

Package ‘xpose4’

August 29, 2016

Title Tools for Nonlinear Mixed-Effect Model Building and Diagnostics

Version 4.5.3

Date 2014-11-24

Depends R (>= 2.2.0), methods, lattice, grid, gam

Imports Hmisc, survival

Description Xpose is collection of functions to be used as a model building aid for non-linear mixed effects (population) analysis using NONMEM. It facilitates data set checkout, exploration and visualization, model diagnostics, candidate covariate identification and model comparison.

LazyLoad yes

LazyData yes

License LGPL (>= 3)

URL <http://xpose.sourceforge.net>

Copyright 2009-2014 Andrew C. Hooker, E. Niclas Jonsson and Mats O. Karlsson. 2005-2008 Andrew C. Hooker, Justin J. Wilkins, Mats O. Karlsson and E. Niclas Jonsson. 1998-2004 E. Niclas Jonsson and Mats Karlsson.

Author Andrew C. Hooker [aut, cre, cph],
Mats O. Karlsson [aut, cph],
Justin J. Wilkins [aut],
E. Niclas Jonsson [aut, trl, cph],
Ron Keizer [ctb] (functionality for bootstrap of GAM and SCM)

Maintainer Andrew C. Hooker <andrew.hooker@farmbio.uu.se>

NeedsCompilation no

Repository CRAN

Date/Publication 2014-11-24 12:05:52

R topics documented:

xpose4-package	5
xpose4classic-package	7
xpose4data-package	9
xpose4generic-package	11
xpose4psn-package	12
xpose4specific-package	13
absval.cwres.vs.cov.bw	17
absval.cwres.vs.pred	18
absval.cwres.vs.pred.by.cov	20
absval.delta.model.comp	22
absval.iwres.cwres.vs.ipred.pred	24
absval.iwres.vs.ipred	25
absval.iwres.vs.ipred.by.cov	27
absval.iwres.vs.pred	29
absval.wres.vs.cov.bw	30
absval.wres.vs.idv	32
absval.wres.vs.pred	34
absval.wres.vs.pred.by.cov	35
add.functions	37
add.grid.table	39
add.model.comp	41
addit.gof	42
autocorr.cwres	44
autocorr.wres	46
basic.gof	47
basic.model.comp	49
boot.hist	51
bootgam.print	52
bootscm.import	53
cat.dv.vs.idv.sb	54
cat.pc	56
change.graphical.parameters	58
change.misc.parameters	60
change.parm	63
change.var.name	64
change.xlabel	65
change.xvardef	66
classic	68
compute.cwres	68
create.mirror	77
create.xpose.plot.classes	78
createXposeClasses	78
cwres.dist.hist	79
cwres.dist.qq	80
cwres.vs.cov	81
cwres.vs.idv	82

cwres.vs.idv.bw	84
cwres.vs.pred	85
cwres.vs.pred.bw	87
cwres.wres.vs.idv	88
Data	90
data.checkout	92
db.names	93
dist.hist	94
dist.qq	96
dOFV.vs.cov	97
dOFV.vs.id	99
dOFV1.vs.dOFV2	101
dv.preds.vs.idv	102
dv.vs.idv	103
dv.vs.ipred	105
dv.vs.ipred.by.cov	106
dv.vs.ipred.by.idv	108
dv.vs.pred	109
dv.vs.pred.by.cov	110
dv.vs.pred.by.idv	112
dv.vs.pred.ipred	113
export.graph.par	115
export.variable.definitions	116
functions	117
gam.functions	118
generic	121
gof	122
import.graph.par	124
import.variable.definitions	126
ind.plots	127
ind.plots.wres.hist	130
ind.plots.wres.qq	133
internal.functions	135
ipred.vs.idv	136
iwres.dist.hist	138
iwres.dist.qq	139
iwres.vs.idv	140
kaplan.plot	141
make.sb.data	145
npc.coverage	146
nsim	148
parm.splom	149
parm.vs.cov	151
parm.vs.parm	153
pred.vs.idv	154
print.xpose.multiple.plot	155
randtest.hist	156
ranpar.vs.cov	157

Read NONMEM output files	158
read.nm.tables	160
read.npc.vpc.results	162
read.TTE.sim.data	163
read.vpctab	164
reset	165
runsum	166
simpraz.xpdb	168
simprazExample	171
summary	171
tables	173
tabulate.parameters	174
wres.dist.hist	175
wres.dist.qq	176
wres.vs.cov	177
wres.vs.idv	178
wres.vs.idv.bw	180
wres.vs.pred	181
wres.vs.pred.bw	182
xlabel	184
xp.boot.par.est	185
xp.boot.par.est.corr	186
xp.distr.mod.size	187
xp.dofv.plot	188
xp.inc.prob	189
xp.inc.prob.comb.2	189
xp.inc.stab.cov	190
xp.incl.index.cov	191
xp.incl.index.cov.comp	192
xp.incl.index.cov.ind	192
xpose.ask.for.filename	193
xpose.create.title	194
xpose.create.title.text	195
xpose.data	196
xpose.data-class	198
xpose.dev.new	200
xpose.gam	200
xpose.license.citation	203
xpose.multiple.plot	203
xpose.multiple.plot.default	205
xpose.panel.bw	206
xpose.panel.default	209
xpose.panel.histogram	222
xpose.panel.qq	225
xpose.panel.splom	226
xpose.plot.bw	228
xpose.plot.default	231
xpose.plot.histogram	236

xpose.plot.qq	239
xpose.plot.splom	242
xpose.prefs-class	245
xpose.string.print	247
xpose.VPC	247
xpose.VPC.both	253
xpose.VPC.categorical	255
xpose.yscale.components.log10	257
xsubset	258
xvardef	259

Index**261**

xpose4-package	<i>The Xpose Package</i>
----------------	--------------------------

Description

Xpose is an R-based model building aid for population analysis using NONMEM. It facilitates data set checkout, exploration and visualization, model diagnostics, candidate covariate identification and model comparison.

