Package ‘distrTEst’

April 23, 2016

Version 2.6
Date 2016-04-23
Title Estimation and Testing Classes Based on Package ‘distr’
Description Evaluation (S4-)classes based on package distr for evaluating procedures (estimators/tests) at data/simulation in a unified way.
Depends R(>= 2.6.0), methods, graphics, setRNG(>= 2006.2-1),
distrSim(>= 2.2)
Imports startupmsg, utils
Suggests distrEx(>= 2.2)
ByteCompile yes
Encoding latin1
License LGPL-3
URL http://distr.r-forge.r-project.org/
LastChangedDate {$LastChangedDate: 2015-11-07 22:47:41 +0100 (Sa, 07 Nov 2015) $}
LastChangedRevision {$LastChangedRevision: 1073 $}
SVNRevision 1091
NeedsCompilation no
Author Florian Camphausen [ctb] (contributed as student in the initial phase --2005),
Matthias Kohl [aut, cph],
Peter Ruckdeschel [cre, cph],
Thomas Stabla [ctb] (contributed as student in the initial phase --2005)
Maintainer Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>
Repository CRAN
Date/Publication 2016-04-23 15:14:11
R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>distrTEst-package</td>
<td>2</td>
</tr>
<tr>
<td>call.ev-methods</td>
<td>4</td>
</tr>
<tr>
<td>Data-methods</td>
<td>4</td>
</tr>
<tr>
<td>distrTEstoptions</td>
<td>5</td>
</tr>
<tr>
<td>estimator-methods</td>
<td>6</td>
</tr>
<tr>
<td>evaluate-methods</td>
<td>6</td>
</tr>
<tr>
<td>Evaluation-class</td>
<td>7</td>
</tr>
<tr>
<td>EvaluationList-class</td>
<td>9</td>
</tr>
<tr>
<td>filename-methods</td>
<td>11</td>
</tr>
<tr>
<td>name-methods</td>
<td>11</td>
</tr>
<tr>
<td>numericorNULL-class</td>
<td>11</td>
</tr>
<tr>
<td>plot-methods</td>
<td>12</td>
</tr>
<tr>
<td>print-methods</td>
<td>13</td>
</tr>
<tr>
<td>result-methods</td>
<td>13</td>
</tr>
<tr>
<td>savedata-methods</td>
<td>14</td>
</tr>
<tr>
<td>summary-methods</td>
<td>14</td>
</tr>
</tbody>
</table>

Index

16

distrTEst-package

distrTEst – Estimation and Testing Classes Based on Package distr

Description

distrTest provides (S4-)classes for evaluating procedures (estimators/tests) at data/simulation in a unified way based on package distr. This is achieved by means of the S4 class Evaluation. The package is based on our packages distr and distrSim, hence uses distribution classes and simulation classes as introduced there to capture the situation from which the simulations stem.

Details

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Package</td>
<td>distrTEst</td>
</tr>
<tr>
<td>Version</td>
<td>2.6</td>
</tr>
<tr>
<td>Date</td>
<td>2016-04-23</td>
</tr>
<tr>
<td>Depends</td>
<td>R(&gt;= 2.6.0), methods, graphics, setRNG(&gt;= 2006.2-1), distrSim(&gt;= 2.2)</td>
</tr>
<tr>
<td>Imports</td>
<td>startupmsg, utils</td>
</tr>
<tr>
<td>Suggests</td>
<td>distrEx(&gt;= 2.2)</td>
</tr>
<tr>
<td>LazyLoad</td>
<td>yes</td>
</tr>
<tr>
<td>License</td>
<td>LGPL-3</td>
</tr>
<tr>
<td>URL</td>
<td><a href="http://distr.r-forge.r-project.org/">http://distr.r-forge.r-project.org/</a></td>
</tr>
<tr>
<td>SVNRevision</td>
<td>1091</td>
</tr>
</tbody>
</table>
Classes

"Evaluation"
slots: [<name>({<class>})]
  name(character), filename(character), call.ev(call),
  Data(Dataclass), result(DataframeorNULL),
  estimator(OptionalalFunction)

"EvaluationList"
slots: [<name>({<class>})]
  name(character), EList(list)

Objects of class "Evaluation" are generated by a call to evaluate.

