\pm$adaptive.tau.factor), multiplied by the proposal scale of ρ for
                    adaptive MCMC. Default is 0.1.

sample.size   The number of the posterior samples of each model parameter.

warmingup.size   The number of burn-in samples for MCMC.

Details

The function `bayesian.multiband` produces posterior samples of the model parameters, where the
time delay (delta) is of primary interest. For now, this function only supports doubly-lensed data
observed in two bands. The authors plan to generalize this code to account for more than two bands
and more than two lens.

Please note that when astronomical time series data are loaded on R by `read.table`, `read.csv`, etc.,
some decimal places of the the observation times are automatically rounded because R's default is
to load seven effective digits. For example, R will load the observation time 51075.412789 as
51075.41. This default will produce many ties in observation times even though there is actually no
tie in observation times. To prevent this, please type `"options(digits = 11)"` before loading the
data if the observation times are in seven effective digits.

Value

The outcomes of `bayesian.multiband` are composed of:

delta   A vector for m posterior samples of the time delay.

beta   An m by $k+1$ matrix containing posterior samples of the polynomial regression
                    coefficients, where m is the size of the posterior sample, and k is the polynomial
                    order for microlensing.

rho   A vector for m posterior samples of the cross-correlation parameter.

sigma   An m by 2 matrix containing posterior samples of the short-term variability
                    (standard deviation) of the O-U process. The first column is composed ot the
                    m posterior samples of σ_1 in band 1, and the second column contains the m
                    posterior samples of σ_2 in band 2.

tau
An m by 2 matrix containing posterior samples of the timescale of the O-U process. The first column is composed ot the m posterior samples of τ_1 in band 1, and the second column contains the m posterior samples of τ_2 in band 2.

tau.accept.rate
The acceptance rate of the MCMC for τ_1 and τ_2.

sigma.accept.rate
The acceptance rate of the MCMC for σ_1 and σ_2.

delta.accept.rate
The acceptance rate of the MCMC for the time delay

rho.accept.rate
The acceptance rate of the MCMC for ρ.

Author(s)

Zhirui Hu and Hyungsuk Tak

References

Hyungsuk Tak, Kaisey Mandel, David A. van Dyk, Vinay L. Kashyap, Xiao-Li Meng, and Aneta Siemiginowska (2017). "Bayesian Estimates of Astronomical Time Delays between Gravitationally Lensed Stochastic Light Curves," The Annals of Applied Statistics, 11 (3), 1309-1348. Hyungsuk Tak, Xiao-Li Meng, and David A. van Dyk (2018), "A Repelling-Attracting Metropolis Algorithm for Multimodality", Journal of Computational and Graphical Statistics, 27 (3), 479-490. Zhirui Hu and Hyungsuk Tak (2020+), "Modeling Stochastic Variability in Multi-Band Time Series Data," arXiv:2005.08049.

Examples

```
# Loading datasets
data(simple.band1)
data(simple.band2)

# Doubly-lensed quasar data set observed in two bands
# Each data set contains doubly-lensed light curves observed in one band.
head(simple.band1)
head(simple.band2)

# The length of each data set (i.e., number of observation times)
# do not need to be the same.

dim(simple.band1)
dim(simple.band2)

################################################
# Time delay estimation via Bayesian approach #
################################################

# Cubic microlensing model (m = 3)

output <- bayesian.multiband(data.band1 = simple.band1,
```

```
              data.band2 = simple.band2, n.bands = 2,
              theta.ini = c(0.01, 0.01, 100, 100, 0.5),
              delta.ini = 100, delta.uniform.range = c(-500, 500),
              tau.proposal.scale = 1, tau.prior.shape = 1, tau.prior.scale = 1,
         sigma2.proposal.scale = 0.5, sigma2.prior.shape = 1, sigma2.prior.scale = 1e-7,
           rho.proposal.scale = 0.1, beta.prior.diag = 10 * c(0.1, 0.01, 1e-3, 1e-5)^2,
              micro = 3, timesc = 100, adaptive.frequency = 100,
              adaptive.delta.factor = 0.1, adaptive.tau.factor = 0.1,
              adaptive.sigma2.factor = 0.1, adaptive.rho.factor = 0.1,
              sample.size = 100, warmingup.size = 100)
names(output)


# hist(output$delta, 20)
# plot(output$delta, type = "l")
# acf(output$delta)
# output$delta.accept.rate
```

---

entirelogprofilelikelihood

Calculating the entire profilel likelihood curve over the given grid values of the time delay

---

Description

entirelogprofilelikelihood calculates the entire profilel likelihood curve over the given grid values of the time delay.

