

Package: pfr (via r-universe)

February 18, 2025

Type Package

Title Interface to the 'C++' Library 'Pf'

Version 1.0.1

Description Builds and runs 'c++' code for classes that encapsulate state space model, particle filtering algorithm pairs. Algorithms include the Bootstrap Filter from Gordon et al. (1993) <[doi:10.1049/ip-f-2.1993.0015](https://doi.org/10.1049/ip-f-2.1993.0015)>, the generic SISR filter, the Auxiliary Particle Filter from Pitt et al (1999) <[doi:10.2307/2670179](https://doi.org/10.2307/2670179)>, and a variety of Rao-Blackwellized particle filters inspired by Andrieu et al. (2002) <[doi:10.1111/1467-9868.00363](https://doi.org/10.1111/1467-9868.00363)>. For more details on the 'c++' library 'pf', see Brown (2020) <[doi:10.21105/joss.02599](https://doi.org/10.21105/joss.02599)>.

License GPL (>= 3)

Imports inline (>= 0.3.19), methods, rstudioapi (>= 0.13)

RoxygenNote 7.2.1

Encoding UTF-8

Suggests BH, Rcpp (>= 1.0.11), RcppEigen, knitr (>= 1.39), rmarkdown (>= 2.23)

VignetteBuilder knitr, rmarkdown

NeedsCompilation no

Author Taylor Brown [aut, cre]
(<<https://orcid.org/0000-0003-4972-6251>>)

Maintainer Taylor Brown <trb5me@virginia.edu>

Date/Publication 2023-12-08 17:40:02 UTC

Repository <https://tbrown122387.r-universe.dev>

RemoteUrl <https://github.com/cran/pfr>

RemoteRef HEAD

RemoteSha 9ad6ce48dbfac9b2ff3398ee98860815341af209

Contents

| | |
|--------------------------------|----------|
| buildModelFuncs | 2 |
| createPFCPPTemplates | 3 |
| Index | 4 |

| | |
|-----------------|--|
| buildModelFuncs | <i>Build c++ particle filtering code for your R session.</i> |
|-----------------|--|

Description

Build c++ particle filtering code for your R session.

Usage

```
buildModelFuncs(myDir, modelName, verbose = FALSE)
```

Arguments

| | |
|-----------|--|
| myDir | directory with your three code files (i.e. model header, model source and export code) |
| modelName | your model name. Must be in all lowercase, and be a substring of the above-mentioned filenames |
| verbose | logical and passed in to inline::cxxfunction() |

Value

an Rcpp Module object

Examples

```
## Not run:
# compile everything from scratch
svol_lev <- buildModelFuncs("~/Desktop", "svol_leverage")

# then use your model's log-likelihood and filtering functions
svol_lev$svol_leverage_bswc_approx_LL(rnorm(100), c(.9, 0.0, 1.0, -.2))
svol_lev$svol_leverage_bswc_approx_filt(rnorm(100), c(.9, 0.0, 1.0, -.2))

## End(Not run)
```

createPFCPPTemplates *Create c++ template files for bootstrap filters (with or without covariates), auxiliary particle filters, sequential importance sampling with resampling filters, or Rao-Blackwellized/Marginal particle filters.*

Description

Create c++ template files for bootstrap filters (with or without covariates), auxiliary particle filters, sequential importance sampling with resampling filters, or Rao-Blackwellized/Marginal particle filters.

Usage

```
createPFCPPTemplates(modname, pfAlgo, fileDir, openNow = TRUE)
```

Arguments

| | |
|---------|--|
| modname | name of model in all lowercase |
| pfAlgo | Either "BSF", "APF", "BSWC", "SISR", "RBPFHMM", or "RBPFKALMAN" |
| fileDir | where to save files. Not saved if NULL (but three files are returned in list). |
| openNow | TRUE if you want to open this now in RStudio. Ignored if fileDir is NULL. |

Value

NULL if saving files, otherwise a list with three character vectors

Examples

```
# return in list of character strings
createPFCPPTemplates("coolmod", "BSF", fileDir = NULL)

## Not run:
# save three files to Desktop, and
# begin editing them in rstudio IDE
createPFCPPTemplates("coolmod", "BSF", fileDir = "~/Desktop/")

## End(Not run)
```

Index

buildModelFuncs, [2](#)

createPFCPPTemplates, [3](#)