Methods

plot  plot method for "Evaluation" and for "EvaluationList"
print,show print/show method for "Evaluation" and for "EvaluationList"
summary summary method for "Evaluation" and for "EvaluationList"
Data accessor method for "Evaluation", and, for "EvaluationList"
  returns common Data

Slot accessors / -replacement functions

All slots are inspected / modified by corresponding accessors / -replacement functions, e.g. call.ev(X)
  or filename(X)<="myevaluation.sav" for an object of class "Evaluation".

Start-up-Banner

You may suppress the start-up banner/message completely by setting options("StartupBanner"="off")
  somewhere before loading this package by library or require in your R-code / R-session.
If option "StartupBanner" is not defined (default) or setting options("StartupBanner"=NULL)
  or options("StartupBanner"="complete") the complete start-up banner is displayed.
For any other value of option "StartupBanner" (i.e., not in c(NULL,"off","complete")) only
  the version information is displayed.
The same can be achieved by wrapping the library or require call into either suppressStartupMessages()
  or onlytypeStartupMessages(.,atypes="version").
As for general packageStartupMessage’s, you may also suppress all the start-up banner by wrapping
  the library or require call into suppressPackageStartupMessages() from startupmsg-
  version 0.5 on.

Package versions

Note: The first two numbers of package versions do not necessarily reflect package-individual
development, but rather are chosen for the distrXXX family as a whole in order to ease updating
"depends" information.

Note

Global options controlling the plots and summaries of EvaluationList objects may be inspected / set
by distrTESoptions() and getdistrTESoption().
Author(s)

Thomas Stabla <statho3@web.de>,
Florian Camphausen <fcampi@gmx.de>,
Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>,
Matthias Kohl <Matthias.Kohl@stamats.de>

Maintainer: Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>

References

A vignette for packages `distr`, `distrSim`, `distrTEst`, and `distrEx` is included into the mere documentation package `distrDoc` and may be called by `require("distrDoc");vignette("distr").` A homepage to this package is available under `http://distr.r-forge.r-project.org/`

See Also

distr-package, distrSim-package, setRNG

call.ev-methods

Methods for Function call.ev in Package `distrTEst`

Description

call.ev-methods

Methods

call.ev signature(object = "Evaluation"): returns the call which created the object

Data-methods

Methods for Function Data in Package `distrTEst`

Description

Data-methods

Methods

Data signature(object = "Evaluation"): returns the Data slot

Data signature(object = "EvaluationList"): returns the common Data slot of the respective list elements
distrTEstoptions

functions to change the global variables of the package 'distrTEst'

Description

With distrTEstoptions and getdistrTEstOption you may inspect and change the global variables used by package distrTEst.

Usage

distrTEstoptions(...)
getdistrTEstOption(x)

Arguments

... any options can be defined, using name = value or by passing a list of such tagged values.
x a character string holding an option name.

Details

Invoking distrTEstoptions() with no arguments returns a list with the current values of the options. To access the value of a single option, one should use getdistrTEstOption("MaxNumberofSummarizedEvaluations") e.g., rather than distrTEstoptions("MaxNumberofSummarizedEvaluations") which is a list of length one.

Value

distrTEstoptions() returns a list of the global options of distrTEst.
distrTEstoptions("MaxNumberofSummarizedEvaluations") returns the global option MaxNumberofSummarizedEvaluations as a list of length 1.
distrTEstoptions("MaxNumberofSummarizedEvaluations" = 3) sets the value of the global option MaxNumberofSummarizedEvaluations to 3. getdistrTEstOption("MaxNumberofSummarizedEvaluations") the current value set for option MaxNumberofSummarizedEvaluations.