Details

Xpose is an R library for post-processing of NONMEM output. It takes one or more standard NONMEM table files as input and generates graphs or other analyses. It is assumed that each NONMEM run can be uniquely identified by a run number (see section below for how to generate the appropriate input to Xpose). Xpose is implemented using the lattice graphics library.

The Xpose library can be divided up into five sub-sections:

[xpose4data-package](#) Functions for managing the input data and manipulating the Xpose database.

[xpose4generic-package](#) Generic wrapper functions around the lattice functions. These functions can be invoked by the user but require quite detailed instructions to generate the desired output.

[xpose4specific-package](#) These functions are single purpose functions that generate specific output given only the Xpose database as input. The behavior can, to some extent, be influenced by the user.

[xpose4classic-package](#) Xpose has a text based menu interface to make it simple for the user to invoke the Xpose specific functions. This interface is called Xpose Classic. Given the limitations a text based interface imposes, Xpose Classic is not very flexible but may be useful for quick assessment of a model and for learning to use Xpose.

[xpose4psn-package](#) These functions are the interface between Xpose and PsN, i.e. they do not post-process NONMEM output but rather PsN output.

How to make NONMEM generate input to Xpose

Xpose recognizes NONMEM runs, and files associated to a particular run, though the run number. This is a number that is used in the name of NONMEM model files, output files and table files. The fundamental input to Xpose is one or more NONMEM table files. These table files should be named as below followed by the run number, for example xptab1 for run number 1. Xpose looks for files according to the following pattern, where * is your run number:

sdtab* Standard table file, containing ID, IDV, DV, PRED, IPRED, WRES, IWRES, RES, IRES, etc.

patab* Parameter table, containing model parameters - THETAs, ETAs and EPSes

catab* Categorical covariates, e.g. SEX, RACE

cotab* Continuous covariates, e.g. WT, AGE

extra*, **mutab***, **mytab***, **xptab***, **cwtab*** Other variables you might need to have available to Xpose

run*.mod Model specification file

run*.lst NONMEM output

Strictly, only one table file is needed for xpose (for example sdtab* or xptab*). However, using patab*, cotab*, catab* will influence the way that Xpose interprets the data and are recommended to get full benefit from Xpose.

You can use code in NONMEM similar to the following to generate the tables you need. NONMEM automatically appends DV, PRED, WRES and RES unless NOAPPEND is specified. Don't forget to leave at least one blank line at the end of the NONMEM model specification file.

```
$TABLE ID TIME IPRED IWRES EVID MDV NOPRINT ONEHEADER FILE=sdtab1 $TABLE ID CL V2 KA K SLP KENZ NOPRINT
$TABLE ID WT HT AGE BMI PKG NOPRINT ONEHEADER FILE=cotab1 $TABLE ID SEX SMOK ALC NOPRINT ONEHEADER FILE
```

Running Xpose

Xpose can be run in two different ways, using the classic menu system and by using the command line in R.

Running Xpose from the classic menu system:

Xpose can be run through a series of menu selections. This method of running Xpose is useful for the beginning "Xposer" and for learning about the functionality available in Xpose. The menu system does not allow for as much customization as the command line interface, nor does it allow for scripting.

To run the classic Xpose interface, at the R command prompt, type:

```
library(xpose4) xpose4()
```

And follow the menu prompts.

Running Xpose from the command line: Each function in Xpose is independently available from the command line, once the library is loaded. For example, you might do the following:

```
library(xpose4) xpdb <- xpose.data(5) basic.gof(xpdb)
```

Getting Help

All Xpose 4.0 functions are documented within the R help system. For example, if you want to know more about DV vs PRED plots and their options, type at the R command line after loading Xpose:

```
?dv.vs.pred
```

You can also use the interactive help system in most R-interfaces: choose Help from the menu system, then click on packages, then xpose4.

The most interesting may be the specific functions [xpose4specific-package](#). These are functions that are ready to use just by telling the functions which xpose database to use (i.e. xpdb after running the command `xpdb <- xpose.data(run.number)`).

The Xpose 4 Bestiary

A more detailed description of Xpose with example plots and explanations for every specific function in the package is available in our Bestiary: [http://xpose.sourceforge.net/bestiary_v1.0.pdf](http://xpose.sourceforge.net/bestiary/v1.0.pdf).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins and Andrew Hooker

References

[Xpose Homepage](#), [PsN Homepage](#)

Examples

```
## Not run:
# run the classic interface
library(xpose4)
xpose4()

# command line interface
library(xpose4)
xpdb <- xpose.data(5)
basic.gof(xpdb)

## End(Not run)
```

xpose4classic-package *Xpose classic menu system*

Description

Functions used to control the classic menu system in Xpose.

Details

Xpose has a text based menu interface to make it simple for the user to invoke the Xpose specific functions. This interface is called Xpose Classic. Given the limitations a text based interface imposes, Xpose Classic is not very flexible but may be useful for quick assessment of a model and for learning to use Xpose.