Usage

```
entirelogprofilelikelihood(data.lcA, data.lcB, grid,
                                initial, data.flux,
                                delta.uniform.range, micro)
```

Arguments

data.lcA   A (n_1 by 3) matrix for image A (light curve A); the first column has n_1 observation times, the second column contains n_1 flux (or magnitude) values, the third column includes n_1 measurement errors.

data.lcB   A (n_2 by 3) matrix for image B (light curve B); the first column has n_2 observation times, the second column contains n_2 flux (or magnitude) values, the third column includes n_2 measurement errors.

grid   A vector containing values of the time delay on which the profile likelihood values are calculated. We recommend using the grid interval equal to 0.1.

initial   The initial values of the other model parameters (mu, log(sigma), log(tau), beta). We take log on sigma and tau for numerical stability.

data.flux   "True" if data are recorded on flux scale or "FALSE" if data are on magnitude scale.

delta.uniform.range

         The range of the Uniform prior distribution for the time delay.

micro   It determines the order of a polynomial regression model that accounts for the difference between microlensing trends. Default is 3. When zero is assigned, the Bayesian model fits a curve-shifted model.

Details

The function `entirelogprofilelikelihood` is used to obtain the entire profile likelihood curve over the given grid values of the time delay.

Value

The outcome of `entirelogprofilelikelihood` is the values of the log profile likelihood function over the given grid values of the time delay.

Author(s)

Hyungsuk Tak

References

Hyungsuk Tak, Kaisey Mandel, David A. van Dyk, Vinay L. Kashyap, Xiao-Li Meng, and Aneta Siemiginowska (2017). "Bayesian Estimates of Astronomical Time Delays between Gravitationally Lensed Stochastic Light Curves," The Annals of Applied Statistics, 11 (3), 1309-1348.

Examples

```
# Loading datasets
data(simple)
head(simple)

#################################################
# Time delay estimation via profile likelihood #
#################################################


# Subset (data for image A) of the typical quasar data set
lcA <- simple[, 1 : 3]

# Another subset (data for image B) of the typical quasar data set
# The observation times for image B are not necessarily the same as those for image A
lcB <- simple[, c(1, 4, 5)]

# The two subsets do not need to have the same number of observations
# For example, here we add one more observation time for image B
lcB <- rbind(lcB, c(290, 1.86, 0.006))

dim(lcA)
```

```
dim(lcB)

###### The entire profile likelihood values on the grid of values of the time delay.

# Cubic microlensing model
ti1 <- lcB[, 1]
ti2 <- lcB[, 1]^2
ti3 <- lcB[, 1]^3
ss <- lm(lcB[, 2] - mean(lcA[, 2]) ~ ti1 + ti2 + ti3)

initial <- c(mean(lcA[, 2]), log(0.01), log(200), ss$coefficients)
delta.uniform.range <- c(0, 100)
grid <- seq(0, 100, by = 0.1)
# grid interval "by = 0.1" is recommended,
# but users can set a finer grid of values of the time delay.


logprof <- entirelogprofilelikelihood(data.lcA = lcA, data.lcB = lcB, grid = grid,
                                      initial = initial, data.flux = FALSE,
                                 delta.uniform.range = delta.uniform.range, micro = 3)
plot(grid, logprof, type = "l",
     xlab = expression(bold(Delta)),
     ylab = expression(bold(paste("log L"[prof], "(", Delta, ")"))))
prof <- exp(logprof - max(logprof))  # normalization
plot(grid, prof, type = "l",
     xlab = expression(bold(Delta)),
     ylab = expression(bold(paste("L"[prof], "(", Delta, ")"))))
```

---

simple	*Simulated simple data of a doubly-lensed quasar*

Description

We simulated a simple data set of a doubly-lensed quasar with generative parameter values (Delta, beta0, mu, sigma, tau) equal to (50, 2, 0, 0.03, 100). The number of epochs is 78 and measurement errors are the same to 0.005.

Usage

```
data(simple)
```

Format

A simple data set of a doubly-lensed quasar:

time observation times

x magnitudes of light A

se.x measurement errors of x

y magnitudes of light B

se.y measurement errors of y

Examples

```
data(simple)
```

---

simple.band1	*Simulated simple data of a doubly-lensed quasar observed in band 1*

Description

A 50 by 5 matrix containing a data set of a doubly-lensed quasar observed in band 1, which is modified from realistic data.

Usage

```
data(simple.band1)
```

Format

A simple data set of a doubly-lensed quasar:

time observation times

x magnitudes of light A observed in band 1

se.x measurement errors of x

y magnitudes of light B observed in band 1

se.y measurement errors of y

Examples

```
data(simple.band1)
```

---

simple.band2   *Simulated simple data of a doubly-lensed quasar observed in band 2*

Description

A 30 by 5 matrix containing a data set of a doubly-lensed quasar observed in band 2, which is modified from realistic data.