Currently available options

MaxNumberofPlottedEvaluations maximal number of evaluations plotted; defaults to 6
MaxNumberofPlottedEvaluationDims maximal number of evaluation dimensions plotted in parallel; defaults to 6
MaxNumberofSummarizedEvaluations maximal number of evaluations summarized in parallel; defaults to 15
MaxNumberofPrintedEvaluations maximal number of evaluations printed in parallel; defaults to 15
Author(s)

Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>

See Also

options, getOption, distroptions, getdistrOption

Examples

distrTEstoptions()
distrTEstoptions("MaxNumberOfPlottedEvaluationDims")
distrTEstoptions("MaxNumberOfPlottedEvaluationDims" = 5)
# or
getdistrTEstOption("MaxNumberOfPlottedEvaluationDims")

estimator-methods

Methods for Function estimator in Package ‘distrTEst’

Description

estimator-methods

Methods

estimator signature(object = "Evaluation"): returns the estimator

evaluate-methods

Methods for Function evaluate in Package ‘distrTEst’

Description

evaluate-methods to produce objects of class "Evaluation"

Arguments

object the data set / simulation on which the evaluation takes place
estimator the estimation function used; should be able to deal with data in matrix form samplesize x obsDim, and, should return either a univariate result or a vector (with named coordinates, if possible).
resname (a vector of) character(s); the name for the univariate results or, in the case of multivariate results, and if the coordinates of the results have not yet been named, the basic name for them which is pasted to the coordinate number for each coordinate.
name character; the name for the Evaluation object; by default the (R-)name of the Data set object.
filename character; the filename for the Evaluation object (where it is to be saved to); by default the filename of the Data set object which is concatenated with the name of the estimator in savedata.
Details

besides the arguments determining the method dispatch, we have:

```r
evaluate(object, estimator, ..., resname = "res",
         name = as.character(substitute(object)),
         filename = filename(object))
```

Methods

```r
evaluate signature(object = "Dataclass", estimator = "function"): creates an object of class "Evaluation", see there for further information
evaluate signature(object = "Contsimulation", estimator = "function"): creates an object of class "Evaluation", see there for further information
```

See Also

Evaluation-class

Evaluation-class  

Class "Evaluation"

Description

When an estimator is used to data of the type "Dataclass" with the method evaluate, the result is an object of class "Evaluation".

Objects from the Class

Objects could be created by calls of the form `new("Evaluation", Data, estimator, [result, name, filename, call)`. It does not seem to be very useful to generate a new object this way, however. It is to be preferred to use "evaluate" with a Dataclass object!

Slots

```r
call.ev  Object of class "call": the call which created the object, e.g.; "evaluate(Dataclassobject,mean)"
Data  Object of class "Dataclass": the data set / simulation on which the evaluation takes place.
estimator  Object of class "OptionalFunction": estimation function used; this estimation function should be able to deal with data in matrix form `samplesize x obsDim` and should return either a univariate result or a vector (with named coordinates, if possible).
filename  Object of class "character": the filename of the evaluation; by default the filename of the Dataclass object, which was called by evaluate
name  Object of class "character": the name of the evaluation; by default the name of the Dataclass object, which was called by evaluate
result  Object of class "DataFrameorNULL": the result of the evaluation of the estimation on data
```
Accessors/Replacement functions

call.ev no replacement possible
estimator no replacement possible
filename replacement possible
name replacement possible
result no replacement possible

Methods

initialize signature(.Object = "Evaluation"): initialize method
plot signature(object = "Evaluation"): returns a boxplot of the result
print signature(object = "Evaluation"): returns the name of the data object, its filename, the estimator used and the result
savedata signature(object = "Evaluation"): saves the object in two files in the directory of R - one with data, one without as comment file (see example)
summary signature(object = "Evaluation"): returns the name of the data object, its filename, the estimator used and a statistical summary of the result

Note

The saved "evaluation" can be loaded with the usual load-command, the saved comment with the function cload.