- `add.modify.db.items.menu`
- `another.menu`
- `change.dispersion`
- `change.graphical.par`
- `change.medianNorm`
- `change.onlyfirst`
- `change.steppit`
- `change.use.ids`
- `change.xp.obj`
- `covariate.model.menu`
- `cwres.menu`
- `data.checkout.menu`
- `documentation.menu`
- `gam.change.weights`
- `gam.menu`
- `gam.settings.menu`
- `gof.menu`
- `list.gam.settings`
- `main.menu`
- `manage.db`
- `model.comparison.covariates.menu`
- `model.comparison.menu`
- `parameters.menu`
- `preferences.menu`
- `residual.diagnostics.menu`
- `runsum.print`
- `struct.gof.menu`
- `structural.diagnostics.menu`
- `table.file.read.settings.menu`
- `vpc.npc.menu`
- `xpose4`

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins and Andrew Hooker

xpose4data-package *Xpose database creation and manipulation*

Description

Functions used to read in NONMEM table files to an Xpose database and to create and manipulate an Xpose database

Details

- [add.absval](#)
- [add.dichot](#)
- [add.exp](#)
- [add.log](#)
- [add.tad](#)
- [bootscm.import](#)
- [calc.npar](#)
- [change.ab.graph.par](#)
- [change.bw.graph.par](#)
- [change.cat.cont](#)
- [change.cat.levels](#)
- [change.cond.graph.par](#)
- [change.dil.graph.par](#)
- [change.dv.cat.levels](#)
- [change.label.par](#)
- [change.lm.graph.par](#)
- [change.misc.graph.par](#)
- [change.miss](#)
- [change.parm](#)
- [change.pi.graph.par](#)
- [change.smooth.graph.par](#)
- [change.subset](#)
- [change.var.name](#)
- [change.xlabel](#)
- [change.xvardef](#)
- [check.vars](#)
- [create.parameter.list](#)
- [createXposeClasses](#)

- Data
- db.names
- export.graph.par
- export.variable.definitions
- fix.wrapped.lines
- get.doc
- import.graph.par
- import.variable.definitions
- is.readable.file
- make.sb.data
- nsim
- nsim<-
- read.lst
- read.lst6
- read.lst7
- read.nm.tables
- read.npc.vpc.results
- read.phi
- read.TTE.sim.data
- read.vpctab
- reset.graph.par
- reset.variables
- SData
- set.doc
- simpraz.xpdb
- simprazExample
- test.xpose.data
- xlabel
- xpose.ask.for.filename
- xpose.ask.for.lst
- xpose.ask.for.mod
- xpose.bin
- xpose.data
- xpose.data-class
- xpose.prefs-class
- xpose.print
- xpose.read
- xpose.write
- xsubset
- xvardef

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins and Andrew Hooker

xpose4generic-package *Xpose generic multi-purpose functions*

Description

Generic xpose functions

Details

These functions can be invoked by the user but require quite detailed input to generate the desired output. used by [xpose4specific-package](#) to create output.

- [add.grid.table](#)
- [add.grid.text](#)
- [addid](#)
- [categorical.table](#)
- [compute.cwres](#)
- [computePI](#)
- [continuous.table](#)
- [create.mirror](#)
- [create.rand](#)
- [create.strat.rand](#)
- [create.xpose.plot.classes](#)
- [eq.xpose](#)
- [find.right.table](#)
- [get.polygon.regions](#)
- [get.refrunno](#)
- [ind.cwres](#)
- [is.cwres.readable.file](#)
- [print.xpose.multiple.plot](#)
- [read.cwres.data](#)
- [setup.PPI](#)
- [sqrtm](#)
- [xpose.calculate.cwres](#)
- [xpose.create.label](#)
- [xpose.create.title](#)

- `xpose.create.title.hist`
- `xpose.create.title.text`
- `xpose.dev.new`
- `xpose.logTicks`
- `xpose.multiple.plot`
- `xpose.multiple.plot.default`
- `xpose.multiple.plot.title`
- `xpose.panel.bw`
- `xpose.panel.default`
- `xpose.panel.histogram`
- `xpose.panel.qq`
- `xpose.panel.splom`
- `xpose.plot.bw`
- `xpose.plot.default`
- `xpose.plot.histogram`
- `xpose.plot.qq`
- `xpose.plot.splom`
- `xpose.stack`
- `xpose.string.print`
- `xpose.xscale.components.log10`
- `xpose.yscale.components.log10`

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins and Andrew Hooker

xpose4psn-package

Xpose PsN post-processing functions

Description

Xpose functions for post-processing PsN output.

Details

These functions are the interface between Xpose and PsN, i.e. they do not post-process NONMEM output but rather PsN output.

- bootstrap
 - `boot.hist`
- boot_scm
 - `bootscm.import`
 - `bootgam.print`
 - `xp.inc.prob`
 - `xp.inc.prob.comb.2`
 - `xp.distr.mod.size`
 - `xp.inc.stab.cov`
 - `xp.incl.index.cov`
 - `xp.incl.index.cov.ind`
 - `xp.incl.index.cov.comp`
 - `xp.boot.par.est`
 - `xp.boot.par.est.corr`
 - `xp.dofv.plot`
- randtest
 - `randtest.hist`
- vpc and npc
 - `cat.pc`
 - `kaplan.plot`
 - `npc.coverage`
 - `xpose.VPC`
 - `xpose.VPC.both`
 - `xpose.VPC.categorical`

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins and Andrew Hooker

xpose4specific-package

Xpose specific functions for a single purpose

Description

A list of specific functions for tasks to be performed using an Xpose database

Details

These functions are single purpose functions that generate specific output given only the Xpose database as input. The behavior can, to some extent, be influenced by the user.