Usage

```
data(simple.band2)
```

Format

A simple data set of a doubly-lensed quasar:

time observation times

x magnitudes of light A observed in band 2

se.x measurement errors of x

y magnitudes of light B observed in band 2

se.y measurement errors of y

Examples

```
data(simple.band2)
```

---

timedelay   *Time Delay Estimation for Stochastic Time Series of Gravitationally Lensed Quasars*

Description

The R package **timedelay** provides a toolbox to estimate the time delay between the brightness time series of gravitationally lensed quasar images via Bayesian and profile likelihood approaches. The model is based on a state-space representation for irregularly observed time series data generated from a latent continuous-time Ornstein-Uhlenbeck process. Our Bayesian method adopts scientifically motivated hyper-prior distributions and a Metropoli-Hastings within Gibbs sampler, producing posterior samples of the model parameters that include the time delay. A profile likelihood of the time delay is a simple approximation to the marginal posterior distribution of the time delay. Both Bayesian and profile likelihood approaches complement each other, producing almost identical results; the Bayesian way is more principled but the profile likelihood is easier to be implemented. A new functionality is added in version 1.0.9 for estimating the time delay between doubly-lensed light curves observed in two bands.

Details

Package:	timedelay
Type:	Package
Version:	1.0.11
Date:	2020-05-18
License:	GPL-2
Main functions:	bayesian, bayesian.multiband, entirelogprofilelikelihood

Author(s)

Hyungsuk Tak, Kaisey Mandel, David A. van Dyk, Vinay L. Kashyap, Xiao-Li Meng, and Aneta Siemiginowska

Maintainer: Hyungsuk Tak <hyungsuk.tak@gmail.com>

References

Hyungsuk Tak, Kaisey Mandel, David A. van Dyk, Vinay L. Kashyap, Xiao-Li Meng, and Aneta Siemiginowska (2017). "Bayesian Estimates of Astronomical Time Delays between Gravitationally Lensed Stochastic Light Curves," The Annals of Applied Statistics, 11 (3), 1309-1348. Hyungsuk Tak, Xiao-Li Meng, and David A. van Dyk (2018), "A Repelling-Attracting Metropolis Algorithm for Multimodality", Journal of Computational and Graphical Statistics, 27 (3), 479-490. Zhirui Hu and Hyungsuk Tak (2020+), "Modeling Stochastic Variability in Multi-Band Time Series Data," arXiv:2005.08049.

