Package ‘BIKER’

October 6, 2021

Title Bayesian Inference/Inversion of the K600 Evasion Rate

Version 0.0.0.9000

Description Uses Bayesian inference + SWOT observations to estimate the normalized gas transfer velocity from rivers, no in situ info needed.

License file LICENSE

Encoding UTF-8

LazyData true

Biarch true

Depends R (>= 3.4.0)

Imports methods, Rcpp (>= 0.12.0), rstan (>= 2.18.1), rstantools (>= 2.0.0), dplyr, reshape2, rlang, settings

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), rstan (>= 2.18.1), StanHeaders (>= 2.18.0)

SystemRequirements GNU make

RoxygenNote 7.1.2

URL `https://github.com/craigbrinkerhoff/BIKER`

BugReports `https://github.com/craigbrinkerhoff/BIKER/issues`

NeedsCompilation yes

Author Craig Brinkerhoff [aut, cre] (<https://orcid.org/0000-0001-6701-4835>)

Maintainer Craig Brinkerhoff <`cbrinkerhoff@umass.edu`>

Archs i386, x64

R topics documented:

`BIKER-package` *The 'BIKER' package.*

Description

Uses Bayesian Inference + SWOT observations to estimate normalized gas transfer velocity from large rivers

References

Stan Development Team (2020). RStan: the R interface to Stan. R package version 2.19.3. https://mc-stan.org

---

`biker_check_args` *Performs the following checks: - types: - everything else matrix - dimensions: - all matrices have same dims*

Description

Performs the following checks: - types: - everything else matrix - dimensions: - all matrices have same dims

Usage

```
biker_check_args(datalist)
```

Arguments

`datalist`	A list of biker data inputs

`biker_check_nas` *Add missing-data inputs to data list*

Description

Binary matrices indicating where data are/aren't missing are added to the data list. This is required in order to run ragged-array data structures in the stanfile.

Usage

```
biker_check_nas(datalist)
```

Arguments

`datalist` a list of BIKER inputs

Details

Previously this function omitted any times with missing data, but now that ragged arrays are accommodated in the stanfile the operations are entirely different.

`biker_data` *Preprocess data for BIKER estimation*

Description

Produces a bikerdata object that can be passed to biker_estimate function

Usage

```
biker_data(w, s, dA, priorQ, max_xs = 30L, seed = NULL)
```

Arguments

`w` Matrix (or data frame) of widths: time as columns, space as rows

`s` Matrix of slopes: time as columns, space as rows

`dA` Matrix of area above base area: time as columns, space as rows

`priorQ` Mean annual flow prior for Q

`max_xs` Maximum number of cross-sections to allow in data. Used to reduce sampling time. Defaults to 30.

`seed` RNG seed to use for sampling cross-sections, if nx > max_xs.

`biker_estimate` *Estimate BIKER*

Description

Fits a BIKER model of one of several variants using Hamiltonian Monte Carlo.

Usage

```
biker_estimate(
  bikerdata,
  bikerpriors,
  cores = getOption("mc.cores", default = parallel::detectCores()),
  meas_error = FALSE,
  chains = 3L,
  iter = 1000L,
  CI = 0.95,
  chainExtract = "all",
  pars = NULL,
  include = FALSE,
  suppressOutput = 1,
  ...
)
```

Arguments

`bikerdata` A bikerdata object, as produced by `biker_data()`

`bikerpriors` A bikerpriors object. If none is supplied, defaults are used from calling `biker_priors(bikerdata)` (with no other arguments).

`cores` Number of processing cores for running chains in parallel. See `?rstan::sampling`. Defaults to `parallel::detectCores()`.

`meas_error` Should we run with latent variables accounting for uncertainity in SWOT measurements. LEAVE THIS OFF, IT IS IN ACTIVE DEVELOPMENT

`chains` A positive integer specifying the number of Markov chains. The default is 3.

`iter` Number of iterations per chain (including warmup). Defaults to 1000.

`CI` A postive integer between 0 and 1 indicating the confidence interval to return with the estimate. Default is 0.95.

`chainExtract` Either an integer specifying which chain(s) to extract statistics from, or "all" (the default), in which case they are extracted from all chains.

`pars` (passed to `rstan::sampling()`) A vector of character strings specifying parameters of interest to be returned in the stanfit object. If not specified, a default parameter set is returned.