- `absval.cwres.vs.cov.bw`
- `absval.cwres.vs.pred`
- `absval.cwres.vs.pred.by.cov`
- `absval.dcwres.vs.cov.model.comp`
- `absval.dipred.vs.cov.model.comp`
- `absval.diwres.vs.cov.model.comp`
- `absval.dpred.vs.cov.model.comp`
- `absval.dwres.vs.cov.model.comp`
- `absval.iwres.cwres.vs.ipred.pred`
- `absval.iwres.vs.cov.bw`
- `absval.iwres.vs.idv`
- `absval.iwres.vs.ipred`
- `absval.iwres.vs.ipred.by.cov`
- `absval.iwres.vs.pred`
- `absval.iwres.wres.vs.ipred.pred`
- `absval.wres.vs.cov.bw`
- `absval.wres.vs.idv`
- `absval.wres.vs.pred`
- `absval.wres.vs.pred.by.cov`
- `add.model.comp`
- `addit.gof`
- `autocorr.cwres`
- `autocorr.iwres`
- `autocorr.wres`
- `basic.gof`
- `basic.model.comp`
- `bootgam.print`
- `cat.dv.vs.idv.sb`
- `cat.pc`
- `check.gamobj`
- `cov.hist`
- `cov.qq`
- `cov.splom`
- `cov.summary`

- `cwres.dist.hist`
- `cwres.dist.qq`
- `cwres.vs.cov`
- `cwres.vs.idv`
- `cwres.vs.idv.bw`
- `cwres.vs.pred`
- `cwres.vs.pred.bw`
- `cwres.wres.vs.idv`
- `cwres.wres.vs.pred`
- `data.checkout`
- `d0FV.vs.cov`
- `d0FV.vs.id`
- `d0FV1.vs.d0FV2`
- `dv.preds.vs.idv`
- `dv.vs.idv`
- `dv.vs.ipred`
- `dv.vs.ipred.by.cov`
- `dv.vs.ipred.by.idv`
- `dv.vs.pred`
- `dv.vs.pred.by.cov`
- `dv.vs.pred.by.idv`
- `dv.vs.pred.ipred`
- `gof`
- `gofSetup`
- `ind.plots`
- `ind.plots.cwres.hist`
- `ind.plots.cwres.qq`
- `ind.plots.wres.hist`
- `ind.plots.wres.qq`
- `ipred.vs.idv`
- `iwres.dist.hist`
- `iwres.dist.qq`
- `iwres.vs.idv`
- `kaplan.plot`
- `parm.hist`
- `parm.qq`
- `parm.splom`

- `parm.summary`
- `parm.vs.cov`
- `parm.vs.parm`
- `pred.vs.idv`
- `ranpar.hist`
- `ranpar.qq`
- `ranpar.splom`
- `ranpar.vs.cov`
- `runsum`
- `tabulate.parameters`
- `wres.dist.hist`
- `wres.dist.qq`
- `wres.vs.cov`
- `wres.vs.idv`
- `wres.vs.idv.bw`
- `wres.vs.pred`
- `wres.vs.pred.bw`
- `xp.akaike.plot`
- `xp.boot.par.est`
- `xp.boot.par.est.corr`
- `xp.check.scope`
- `xp.cook`
- `xp.distr.mod.size`
- `xp.dofv.plot`
- `xp.gam`
- `xp.get.disp`
- `xp.inc.prob`
- `xp.inc.prob.comb.2`
- `xp.inc.stab.cov`
- `xp.incl.index.cov`
- `xp.incl.index.cov.comp`
- `xp.incl.index.cov.ind`
- `xp.ind.inf.fit`
- `xp.ind.inf.terms`
- `xp.ind.stud.res`
- `xp.plot`
- `xp.scope3`

- [xp.summary](#)
- [xpose.gam](#)
- [xpose.license.citation](#)
- [xpose.VPC](#)
- [xpose.VPC.both](#)
- [xpose.VPC.categorical](#)
- [xpPage](#)

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins and Andrew Hooker

absval.cwres.vs.cov.bw

Absolute conditional weighted residuals vs covariates for Xpose 4

Description

This creates a stack of box and whisker plot of absolute population conditional weighted residuals (ICWRES) vs covariates, and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the [codexpose.plot.bw](#) function. Most of the options take their default values from xpose.data object but may be overridden by supplying them as arguments.

Usage

```
absval.cwres.vs.cov.bw(object,
  xlb = "|CWRES|",
  main="Default",
  ...)
```

Arguments

object	An xpose.data object.
xlb	A string giving the label for the x-axis. NULL if none.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
...	Other arguments passed to xpose.plot.bw .

Details

Each of the covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$Covariates`, is evaluated in turn, creating a stack of plots.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling box-and-whisker plots are available. See [xpose.plot.bw](#) for details.

Value

Returns a stack of box-and-whisker plots of |CWRES| vs covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.bw](#), [xpose.panel.bw](#), [compute.cwres](#), [bwplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.cwres.vs.cov.bw(xpdb)

## A custom plot
absval.cwres.vs.cov.bw(xpdb, bwdotcol="white",
  bwdotpch=15,
  bwreccol="red",
  bwrecfill="red",
  bwumbcol="red",
  bwoutpch=5,
  bwoutcol="black")

## End(Not run)
```

`absval.cwres.vs.pred` *Absolute population conditional weighted residuals vs population predictions for Xpose 4*

Description

This is a plot of absolute population conditional weighted residuals (|CWRES|) vs population predictions (PRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
absval.cwres.vs.pred(object,
  idsdir="up",
  type="p",
  smooth=TRUE,
  ...)
```

Arguments

object	An xpose.data object.
idsdir	Direction for displaying point labels. The default is "up", since we are displaying absolute values.
type	Type of plot. The default is points ("p"), but lines ("l") and both ("b") are also available.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns an xyplot of $|CWRES|$ vs PRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
```

```

data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.cwres.vs.pred(xpdb)

## A conditioning plot
absval.cwres.vs.pred(xpdb, by="HCTZ")

## Custom heading and axis labels
absval.cwres.vs.pred(xpdb, main="My conditioning plot", ylb="|CWRES|", xlb="PRED")

## Custom colours and symbols, no IDs
absval.cwres.vs.pred(xpdb, cex=0.6, pch=3, col=1, ids=FALSE)

```

```
absval.cwres.vs.pred.by.cov
```

Absolute value of the conditional weighted residuals vs. population predictions, conditioned on covariates, for Xpose 4

Description

This is a plot of absolute population conditional weighted residuals (|CWRES|) vs population predictions (PRED) conditioned by covariates, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```

absval.cwres.vs.pred.by.cov(object,
  ylb = "|CWRES|",
  type="p",
  smooth=TRUE,
  idsdir="up",
  main="Default",
  ...)