Examples

```
# Loading datasets
data(simple)
head(simple)

# Subset (data for image A) of the typical quasar data set
lcA <- simple[, 1 : 3]

# Another subset (data for image B) of the typical quasar data set
# The observation times for image B are not necessarily the same as those for image A
lcB <- simple[, c(1, 4, 5)]

# The two subsets do not need to have the same number of observations
# For example, here we add one more observation time for image B
lcB <- rbind(lcB, c(290, 1.86, 0.006))

dim(lcA)
dim(lcB)

#################################################
# Time delay estimation via Bayesian approach #
```

```
    ###############################################


    # Cubic microlensing model (m = 3)
    output <- bayesian(data.lcA = lcA, data.lcB = lcB,
                       data.flux = FALSE, theta.ini = c(0, 0.01, 200),
                       delta.ini = 50, delta.uniform.range = c(0, 100),
                       delta.proposal.scale = 1,
                       tau.proposal.scale = 3,
                       tau.prior.shape = 1, tau.prior.scale = 1,
                       sigma.prior.shape = 1, sigma.prior.scale = 2 / 10^7,
                       asis = TRUE, micro = 3,
                       sample.size = 100, warmingup.size = 50)

    names(output)
    # hist(output$delta)
    # plot(output$delta, type = "l")
    # acf(output$delta)

    ### Argument description

    # data.lcA: An n1 by three matrix for image A (light curve A) with
    #           n1 observation times in the first column,
    #           n1 observed magnitudes (or in flux) in the second column,
    #           n1 measurement errors for image A in the third column.

    # data.lcB: An n2 by three matrix for image B (light curve B) with
    #           n2 observation times in the first column,
    #           n2 observed magnitudes (or in flux) in the second column,
    #           n2 measurement errors for image A in the third column.

# data.flux: "True" if data are recorded on flux scale or "FALSE" if data are on magnitude scale.
 # If your observed time series can take on negative values, then data.flux = FALSE.

 # theta.ini: Initial values of theta = (mu, sigma, tau) for MCMC.
 # delta.ini: Initial values of the time delay for MCMC.
 # delta.uniform.range: The range of the Uniform prior distribution for the time delay.
 #                      The feasible entire support is
 #            c(min(simple[, 1]) - max(simple[, 1]), max(simple[, 1]) - min(simple[, 1]))

 # delta.proposal.scale: The proposal scale of the Metropolis step for the time delay.
 # tau.proposal.scale: The proposal scale of the Metropolis-Hastings step for tau.

# tau.prior.shape: The shape parameter of the Inverse-Gamma hyper-prior distribution for tau.
# tau.prior.scale: The scale parameter of the Inverse-Gamma hyper-prior distribution for tau.
 # sigma.prior.shape: The shape parameter of
 #                    the Inverse-Gamma hyper-prior distribution for sigma^2.
 # sigma.prior.scale: The scale parameter of
 #                    the Inverse-Gamma hyper-prior distribution for sigma^2.
 # micro: It determines the order of a polynomial regression model that accounts
 #        for the difference between microlensing trends. Default is 3.
 #        When zero is assigned, the Bayesian model fits a curve-shifted model.
```

```
# asis: "TRUE" if we use the ancillarity-sufficiency interweaving strategy (ASIS)
#        for c (always recommended)
# multimodality: "TRUE" if we use the repelling-attracting Metropolis algorithm
#                for sampling Delta in the presence of multimodality.
#                Please make sure that "adaptive.delta = FALSE" so that
#                adaptive MCMC for Delta is not used in the presence of multimodality.

# adaptive.freqeuncy: The adaptive MCMC is applied for every specified frequency.
#                     If it is specified as 100,
#                     the adaptive MCMC is applied to every 100th iterstion.
# adaptive.delta: "TRUE" if we use the adaptive MCMC for the time delay.
# adaptive.delta.factor: The factor, exp(adaptive.delta.factor) or exp(-adaptive.delta.factor),
#                        multiplied to the proposal scale of the time delay for adaptive MCMC.
#                          Default is 0.01.

# adaptive.tau: "TRUE" if we use the adaptive MCMC for tau.
# adaptive.tau.factor: The factor, exp(adaptive.tau.factor) or exp(-adaptive.tau.factor),
#                        multiplied to the proposal scale of tau for adaptive MCMC.

# sample.size: The number of posterior samples for each parameter
# warmingup.size: The number of burn-ins

# Type ?bayesian for further details on graphical model checking.

################################################
# Time delay estimation via profile likelihood #
################################################

###### The entire profile likelihood values on the grid of values of the time delay.


# Cubic microlensing model
ti1 <- lcB[, 1]
ti2 <- lcB[, 1]^2
ti3 <- lcB[, 1]^3
ss <- lm(lcB[, 2] - mean(lcA[, 2]) ~ ti1 + ti2 + ti3)

initial <- c(mean(lcA[, 2]), log(0.01), log(200), ss$coefficients)
delta.uniform.range <- c(0, 100)
grid <- seq(0, 100, by = 0.1)
# grid interval "by = 0.1" is recommended for accuracy,
# but users can set a finer grid of values of the time delay.

###  Running the following codes takes more time than CRAN policy
### Please type the following lines without "#" to run the function and to see the results

#  logprof <- entirelogprofilelikelihood(data.lcA = lcA, data.lcB = lcB, grid = grid,
#                                         initial = initial, data.flux = FALSE,
#                                         delta.uniform.range = delta.uniform.range, micro = 3)

#  plot(grid, logprof, type = "l",
#       xlab = expression(bold(Delta)),
#       ylab = expression(bold(paste("log L"[prof], "(", Delta, ")"))))
```

```
#  prof <- exp(logprof - max(logprof))  # normalization
#  plot(grid, prof, type = "l",
#        xlab = expression(bold(Delta)),
#        ylab = expression(bold(paste("L"[prof], "(", Delta, ")"))))

### Argument description

# data.lcA: The data set (n by 3 matrix) for light curve A
# (1st column: observation times, 2nd column: values of fluxes or magnitudes,
#  3rd column: measurement errors)
# data.lcB: The data set (w by 3 matrix) for light curve B
# (1st column: observation times, 2nd column: values of fluxes or magnitudes,
#  3rd column: measurement errors)
# grid: the vector of grid values of the time delay
#        on which the log profile likelihood values are calculated.
# initial: The initial values of the other model parameters (mu, log(sigma), log(tau), beta)
# data.flux: "True" if data are recorded on flux scale or "FALSE" if data are on magnitude scale.
# delta.uniform.range: The range of the Uniform prior distribution for the time delay.
# micro: It determines the order of a polynomial regression model that accounts
#        for the difference between microlensing trends. Default is 3.
#        When zero is assigned, the Bayesian model fits a curve-shifted model.
```

Index