Author(s)

Thomas Stabla <statho3@web.de>,
Florian Camphausen <fcampi@gmx.de>,
Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>,
Matthias Kohl <Matthias.Kohl@stamats.de>

See Also

Dataclass-class Simulation-class Contsimulation-class load cload savedata-methods plot-methods simulate-methods summary-methods

Examples

N <- Norm() # N is a standard normal distribution.
C <- Cauchy() # C is a Cauchy distribution
cs <- Contsimulation(filename = "csim",
runs = 5,
samplesize=5000,
seed=setRNG(),
distribution.id = N,
distribution.c = C,
rate = 0.1)
simulate(cs)
# Each of the 25000 random numbers is ideal (N-distributed) with 
# probability 0.9 and contaminated (C-distributed) with probability = 0.1 

summary(cs) 
ev1 <- evaluate(cs, mean, resname="mean") # estimates the data with mean 
ev1 # bad results 
ev2 <- evaluate(cs, median, resname="median") # estimates the data with median 
ev2 # better results because median is robust 
savedata(ev1) 
# saves the evaluation with result as "csim.mean" and without result as 
# "csim.mean.comment" in the working directory # of R - "csim" is the 
# filename of the Contsimulation object, mean the name of the estimator 
rm(ev1) 
cload("csim.mean") 
# loads the evaluation without result - the object is called ev1.comment 
ev1.comment 
load("csim.mean") # loads the evaluation with result 
ev1 
plot(ev1) 
# 
# another function to be evaluated: 
severalThings<- function(x) {list("mean"=mean(x),"sd"=sd(as.vector(x)), "mad"=mad(x))} 
ev3 <- evaluate(cs, severalThings, resname="several") 
plot(ev3) 
plot(ev3, ylim=c(0,10), col=c("blue","green", "red"))

Description

Several objects of class "Evaluation" may be gathered in a list of class "EvaluationList", if they all 
have the same result-format and also share the same data-set.

Objects from the Class

Objects may be created by the generating function EvaluationList, i.e.; EvaluationList(., ., name0 = "a list of "Evaluation", where all arguments passed through . . have to be objects of class "Evaluation", the corresponding 
result-slots have to contain data.frames of identical dimension; the corresponding calls have to have 
identical object-arguments (for the data set), and the corresponding Data-slots have to be identical.

Slots

name: Object of class "character": the name of the EvaluationList object 
Elist: Object of class "list": the list of Evaluation objects

Accessor/Replacement methods

Elist signature(object = "EvaluationList"): returns the list with the Evaluation objects 
name signature(object = "EvaluationList"): returns/modifies the name of the Evaluation-
List object
Methods

Data signature(object = "EvaluationList"): returns the common Data-slot of one of the Evaluation objects

plot signature(object = "EvaluationList"): returns grouped boxplots of the results

print signature(object = "EvaluationList"): for each list element returns the name of the data object, its filename, the estimator used and the result

show signature(object = "EvaluationList"): as print

summary signature(object = "EvaluationList"): returns the name of the data object, its filename, the estimator used and a statistical summary of the result

Author(s)

Thomas Stabla <statho3@web.de>,
Florian Camphausen <fcampi@gmx.de>,
Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>,
Matthias Kohl <Matthias.Kohl@stamats.de>

See Also

Dataclass-class Simulation-class Contsimulation-class Evaluation-class print-methods plot-methods simulate-methods summary-methods

Examples

N <- Norm() # N is a standard normal distribution.
C <- Cauchy() # C is a Cauchy distribution
cs <- Contsimulation(filename = "csim",
  runs = 15,
  samplesize=500,
  seed=setRNG(),
  distribution.id = N,
  distribution.c = C,
  rate = 0.1)

simulate(cs)
# Each of the 25000 random numbers is ideal (N-distributed) with
# probability 0.9 and contaminated (C-distributed) with probability = 0.1

summary(cs)
ev1 <- evaluate(cs, mean) # estimates the data with mean
ev1 # bad results

ev2 <- evaluate(cs,median) # estimates the data with median
ev2 # better results because median is robust

savedata(ev1)
# saves the EvaluationList with result as "csim.mean" and without result as
# "csim.mean.comment" in the working directory # of R - "csim" is the
# filename of the Contsimulation object, mean the name of the estimator

rm(ev1)
cload("csim.mean")
# loads the EvaluationList without result - the object is called ev1.comment
ev1.comment
load("csim.mean") # loads the EvaluationList with result
filename-methods

Methods for Function `filename` in Package ‘distrTEst’

Description
filename-methods

Methods

`filename` signature(object = "Evaluation"): returns the filename of the evaluated object

name-methods

Methods for Function `name` in Package ‘distrTEst’

Description
name-methods

Methods

`name` signature(object = "Evaluation"): returns the slot name of data object

`name<-` signature(.Object = "Evaluation"): modifies the slot name of data object

numericorNULL-class

Classes "numericorNULL", "CallorNULL", and "DataframeorNULL"

Description

auxiliary classes; may contain either a numeric vector or NULL [or a call / data.frame or NULL, respectively].