```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>ylb</code>	A string giving the label for the y-axis. NULL if none.
<code>idsdir</code>	Direction for displaying point labels. The default is "up", since we are displaying absolute values.
<code>type</code>	Type of plot. The default is points only ("p"), but lines ("l") and both ("b") are also available.

smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Each of the covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$Covariates`, is evaluated in turn, creating a stack of plots.

The main argument is not supported owing to the multiple plots generated by the function.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See `compute.cwres` for details.

A wide array of extra options controlling xyplots are available. See `xpose.plot.default` for details.

Value

Returns a stack of xyplots of |CWRES| vs PRED, conditioned on covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[absval.cwres.vs.pred](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.cwres.vs.pred.by.cov(xpdb)

## Custom axis labels
absval.cwres.vs.pred.by.cov(xpdb, ylb="|CWRES|", xlb="PRED")

## Custom colours and symbols, no IDs
absval.cwres.vs.pred.by.cov(xpdb, cex=0.6, pch=3, col=1, ids=FALSE)
```

```
## End(Not run)
```

```
absval.delta.model.comp
```

Model comparison plots, of absolute differences in goodness-of-fit predictors against covariates, for Xpose 4

Description

These functions plot absolute differences in PRED, IPRED, WRES, CWRES and IWRES against covariates for the two specified model fits.

Usage

```
absval.dpred.vs.cov.model.comp(object,
  object.ref = NULL,
  type = NULL,
  ylb=expression(paste("|", Delta, "PRED|")),
  main="Default",
  ...)
```

```
absval.dipred.vs.cov.model.comp(object,
  object.ref = NULL,
  type = NULL,
  ylb=expression(paste("|", Delta, "IPRED|")),
  main="Default",
  ...)
```

```
absval.dcwres.vs.cov.model.comp(object,
  object.ref = NULL,
  type = NULL,
  ylb=expression(paste("|", Delta, "CWRES|")),
  main="Default",
  ...)
```

```
absval.dwres.vs.cov.model.comp(object,
  object.ref = NULL,
  type = NULL,
  ylb=expression(paste("|", Delta, "WRES|")),
  main="Default",
  ...)
```

```
absval.diwres.vs.cov.model.comp(object,
  object.ref = NULL,
  type = NULL,
  ylb=expression(paste("|", Delta, "IWRES|")),
  main="Default",
  ...)
```

Arguments

object	An xpose.data object.
object.ref	An xpose.data object. If not supplied, the user will be prompted.
type	1-character string giving the type of plot desired. The following values are possible, for details, see 'plot': "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c") for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
ylb	A string giving the label for the y-axis. NULL if none.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

These functions produce plots of the absolute difference in population predictions (`absval.dpred.vs.cov.model.comp`), individual predictions (`absval.dipred.vs.cov.model.comp`), weighted population residuals (`absval.dwres.vs.cov.model.comp`), conditional weighted population residuals (`absval.dcwres.vs.cov.model.comp`) and individual predictions (`absval.diwres.vs.cov.model.comp`).

Conditional weighted residuals (CWRES) require some extra steps to calculate. See `compute.cwres` for details.

A wide array of extra options controlling xyplots are available. See `xpose.plot.default` for details.

Value

Returns a stack of plots comprising comparisons of PRED, IPRED, WRES (or CWRES) and IWRES for the two specified runs.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

`xpose.plot.default`, `xpose.panel.default`, `xyplot`, `compute.cwres`, `xpose.prefs-class`, `xpose.data-class`

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for runs
## 5 and 6 in the current working directory
xpdb5 <- xpose.data(5)
xpdb6 <- xpose.data(6)
```

```
## A basic dWRES plot, without prompts
absval.dwres.vs.cov.model.comp(xpdb5, xpdb6)

## Custom colours and symbols, no user IDs
absval.dpred.vs.cov.model.comp(xpdb5, xpdb6, cex=0.6, pch=8, col=1, ids=NULL)

## End(Not run)
```

```
absval.iwres.cwres.vs.ipred.pred
```

Absolute population weighted residuals vs population predictions, and absolute individual weighted residuals vs individual predictions, for Xpose 4

Description

This is a matrix plot of absolute population weighted residuals (ICWRESI) vs population predictions (PRED) and absolute individual weighted residuals (IIWRESI) vs individual predictions (IPRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `absval.cwres.vs.pred` and `absval.iwres.vs.ipred` functions.

Usage

```
absval.iwres.wres.vs.ipred.pred(object,
                               main="Default",
                               ...)
absval.iwres.cwres.vs.ipred.pred(object,
                                 main="Default",
                                 ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>main</code>	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

The plots created by the `absval.wres.vs.pred` and `absval.iwres.vs.ipred` functions are presented side by side for comparison.

A wide array of extra options controlling xyplots are available. See `xpose.plot.default` for details.

Value

Returns a compound plot.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[absval.wres.vs.pred](#), [absval.iwres.vs.ipred](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.iwres.wres.vs.ipred.pred(xpdb)
absval.iwres.cwres.vs.ipred.pred(xpdb)

## Custom colours and symbols
absval.iwres.cwres.vs.ipred.pred(xpdb, cex=0.6, pch=8, col=1)
```

`absval.iwres.vs.ipred` *Absolute individual weighted residuals vs individual predictions for Xpose 4*

