Package 'bennu'

July 22, 2025

Title Bayesian Estimation of Naloxone Kit Number Under-Reporting

Version 0.3.0

Description Bayesian model and associated tools for generating estimates of total naloxone kit numbers distributed and used from naloxone kit orders data. Provides functions for generating simulated data of naloxone kit use and functions for generating samples from the posterior.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

Biarch true

Depends R (>= 3.4.0)

Imports dplyr, ggplot2, glue, lifecycle, magrittr, methods, Rcpp (>= 0.12.0), RcppParallel (>= 5.0.1), rlang, rstan (>= 2.26.0), rstantools (>= 2.2.0), scales, tidybayes, tidyr

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), RcppParallel (>= 5.0.1), rstan (>= 2.26.0), StanHeaders (>= 2.26.0)

SystemRequirements GNU make

Suggests bayesplot, covr, knitr, posterior, rmarkdown, stringr, testthat (>= 3.0.0)

Config/testthat/edition 3

URL https://sempwn.github.io/bennu/

BugReports https://github.com/sempwn/bennu/issues

VignetteBuilder knitr

NeedsCompilation yes

Author Mike Irvine [aut, cre, cph] (ORCID: <https://orcid.org/0000-0003-4785-8998>), Samantha Bardwell [ctb], Andrew Johnson [ctb]

Maintainer Mike Irvine <mike.irvine@bccdc.ca>

Repository CRAN

Date/Publication 2023-09-14 00:20:02 UTC

Contents

`bennu-package` *The 'bennu' package.*

Description

Bayesian Estimation of Naloxone use Number Under-reporting

References

Stan Development Team (2020). RStan: the R interface to Stan. R package version 2.21.2. https://mc-stan.org

---

`est_naloxone` *Run Bayesian estimation of naloxone number under-reporting*

Description

Samples from Bayesian model using input from data frame

Usage

```
est_naloxone(
  d,
  psi_vec = c(0.7, 0.2, 0.1),
  max_delays = 3,
  delay_alpha = 2,
  delay_beta = 1,
  run_estimation = TRUE,
  rw_type = 1,
  chains = 4,
```

```
  iter = 2000,
  seed = 42,
  adapt_delta = 0.85,
  ...
)
```

Arguments

d	data frame with format **regions** unique id for region **times** time in months **Orders** Kits ordered **Reported_Used** Kits reported as used **Reported_Distributed** Kits reported as distributed **region_name** Optional label for region
psi_vec	reporting delay distribution
max_delays	maximum delay from kit ordered to kit distributed
delay_alpha	shape parameter for order to distributed delay distribution
delay_beta	shape parameter for order to distributed delay distribution
run_estimation	if TRUE will sample from posterior otherwise will sample from prior only
rw_type	1 - random walk of order one. 2 - random walk of order 2.
chains	A positive integer specifying the number of Markov chains. The default is 4.
iter	A positive integer specifying the number of iterations for each chain (including warmup). The default is 2000.
seed	Seed for random number generation
adapt_delta	(double, between 0 and 1, defaults to 0.8)
...	other parameters to pass to rstan::sampling

Value

An S4 rstan::stanfit class object containing the fitted model

See Also

Other inference: `est_naloxone_vec()`

Examples

```
## Not run:
library(rstan)
library(bayesplot)

rstan_options(auto_write = TRUE)
options(mc.cores = parallel::detectCores(logical = FALSE))

d <- generate_model_data()
```

```
fit <- est_naloxone(d, iter = 100, chains = 1)
mcmc_pairs(fit,
  pars = c("sigma", "mu0"),
  off_diag_args = list(size = 1, alpha = 0.5)
)

## End(Not run)
```

`est_naloxone_vec` *Run Bayesian estimation of naloxone number under-reporting*

Description

Samples from Bayesian model

Usage

```
est_naloxone_vec(
  N_region,
  N_t,
  N_distributed,
  regions,
  times,
  Orders2D,
  Reported_Distributed,
  Reported_Used,
  region_name,
  psi_vec = c(0.7, 0.2, 0.1),
  max_delays = 3,
  delay_alpha = 2,
  delay_beta = 1,
  run_estimation = TRUE,
  rw_type = 1,
  chains = 4,
  iter = 2000,
  seed = 42,
  adapt_delta = 0.85,
  ...
)
```

Arguments

`N_region`	Number of regions
`N_t`	number of time steps
`N_distributed`	Number of samples of reporting for distribution of kits
`regions`	vector (time, region) of regions (coded 1 to N_region)
`times`	vector (time, region) of regions (coded 1 to N_t)

Orders2D	vector (time, region) of orders
Reported_Distributed	vector (time, region) reported as distributed
Reported_Used	vector (time, region) reported as used
region_name	bring in region names
psi_vec	reporting delay distribution
max_delays	maximum delay from kit ordered to kit distributed
delay_alpha	shape parameter for order to distributed delay distribution
delay_beta	shape parameter for order to distributed delay distribution
run_estimation	if TRUE will sample from posterior otherwise will sample from prior only
rw_type	1 - random walk of order one. 2 - random walk of order 2.
chains	A positive integer specifying the number of Markov chains. The default is 4.
iter	A positive integer specifying the number of iterations for each chain (including warmup). The default is 2000.
seed	Seed for random number generation
adapt_delta	(double, between 0 and 1, defaults to 0.8)
...	other parameters to pass to rstan::sampling

Value

An S4 rstan::stanfit class object containing the fitted model

See Also

Other inference: `est_naloxone()`

`generate_model_data` *generate model data for testing purposes*

Description

[Deprecated]

Simulate kits ordered and kits distributed for a set number of regions and time-points.