Objects from the Class

A virtual Class: No objects may be created from it.
Methods

No methods defined with class "numericorNULL", "CallorNULL", and "DataframeorNULL" in the signature.

Note

From version 1.8, the result slot of an object of class evaluation is of type "DataframeorNULL"

Author(s)

Thomas Stabla <statho@web.de>,
Florian Camphausen <fcampi@gmx.de>,
Peter Ruckdeschel <peter.ruckdeschel@uni-oldenburg.de>,
Matthias Kohl <Matthias.Kohl@stamats.de>

See Also

Evaluation-class

Description

plot-methods

Methods

plot signature(x = "Evaluation", y="missing"): returns a boxplot of the result

plot signature(x = "EvaluationList", y="missing"): regroups the list according to the different columns/coordinates of the result of the evaluation; for each such coordinate a boxplot is generated containing possibly several procedures and if evaluated at a Contsimulation object also grouped into evaluations on ideal and real data. The plot-arguments main and ylim may be transmitted coordinatewise (i.e.; a vector of length (result-dimension), respectively a 2 x (result-dimension) matrix) or globally, using the usual recycling rules.
### print-methods

**Methods for Functions print and show in Package ‘distrTEst’**

#### Description

print/show-methods

#### Methods

**print**

signature(x = "Evaluation"): returns the name of the data object, its filename, the estimator used and the result; optional arguments:

- **runs0**: the indices of runs to be summarized;
- **dims0**: the indices of result dimensions to be summarized;

  *internal argument:*

- **inList**: decides if name of Dataobject and Datafile are printed out (which is done if inList==FALSE); defaults to FALSE but is TRUE when summary is called from summary-method for signature(object = "EvaluationList");

**print**

signature(object = "EvaluationList"): after printing the name of the list, for each member of the list print is executed; optional arguments:

- **eval0**: the indices of evaluations to be summarized;— of this vector eval0 maximally MaxNumberOfSummarizedEvaluations evaluations are summarized where MaxNumberOfPrintedEvaluations is a global option, see `distrTEstoptions`
- **runs0**: the indices of runs to be summarized;
- **dims0**: the indices of observation dimensions to be summarized;

**show**

signature(x = "Evaluation"): the same as print (without optional arguments)

**show**

signature(x = "EvaluationList"): the same as print (without optional arguments)

### result-methods

**Methods for Function result in Package ‘distrTEst’**

#### Description

result-methods

#### Methods

**result**

signature(object = "Evaluation"): returns the result of an evaluation
Methods

**savedata** signature(object = "Evaluation"): saves the object in two files in the directory of R - one with data —filename = <filename>—, one without as comment file —filename = <filename>.comment— (see example); <filename>the filename can be specified in the optional argument filename; by default it is concatenated from the filename of the `dataclass` object and the estimatorname, which you may either pass as argument estimatorName or it is taken as the R-name of the corresponding R-function.

**Note**

For an example, see `Simulation-class` and `Contsimulation-class`

**See Also**

`Dataclass-class` `Simulation-class` `Contsimulation-class` `Evaluation-class`

Methods

**summary** signature(object = "Evaluation"): returns the name of the data object, its filename, the estimator used and a statistical summary of the result; optional arguments:

- **runs0** the indices of runs to be summarized;
- **dims0** the indices of result dimensions to be summarized;
- **inList** decides if name of Dataobject and Datafile are printed out (which is done if inLIST==FALSE); defaults to FALSE but is TRUE when summary is called from summary-method for signature(object = "EvaluationList");