Description

This is a plot of absolute individual weighted residuals (IWRES) vs individual predictions (IPRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
absval.iwres.vs.ipred(object,
  ylb = "|IWRES|",
  type="p",
  ids = FALSE,
  idsdir = "up",
  smooth = TRUE,
  ...)
```

Arguments

object	An xpose.data object.
ylb	A string giving the label for the y-axis. NULL if none.
idsdir	Direction for displaying point labels. The default is "up", since we are displaying absolute values.
type	Type of plot. The default is points only ("p"), but lines ("l") and both ("b") are also available.
ids	Should id values be displayed?
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns an xyplot of |IWRES| vs IPRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#), [runsum](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)
```

```
## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.iwres.vs.ipred(xpdb)

## A conditioning plot
absval.iwres.vs.ipred(xpdb, by="HCTZ")

## Custom heading and axis labels
absval.iwres.vs.ipred(xpdb, main="My conditioning plot", ylb="|IWRES|", xlb="IPRED")

## Custom colours and symbols, no IDs
absval.iwres.vs.ipred(xpdb, cex=0.6, pch=3, col=1, ids=FALSE)
```

```
absval.iwres.vs.ipred.by.cov
```

Absolute individual weighted residuals vs individual predictions, conditioned on covariates, for Xpose 4

Description

This is a plot of absolute individual weighted residuals (IWRES) vs individual predictions (IPRED) conditioned by covariates, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
absval.iwres.vs.ipred.by.cov(object,
  ylb = "|IWRES|",
  idsdir="up",
  type="p",
  smooth=TRUE,
  main="Default",
  ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>ylb</code>	A string giving the label for the y-axis. NULL if none.
<code>idsdir</code>	Direction for displaying point labels. The default is "up", since we are displaying absolute values.
<code>type</code>	Type of plot. The default is points only ("p"), but lines ("l") and both ("b") are also available.

smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Each of the covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$Covariates`, is evaluated in turn, creating a stack of plots.

A wide array of extra options controlling xyplots are available. See `xpose.plot.default` for details.

Value

Returns a stack of xyplots of |IWRES| vs IPRED, conditioned by covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[absval.iwres.vs.ipred](#), [xpose.plot.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.iwres.vs.ipred.by.cov(xpdb)

## Custom axis labels
absval.iwres.vs.ipred.by.cov(xpdb, ylb="|IWRES|", xlb="IPRED")

## Custom colours and symbols, no IDs
absval.iwres.vs.ipred.by.cov(xpdb, cex=0.6, pch=3, col=1, ids=FALSE)

## End(Not run)
```

absval.iwres.vs.pred *Absolute individual weighted residuals vs population predictions or independent variable for Xpose 4*

Description

This is a plot of absolute individual weighted residuals (IIWRESI) vs individual predictions (PRED) or independent variable (IDV), specific functions in Xpose 4. These functions are wrappers encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
absval.iwres.vs.pred(object,
  ylb = "|IWRES|",
  smooth = TRUE,
  idsdir = "up",
  type = "p",
  ...)
```

```
absval.iwres.vs.idv(object,
  ylb = "|iWRES|",
  smooth = TRUE,
  idsdir = "up",
  type = "p",
  ...)
```

Arguments

object	An <code>xpose.data</code> object.
ylb	A string giving the label for the y-axis. NULL if none.
idsdir	Direction for displaying point labels. The default is "up", since we are displaying absolute values.
type	Type of plot. The default is points only ("p"), but lines ("l") and both ("b") are also available.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns an xyplot of |IWRES| vs PRED or |IWRES| vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.iwres.vs.pred(xpdb)

## A conditioning plot
absval.iwres.vs.pred(xpdb, by="HCTZ")

## Custom heading and axis labels
absval.iwres.vs.pred(xpdb, main="My conditioning plot", ylb="|IWRES|", xlb="PRED")

## Custom colours and symbols, no IDs
absval.iwres.vs.pred(xpdb, cex=0.6, pch=3, col=1, ids=FALSE)
```

`absval.wres.vs.cov.bw` *Absolute weighted residuals vs covariates for Xpose 4*

Description

This creates a stack of box and whisker plot of absolute population weighted residuals (|WRES| or |IWRES|) vs covariates. It is a wrapper encapsulating arguments to the `xpose.plot.bw` function. Most of the options take their default values from the `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
absval.wres.vs.cov.bw(object,
  xlb = "|WRES|",
  main="Default",
```

```

    ... )

absval.iwres.vs.cov.bw(object,
  xlb = "|iWRES|",
  main="Default",
  ... )

```

Arguments

object	An xpose.data object.
xlb	A string giving the label for the x-axis. NULL if none.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
...	Other arguments passed to xpose.plot.bw .

Details

Each of the covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$Covariates`, is evaluated in turn, creating a stack of plots.

A wide array of extra options controlling box-and-whisker plots are available. See [xpose.plot.bw](#) for details.

Value

Returns a stack of box-and-whisker plots of |WRES| vs covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.bw](#), [xpose.panel.bw](#), [bwplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```

## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.wres.vs.cov.bw(xpdb)

## A custom plot

```

```
absval.wres.vs.cov.bw(xpdb, bwdotcol="white",
  bwdotpch=15,
  bwreccol="red",
  bwrecfill="red",
  bwumbcol="red",
  bwoutpch=5,
  bwoutcol="black")
```