`include` (passed to `rstan::sampling()`) Defaults to FALSE, which excludes parameters specified in `pars` from the returned model.

`suppressOutput` (passed to `rstan::sampling()`) Determines whether stan sampling updates are printed to the console. Defaults to 1. Any value <= 0 will suppress output.

`...` Other arguments passed to rstan::sampling() for customizing the Monte Carlo sampler `biker_priors` *Establish prior hyperparameters for BIKER estimation*

Description

Produces a bikerpriors object that can be passed to biker_estimate function

Usage

```
biker_priors(bikerdata, ...)
```

Arguments

`bikerdata` An object of class bikerdata, as returned by `biker_data`

`...` Optional manually set parameters. Unquoted expressions are allowed, e.g. `logk_sd = cv2sigma(0.8`
Additionally, any variables present in `bikerdata` may be referenced, e.g. `lowerbound_logk = log(m`

`classify_func` *Classify river for expert framework*

Description

Classify river for expert framework

Usage

```
classify_func(Wobs)
```

Arguments

`Wobs` observed widths matrix

`cv2sigma` *Convert coefficient of variation to sigma parameter of lognormal diistribution*

Description

Convert coefficient of variation to sigma parameter of lognormal diistribution

Usage

```
cv2sigma(cv)
```

Arguments

`cv` Coefficient of variation

estimate_A0SD *Estimate base cross-sectional area SD using bam dat*

Description

Estimate base cross-sectional area SD using bam dat

Usage

```
estimate_A0SD(Wobs)
```

Arguments

Wobs	Observed W,as a space-down, time-across matrix.

estimate_logA0 *Estimate base cross-sectional area using bam data*

Description

Estimate base cross-sectional area using bam data

Usage

```
estimate_logA0(Wobs)
```

Arguments

Wobs	Observed W,as a space-down, time-across matrix

estimate_logk *Estimate k_hat using biker data for k600 model*

Description

Estimate k_hat using biker data for k600 model

Usage

```
estimate_logk(Sobs, priorQ)
```

Arguments

Sobs	Observed S,as a space-down, time-across matrix
priorQ	mean annual flow estimate, single number

`estimate_logksd` *Estimate k sd prior using biker data for k600 model*

Description

Estimate k sd prior using biker data for k600 model

Usage

```
estimate_logksd(Sobs)
```

Arguments

`Sobs`	Observed S,as a space-down, time-across matrix

`estimate_logn` *Estimate manning's n using bam data*

Description

Estimate manning's n using bam data

Usage

```
estimate_logn(Wobs, Sobs)
```

Arguments

`Wobs`	Observed W,as a space-down, time-across matrix
`Sobs`	Observed s, as a space-down, time-across matrix

`estimate_lognSD` *Estimate manning's n SD using bam dat*

Description

Estimate manning's n SD using bam dat

Usage

```
estimate_lognSD(Wobs)
```

Arguments

`Wobs`	Observed W,as a space-down, time-across matrix.

`estimate_lowerboundA0` *Estimate base cross-sectional area lowerbound using bam data*

Description

Estimate base cross-sectional area lowerbound using bam data

Usage

```
estimate_lowerboundA0(Wobs)
```

Arguments

`Wobs`	Observed W,as a space-down, time-across matrix.

`estimate_upperboundA0` *Estimate base cross-sectional area upperbound using bam data*

Description

Estimate base cross-sectional area upperbound using bam data

Usage

```
estimate_upperboundA0(Wobs)
```

Arguments

`Wobs`	Observed W,as a space-down, time-across matrix.

`prior_settings` *Options manager for BIKER default prior settings for k600 model*

Description

Options manager for BIKER default prior settings for k600 model

Usage

```
prior_settings(..., .__defaults = FALSE, .__reset = FALSE)
```

Arguments

`...`	(Optional) named settings to query or set.
`.__defaults`	See `?settings::option_manager`
`.__reset`	See `?settings::option_manager`

`sample_xs` *Take a random sample of a bikerdata object’s cross-sections.*

Description

Take a random sample of a bikerdata object’s cross-sections.

Usage

```
sample_xs(bikerdata, n, seed = NULL)
```

Arguments

`bikerdata`	a bikerdata object, as returned by `biker_data()`
`n`	Number of cross-sections to
`seed`	option RNG seed, for reproducibility.

Index