Package: SSVS (via r-universe)

October 9, 2024

Title Functions for Stochastic Search Variable Selection (SSVS)

Version 2.0.0

Description Functions for performing stochastic search variable selection (SSVS) for binary and continuous outcomes and visualizing the results. SSVS is a Bayesian variable selection method used to estimate the probability that individual predictors should be included in a regression model. Using MCMC estimation, the method samples thousands of regression models in order to characterize the model uncertainty regarding both the predictor set and the regression parameters. For details see Bainter, McCauley, Wager, and Losin (2020) Improving practices for selecting a subset of important predictors in psychology: An application to predicting pain, Advances in Methods and Practices in Psychological Science 3(1), 66-80 <DOI:10.1177/2515245919885617>.

URL `https://github.com/sabainter/SSVS`

BugReports `https://github.com/sabainter/SSVS/issues`

Depends R (>= 2.10)

Imports bayestestR, BoomSpikeSlab, checkmate, ggplot2, graphics, rlang, stats

Suggests AER, bslib, foreign, glue, knitr, psych, reactable, readxl, rmarkdown, scales, shiny, shinyjs, shinyWidgets, testthat (>= 3.0.0), tools, utils

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Roxygen list(markdown = TRUE)

Config/testthat/edition 3

Repository https://sabainter.r-universe.dev

RemoteUrl https://github.com/sabainter/ssvs

RemoteRef HEAD

RemoteSha c58c8ff3ef92b87e2a0f683695b4487a316ef31c

Contents

dat 2
launch 2
plot.ssvs 3
ssvs 3
summary.ssvs 5

Index 7

`dat` *Example dataset for `ssvs` function @format A data frame with 74 records and 76 variables*

Description

Example dataset for `ssvs` function @format A data frame with 74 records and 76 variables

Usage

```
dat
```

Format

An object of class `data.frame` with 74 rows and 76 columns.

`launch` *Run an interactive analysis tool (Shiny app) that lets you perform SSVS in a browser*

Description

Run an interactive analysis tool (Shiny app) that lets you perform SSVS in a browser

Usage

```
launch()
```

`plot.ssvs` *Plot results of an SSVS model*

Description

Plot results of an SSVS model

Usage

```
## S3 method for class 'ssvs'
plot(x, threshold = 0.5, legend = TRUE, title = NULL, color = TRUE, ...)
```

Arguments

`x`	An SSVS result object obtained from `ssvs()`
`threshold`	An MIP threshold to show on the plot, must be between 0-1. If `NULL`, no threshold is used.
`legend`	If `TRUE`, show a legend for the shapes based on the threshold. Ignored if `threshold = NULL`.
`title`	The title of the plot. Set to `NULL` to use a default title.
`color`	If `TRUE`, the data points will be colored based on the threshold.
`...`	Ignored

Value

Creates a plot of the inclusion probabilities by variable

Examples

```
outcome <- "qsec"
predictors <- c("cyl", "disp", "hp", "drat", "wt", "vs", "am", "gear", "carb", "mpg")
results <- ssvs(x = predictors, y = outcome, data = mtcars, progress = FALSE)
plot(results)
```

`ssvs` *Perform SSVS for continuous and binary outcomes*

Description

For continuous outcomes, a basic Gibbs sampler is used. For binary outcomes, `BoomSpikeSlab::logit.spike()` is used.

Usage

```
ssvs(
  data,
  y,
  x,
  continuous = TRUE,
  inprob = 0.5,
  runs = 20000,
  burn = 5000,
  a1 = 0.01,
  b1 = 0.01,
  prec.beta = 0.1,
  progress = TRUE
)
```

Arguments

`data`	The dataframe used to extract predictors and response values
`y`	The response variable
`x`	The set of predictor variables
`continuous`	If `TRUE`, treat the response variable as continuous. If `FALSE`, treat the response variable as binary.
`inprob`	Prior inclusion probability value, which applies to all predictors. The prior inclusion probability reflects the prior belief that each predictor should be included in the model. A prior inclusion probability of .5 reflects the belief that each predictor has an equal probability of being included or excluded. Note that a value of .5 also implies a prior belief that the true model contains half of the candidate predictors. The prior inclusion probability will influence the magnitude of the marginal inclusion probabilities (MIPs), but the relative pattern of MIPs is expected to remain fairly consistent, see Bainter et al. (2020) for more information.
`runs`	Total number of iterations (including burn-in). Results are based on the Total - Burn-in iterations.
`burn`	Number of burn-in iterations. Burn-in iterations are discarded warmup iterations used to achieve MCMC convergence. You may increase the number of burn-in iterations if you are having convergence issues.
`a1`	Prior parameter for Gamma(a,b) distribution on the precision (1/variance) residual variance. Only used when `continuous = TRUE`.
`b1`	Prior parameter for Gamma(a,b) distribution on the precision (1/variance) residual variance. Only used when `continuous = TRUE`.
`prec.beta`	Prior precision (1/variance) for beta coefficients. Only used when `continuous = TRUE`.
`progress`	If `TRUE`, show progress of the model creation. When `continuous = TRUE`, progress plots will be created for every 1000 iterations. When `continuous = FALSE`, 10 progress messages will be printed. Only used when `continuous = TRUE`.

Value

An SSVS object that can be used in `summary()` or `plot()`.

Examples

```
# Example 1: continuous response variable
outcome <- "qsec"
predictors <- c("cyl", "disp", "hp", "drat", "wt", "vs", "am", "gear", "carb", "mpg")
results <- ssvs(data = mtcars, x = predictors, y = outcome, progress = FALSE)

# Example 2: binary response variable
library(AER)
data(Affairs)
Affairs$hadaffair[Affairs$affairs > 0] <- 1
Affairs$hadaffair[Affairs$affairs == 0] <- 0
outcome <- "hadaffair"
predictors <- c("gender", "age", "yearsmarried", "children", "religiousness",
"education", "occupation", "rating")
results <- ssvs(data = Affairs, x = predictors, y = outcome, continuous = FALSE, progress = FALSE)
```

`summary.ssvs` *Summarize results of an SSVS model*

Description

Summarize results from SSVS including marginal inclusion probabilities, Bayesian model averaged parameter estimates, and 95% highest posterior density credible intervals. Estimates and credible intervals are based on standardized X variables.

Usage

```
## S3 method for class 'ssvs'
summary(object, interval = 0.89, threshold = 0, ordered = FALSE, ...)
```

Arguments

`object`	An SSVS result object obtained from `ssvs()`
`interval`	The desired probability for the credible interval, specified as a decimal
`threshold`	Minimum MIP threshold where a predictor will be shown in the output, specified as a decimal
`ordered`	If `TRUE`, order the results based on MIP (in descending order)
`...`	Ignored

Value

A dataframe with results

Examples

```
outcome <- "qsec"
predictors <- c("cyl", "disp", "hp", "drat", "wt", "vs", "am", "gear", "carb", "mpg")
results <- ssvs(data = mtcars, x = predictors, y = outcome, progress = FALSE)
summary(results, interval = 0.9, ordered = TRUE)
```

Index

∗ **datasets**
    dat, 2

BoomSpikeSlab::logit.spike(), *3*

dat, 2

launch, 2

plot(), *5*
plot.ssvs, 3

ssvs, 3
ssvs(), *3*, *5*
summary(), *5*
summary.ssvs, 5