Package: voiageR (via r-universe)

July 22, 2026

Type Package

Title Value of Information Analysis in R

Version 1.0.0

Author voiage Development Team

Maintainer voiage Development Team <voiage@users.noreply.github.com>

Description An R interface to the Rust-backed voiage core for Value of Information analysis, with a direct Rust C ABI EVPI path and helper functions for Python environment management where advanced methods still use reticulate. Provides functions for calculating Expected Value of Perfect Information (EVPI), Expected Value of Partial Perfect Information (EVPPI), and Expected Value of Sample Information (EVSI).

URL https://github.com/edithatogo/voiage

BugReports https://github.com/edithatogo/voiage/issues

License Apache License (>= 2.0)

Encoding UTF-8

NeedsCompilation no

RoxygenNote 7.3.2

Config/testthat/edition 3

SystemRequirements Python (>= 3.12)

Suggests reticulate (>= 1.20), knitr (>= 1.45), rmarkdown (>= 2.26), testthat (>= 3.0.0)

VignetteBuilder knitr

Config/pak/sysreqs python3

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

Date/Publication 2026-07-22 14:22:19 UTC

RemoteUrl https://github.com/edithatogo/voiage

RemoteRef HEAD

RemoteSha 9e6c4fcca0c284f9993fc73c73897690ad549391

RemoteSubdir r-package/voiageR

Contents

`evpi` *Calculate Expected Value of Perfect Information*

Description

Calculates Expected Value of Perfect Information using the voiage Python library.

Usage

```
evpi(net_benefits, population = NULL, time_horizon = NULL,
  discount_rate = NULL)
```

Arguments

`net_benefits`	A matrix or data frame with PSA samples in rows and strategies in columns.
`population`	Optional population size for scaling.
`time_horizon`	Optional time horizon for scaling.
`discount_rate`	Optional discount rate for scaling.

Value

The calculated EVPI value.

Examples

```
## Not run:
net_benefits <- matrix(rnorm(2000), nrow = 1000, ncol = 2)
evpi(net_benefits)

## End(Not run)
```

evppi *Calculate Expected Value of Partial Perfect Information*

Description

Calculates Expected Value of Partial Perfect Information using the voiage Python library.

Usage

```
evppi(net_benefits, parameter_samples,
  parameters_of_interest = NULL, population = NULL, time_horizon = NULL,
  discount_rate = NULL)
```

Arguments

net_benefits	A matrix or data frame with PSA samples in rows and strategies in columns.
parameter_samples	A named list or data frame with parameter samples.
parameters_of_interest	Optional character vector of parameter names to retain. Defaults to all named parameters in parameter_samples.
population	Optional population size for scaling.
time_horizon	Optional time horizon for scaling.
discount_rate	Optional discount rate for scaling.

Value

The calculated EVPPI value.

Examples

```
## Not run:
net_benefits <- matrix(rnorm(2000), nrow = 1000, ncol = 2)
param_samples <- list(effect = rnorm(1000), cost = rnorm(1000))
evppi(net_benefits, param_samples)

## End(Not run)
```

evsi *Calculate Expected Value of Sample Information*

Description

Calculates Expected Value of Sample Information using the voiage Python library.

Usage

```
evsi(model_func, prior_samples, trial_design, population = NULL, time_horizon = NULL,
  discount_rate = NULL, method = c("two_loop", "regression", "efficient",
 "moment_based"), n_outer_loops = 100, n_inner_loops = 1000, metamodel = "linear")
```

Arguments

model_func	A function that takes parameter samples and returns net benefits.
prior_samples	A named list or data frame with prior parameter samples.
trial_design	A list of trial arm specifications.
population	Optional population size for scaling.
time_horizon	Optional time horizon for scaling.
discount_rate	Optional discount rate for scaling.
method	EVSI approximation method.
n_outer_loops	Number of outer Monte Carlo loops.
n_inner_loops	Number of inner Monte Carlo loops.
metamodel	Metamodel used for the efficient approximation.

Value

The calculated EVSI value.

Examples

```
## Not run:
model_func <- function(params) {
  cbind(params$param1, params$param2)
}
prior_samples <- list(param1 = rnorm(1000), param2 = rnorm(1000))
trial_design <- list(
  arms = list(
    list(name = "Treatment", sample_size = 50),
    list(name = "Control", sample_size = 50)
  )
)
evsi(model_func, prior_samples, trial_design, method = "efficient")

## End(Not run)
```

init_voiage *Initialize the voiage Python Module*

Description

Imports and caches the voiage Python module through reticulate.

Usage

```
init_voiage()
```

Value

Invisibly returns `NULL`.

Examples

```
## Not run:
init_voiage()

## End(Not run)
```

is_voiage_available *Check Whether voiage Is Available*

Description

Checks whether the voiage Python package is available to reticulate.

Usage

```
is_voiage_available()
```

Value

A logical value indicating whether the Python voiage package can be imported.

Examples

```
## Not run:
is_voiage_available()

## End(Not run)
```

set_voiage_env *Set the Python Environment for voiage*

Description

Configures the Python environment reticulate should use before importing voiage.

Usage

```
set_voiage_env(env, type = c("virtualenv", "conda"))
```

Arguments

env	The virtualenv or conda environment name or path.
type	The environment type, either "virtualenv" or "conda".

Value

Invisibly returns NULL.

Examples

```
## Not run:
set_voiage_env("myenv", type = "virtualenv")
set_voiage_env("myenv", type = "conda")

## End(Not run)
```

voiageR *voiageR: An R Interface to the voiage Python Library*

Description

An R interface to the voiage Python library for Value of Information analysis, with helper functions for Python environment management and availability checks via reticulate.

Details

The package exports init_voiage, evpi, evppi, evsi, is_voiage_available, and set_voiage_env.

Documentation policy:

- The release ships a PDF reference manual built from the package help pages.
- The release also ships a narrative vignette as the long-form companion guide.

Author(s)

voiage Development Team

Index