```
## A vanilla plot using IWRES
absval.iwres.vs.cov.bw(xpdb)
```

```
## End(Not run)
```

absval.wres.vs.idv *Absolute value of (C)WRES vs. independent variable plot in Xpose4.*

Description

This is a plot of the absolute value of the CWRES (default, other residuals as an option) vs independent variable, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the [xpose.plot.default](#) function. Most of the options take their default values from the `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
absval.wres.vs.idv(object,
  idv = "idv",
  wres = "Default",
  ylb = "Default",
  smooth = TRUE,
  idsdir = "up",
  type = "p",
  ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>idv</code>	the independent variable.
<code>wres</code>	Which weighted residual to use. "Default" is the CWRES.
<code>ylb</code>	Y-axis label.
<code>smooth</code>	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
<code>idsdir</code>	Direction for displaying point labels. The default is "up", since we are displaying absolute values.
<code>type</code>	Type of plot. The default is points only ("p"), but lines ("l") and both ("b") are also available.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns an xyplot of |CWRES| vs idv (often TIME, defined by [xvardef](#)).

Author(s)

Andrew Hooker

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.wres.vs.idv(xpdb)

## A conditioning plot
absval.wres.vs.idv(xpdb, by="HCTZ")

## Custom heading and axis labels
absval.wres.vs.idv(xpdb, main="Hello World", ylb="|CWRES|", xlb="IDV")

## Custom colours and symbols
absval.wres.vs.idv(xpdb, cex=0.6, pch=3, col=1)

## using the NPDEs instead of CWRES
absval.wres.vs.idv(xpdb, wres="NPDE")

## subsets
absval.wres.vs.idv(xpdb, subset="TIME<10")
```

absval.wres.vs.pred *Absolute population weighted residuals vs population predictions for Xpose 4*

Description

This is a plot of absolute population weighted residuals (IWRESI) vs population predictions (PRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
absval.wres.vs.pred(object,
  ylb = "|WRES|",
  idsdir="up",
  type="p",
  smooth=TRUE,
  ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>ylb</code>	A string giving the label for the y-axis. NULL if none.
<code>idsdir</code>	Direction for displaying point labels. The default is "up", since we are displaying absolute values.
<code>type</code>	Type of plot. The default is points only ("p"), but lines ("l") and both ("b") are also available.
<code>smooth</code>	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns an xyplot of IWRESI vs PRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.wres.vs.pred(xpdb)

## A conditioning plot
absval.wres.vs.pred(xpdb, by="HCTZ")

## Custom heading and axis labels
absval.wres.vs.pred(xpdb, main="My conditioning plot",
  ylb="|WRES|", xlb="PRED")

## Custom colours and symbols
absval.wres.vs.pred(xpdb, cex=0.6, pch=19, col=1,
  smcol="blue", smlty=2)
```

```
absval.wres.vs.pred.by.cov
```

*Absolute population weighted residuals vs population predictions,
conditioned on covariates, for Xpose 4*

Description

This is a plot of absolute population weighted residuals (|WRES|) vs population predictions (PRED) conditioned by covariates, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
absval.wres.vs.pred.by.cov(object,
  ylb = "|WRES|",
  type="p",
  smooth=TRUE,
  ids = FALSE,
```

```
idsdir="up",
main="Default",
...)
```

Arguments

object	An xpose.data object.
ylb	A string giving the label for the y-axis. NULL if none.
ids	Logical. Should id labels on points be shown?
idsdir	Direction for displaying point labels. The default is "up", since we are displaying absolute values.
type	Type of plot. The default is points only ("p"), but lines ("l") and both ("b") are also available.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Each of the covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$Covariates`, is evaluated in turn, creating a stack of plots.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns a stack of xyplots of |WRES| vs PRED, conditioned on covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[absval.wres.vs.pred](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
```

```
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
absval.wres.vs.pred.by.cov(xpdb)

## Custom axis labels
absval.wres.vs.pred.by.cov(xpdb, ylb="|CWRES|", xlb="PRED")

## Custom colours and symbols, IDs
absval.wres.vs.pred.by.cov(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)

## End(Not run)
```

add.functions

Column-transformation functions for Xpose 4

Description

These functions transform existing Xpose 4 data columns, adding new columns.

Usage

```
add.absval(object,
            listall = TRUE,
            classic = FALSE)

add.dichot(object,
            listall = TRUE,
            classic = FALSE)

add.exp(object,
         listall = TRUE,
         classic = FALSE)

add.log(object,
         listall = TRUE,
         classic = FALSE)

add.tad(object,
         classic = FALSE)
```

Arguments

object An xpose.data object.

listall A logical operator specifying whether the items in the database should be listed.

classic A logical operator specifying whether the function should assume the classic menu system. This is an internal option and need never be called from the command line.

Details

These functions may be used to create new data columns within the Xpose data object by transforming existing ones. `add.absval` creates a column reflecting the absolute value of a column, `add.dichot` creates a categorical data column based upon a continuous variable, `add.exp` creates an exponentiated version of an existing variable, `add.log` provides log transformation, and `add.tad` creates a time-after-dose (TAD) data item based upon the dose and time variables in the dataset.

Value

An `xpose.data` object (`classic == FALSE`) or null (`classic == TRUE`).

Author(s)

Niclas Jonsson & Justin Wilkins

See Also

[xpose.data](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Create a column containing the absolute values of data in another
## column
add.absval(xpdb5)

## Create a categorical data column based on a continuous data column,
## and do not list variables
add.dichot(xpdb5, listall = FALSE)

## Create a column containing the exponentiated values of data in
## another column
add.exp(xpdb5)

## Create a column containing log-transformations of data in another
## column
add.log(xpdb5)

## Create a time-after-dose column
add.tad(xpdb5)

## End(Not run)
```

add.grid.table	<i>Print tables or text in a grid object</i>
----------------	--

Description

These functions take an array of values and labels or an array of text and add it to one or many grid viewports in an orderly fashion.