The kits ordered simulation is a simple square-term multiplied by `region_coeffs`. For example if `region_coeffs = c(1,2)` then the number of kits ordered at month 12 are `c(1,2) * 12^2 = c(144,288)`.

The probability of kit use in time is assumed to increase linearly in inverse logit space at a constant rate `0.1`. The probability of reporting for each month and region is iid distributed $logit^{-1}(p) \sim N(2,5)$ which produces a mean reporting rate of approximately 88%

Usage

```
generate_model_data(
  N_t = 24,
  region_coeffs = c(5, 0.5),
  c_region = c(-1, 2),
  reporting_freq = NULL
)
```

Arguments

`N_t`	number of time-points
`region_coeffs`	vector of coefficients for regions determining kit orders
`c_region`	logit probability of kit use per region
`reporting_freq`	The frequency that distribution data is provided. If `NULL` distribution frequency matches orders frequency

Value

A tibble

Orders Kit orders per time and region

regions Numeric index indicating region of orders and distributions

Reported_Used Number of kits reported as used

Reported_Distributed Number of kits reported as distributed

p_use Probability that a kit was used

p_reported Probability that a distributed kit was reported

times Index for time

region_name String index for the region

See Also

Other data generation: `model_random_walk_data()`

`kit_summary_table` *Summarize model fit*

Description

Provides a summary of:

- Estimated kits distributed
- Percentage of kits distributed that are reported
- Estimated kits used
- percentage of kits used that are reported
- percentage of kits orders that are used
- probability kit used if distributed

Usage

```
kit_summary_table(fit, ..., data = NULL, accuracy = 0.01, cri_range = 0.95)
```

Arguments

`fit`	stanfit object
`...`	variables to group by in estimate
`data`	data used for model fitting. Can also include `p_use` column which can be used to plot true values if derived from simulated data.
`accuracy`	A number to round to. Use (e.g.) 0.01 to show 2 decimal places of precision. If NULL, the default, uses a heuristic that should ensure breaks have the minimum number of digits needed to show the difference between adjacent values.
`cri_range`	The range of the credible interval e.g. 0.95

Value

A tibble::tibble

- Probability of kit use if distributed
- Estimated as distributed
- Proportion kits distributed that are reported
- Estimated kits used
- Proportion kits used that are reported
- Proportion kits ordered that are used

See Also

Other plots: `plot_kit_use()`

Examples

```
## Not run:
  fit <- est_naloxone(d)
  kit_summary_table(fit,regions,data = d)

## End(Not run)
```

`model_random_walk_data` *generate model data for testing purposes*

Description

Model generating process using random walk to match data generating model in Bayesian framework

Usage

```
model_random_walk_data(
  N_t = 24,
  region_coeffs = c(5, 0.5),
  c_region = c(-1, 2),
  sigma = 2,
  zeta = 0.5,
  mu0 = -1,
  Orders = NULL,
  reporting_freq = NULL
)
```

Arguments

`N_t`	number of time-points
`region_coeffs`	vector of coefficients for regions determining kit orders
`c_region`	logit probability of kit use per region
`sigma`	standard deviation of error in logit probability of kit use
`zeta`	standard deviation of random walk in logit space
`mu0`	initial condition of random walk in logit space
`Orders`	A 2D matrix of shape `length(region_coeffs)` by `N_t`
`reporting_freq`	The frequency that distribution data is provided. If `NULL` distribution frequency matches orders frequency

Value

A tibble

Orders Kit orders per time and region

regions Numeric index indicating region of orders and distributions

Reported_Used Number of kits reported as used

Reported_Distributed Number of kits reported as distributed

p_use Probability that a kit was used

p_reported Probability that a distributed kit was reported

times Index for time

region_name String index for the region

See Also

Other data generation: `generate_model_data()`

`plot_kit_use` *Plot of probability of naloxone kit use*

Description

plot can compare between two different model fits or a single model fit by region. If data are simulated then can also include in plot. For more details see the introduction vignette: `vignette("Introduction", package = "bennu")`

Usage

```
plot_kit_use(..., data = NULL)
```

Arguments

`...`	named list of stanfit objects
`data`	data used for model fitting. Can also include `p_use` column which can be used to plot true values if derived from simulated data.

Value

ggplot2 object

See Also

Other plots: `kit_summary_table()`

Index