**summary** signature(object = "EvaluationList"): returns, for each member of the list a summary of the corresponding Evaluation object; optional arguments:
eval0  the indices of evaluations to be summarized; of this vector eval0 maximally MaxNumberOfSummarizedEvaluations evaluations are summarized where MaxNumberOfSummarizedEvaluations is a global option, see distrTestoptions

runs0  the indices of runs to be summarized;

dims0  the indices of observation dimensions to be summarized;
Index

*Topic classes
  numericorNULL-class, 11
*Topic file
  filename-methods, 11
  savedata-methods, 14
*Topic hplot
  plot-methods, 12
*Topic list
  EvaluationList-class, 9
*Topic manip
  Evaluation-class, 7
  EvaluationList-class, 9
*Topic methods
  call.ev-methods, 4
  Data-methods, 4
  estimator-methods, 6
  evaluate-methods, 6
  filename-methods, 11
  name-methods, 11
  plot-methods, 12
  print-methods, 13
  result-methods, 13
  savedata-methods, 14
  summary-methods, 14
*Topic misc
  distrTEstoptions, 5
*Topic package
  distrTEst-package, 2
*Topic print
  print-methods, 13

call.ev (call.ev-methods), 4
call.ev, Evaluation-method
  (call.ev-methods), 4
call.ev-methods, 4
CallorNULL-class (numericorNULL-class), 11
cload, 8
Data (Data-methods), 4
Data, Evaluation-method (Data-methods), 4
Data, EvaluationList-method
  (Data-methods), 4
Data-methods, 4
DataframeorNULL-class
  (numericorNULL-class), 11
distroptions, 6
distrTEst (distrTEst-package), 2
distrTEst-package, 2
distrTEstoptions, 3, 5, 13, 15
Elist (EvaluationList-class), 9
Elist, EvaluationList-method
  (EvaluationList-class), 9
estimator (estimator-methods), 6
estimator, Evaluation-method
  (estimator-methods), 6
estimator-methods, 6
evaluate, 6
evaluate (evaluate-methods), 6
evaluate, Contsimulation, function-method
  (evaluate-methods), 6
evaluate, Dataclass, function-method
  (evaluate-methods), 6
evaluate-methods, 6
Evaluation-class, 7
EvaluationList (EvaluationList-class), 9
EvaluationList-class, 9
filename (filename-methods), 11
filename, Evaluation-method
  (filename-methods), 11
filename-methods, 11
getdistrOption, 6
getdistrTEstOption, 3
getdistrTEstOption (distrTEstoptions), 5
getOption, 6
initialize, Evaluation-method
  (Evaluation-class), 7
initialize, EvaluationList-method
   (EvaluationList-class), 9
load, 8
MaxNumberOfPlottedEvaluationDims
   (distrTestoptions), 5
MaxNumberOfPlottedEvaluations
   (distrTestoptions), 5
MaxNumberOfSummarizedEvaluationDims
   (distrTestoptions), 5
MaxNumberOfSummarizedEvaluations
   (distrTestoptions), 5
name (name-methods), 11
name, Evaluation-method (name-methods), 11
name, EvaluationList-method
   (EvaluationList-class), 9
name-methods, 11
name<-, EvaluationList-method
   (EvaluationList-class), 9
numericorNULL-class, 11
onlytypeStartupMessages, 3
options, 6
plot, Evaluation, missing-method
   (plot-methods), 12
plot, EvaluationList, missing-method
   (plot-methods), 12
plot-methods, 12
print, Evaluation-method
   (print-methods), 13
print, EvaluationList-method
   (print-methods), 13
print-methods, 13
result (result-methods), 13
result, Evaluation-method
   (result-methods), 13
result-methods, 13
savedata, 6
savedata (savedata-methods), 14
savedata, Evaluation-method
   (savedata-methods), 14
savedata, EvaluationList-method
   (EvaluationList-class), 9
savedata-methods, 14
setRNG, 4
show, Evaluation-method (print-methods), 13
show, EvaluationList-method
   (print-methods), 13
show-methods (print-methods), 13
summary, Evaluation-method
   (summary-methods), 14
summary, EvaluationList-method
   (summary-methods), 14
summary-methods, 14
suppressStartupMessages, 3