Usage

```
add.grid.table(txt,  
               col.nams = NULL,  
               ystart,  
               xstart = unit(0, "npc"),  
               start.pt = 1,  
               vp,  
               vp.num = 1,  
               minrow = 5,  
               cell.padding = 0.5,  
               mult.col.padding = 1,  
               col.optimize = TRUE,  
               equal.widths = FALSE,  
               space.before.table = 1,  
               center.table = FALSE,  
               use.rect=FALSE,  
               fill.type=NULL, # all,top,side,both,NULL  
               fill.col="grey",  
               cell.lines.lty=0,  
               ...)
```

```
add.grid.text(txt,  
              ystart,  
              xstart = unit(0, "npc"),  
              start.pt = 1,  
              vp,  
              vp.num = 1,  
              spaces.before = NULL,  
              spaces.after = NULL,  
              v.space.before=0,  
              v.space.after=0,  
              use.rect=FALSE,  
              wdth=NULL,  
              fill.type=NULL,  
              fill.col="grey",  
              cell.lines.lty=0,  
              xpose.table=gTree(),  
              ...)
```

Arguments

txt	The text or table values to add to the grid object.
col.nams	the column names of the table values
ystart	The y location to start printing in the grid viewport
xstart	The x location to start printing in the grid viewport
start.pt	The start point (row) in the table array to start printing
vp	The viewport(s) to add the table or text to
vp.num	the viewport number in vp to start printing to
minrow	The minimum rows before printing more columns to use in the table
cell.padding	padding between cells in the table
mult.col.padding	padding between multiple columns in the table
col.optimize	should we column optimize (TRUE) or row optimize (FALSE)
equal.widths	Should all columns have equal widths
space.before.table	Should there be a space before the table
center.table	should we center the table in the viewport?
use.rect	Should we make rectangles with background color around the table entries TRUE or FALSE
fill.type	Which rectangles should be filled. Allowed values are "all", "top", "side", "both" and NULL.
fill.col	The color of the filled rectangles
cell.lines.lty	The line-type for the lines between the cells, using the same values as lty.
spaces.before	Rows added before text is printed out
spaces.after	Rows added after text is printed out
v.space.before	Vertical space above text in a table in "lines" units.
v.space.after	Vertical space below text in a table in "lines" units.
width	The width of the column of data
xpose.table	A grob object.
...	Other arguments passed to the various functions.

Value

A List is returned with the following components

ystart	new starting point for new text
stop.pt	null if everything gets printed
vp.num	the viewport needed for next text printed
xpose.table	A grob object that can be plotted.

Author(s)

Andrew Hooker

See Also[runsum](#), [grid.text](#)

add.model.comp

*Additional model comparison plots, for Xpose 4***Description**

This creates a stack of four plots, comparing absolute values of PRED, absolute values of IPRED, delta CWRES (or WRES) and delta IWRES estimates for the two specified model fits.

Usage

```
add.model.comp(object,
               object.ref = NULL,
               onlyfirst = FALSE,
               inclZeroWRES = FALSE,
               subset = xsubset(object),
               main="Default",
               force.wres=FALSE,
               ...)
```

Arguments

object	An xpose.data object.
object.ref	An xpose.data object. If not supplied, the user will be prompted.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot. The default is TRUE.
onlyfirst	Logical value indicating whether only the first row per individual is included in the plot.
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
force.wres	Should we use the WRES in the plots instead of CWRES (logical TRUE or FALSE)
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Four model comparison plots are displayed in sequence.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns a stack of plots comprising comparisons of absolute values of PRED, absolute values of IPRED, absolute differences in CWRES (or WRES) and absolute differences in IWRES for the two specified runs.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [compute.cwres](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for runs
## 5 and 6 in the current working directory
xpdb5 <- xpose.data(5)
xpdb6 <- xpose.data(6)

## A vanilla plot, without prompts
add.model.comp(xpdb5, xpdb6, prompt = FALSE)

## Custom colours and symbols, no user IDs
add.model.comp(xpdb5, xpdb6, cex=0.6, pch=8, col=1, ids=NULL)

## End(Not run)
```

Description

This is a compound plot consisting of plots of weighted population residuals (WRES) vs population predictions (PRED), absolute individual weighted residuals (IWRES) vs independent variable (IDV), WRES vs IDV, and weighted population residuals vs log(IDV), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `wres.vs.pred`, `iwres.vs.idv` and `wres.vs.idv` functions.

Usage

```
addit.gof(object,
          type="p",
          title.size=0.02,
          title.just=c("center", "top"),
          main="Default",
          force.wres=FALSE,
          ...)
```

Arguments

object	An xpose.data object.
type	1-character string giving the type of plot desired. The following values are possible, for details, see 'plot': "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c") for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
title.size	Amount, in a range of 0-1, of how much space the title should take up in the plot)
title.just	how the title should be justified
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
force.wres	Plot the WRES even if other residuals are available.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Four additional goodness-of-fit plots are presented side by side for comparison.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.multiple.plot.default](#) for details.

Value

Returns a compound plot comprising plots of weighted population residuals (WRES) vs population predictions (PRED), absolute individual weighted residuals (IWRES) vs independent variable (IDV), WRES vs IDV, and weighted population residuals vs log(IDV).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[wres.vs.pred](#), [iwres.vs.idv](#), [wres.vs.idv](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
addit.gof(xpdb)

## Custom colours and symbols
addit.gof(xpdb, cex=0.3, pch=0, col=8)
```

autocorr.cwres

Autocorrelation of conditional weighted residuals for Xpose 4

Description

This is an autocorrelation plot of conditional weighted residuals, a specific function in Xpose 4. Most of the options take their default values from xpose.data object but may be overridden by supplying them as arguments.

Usage

```
autocorr.cwres(object,
               type="p",
               smooth=TRUE,
               ids=F,
               main = "Default",
               ...)
```

Arguments

object	An xpose.data object.
smooth	Logical value indicating whether a smooth should be superimposed.
type	1-character string giving the type of plot desired. The following values are possible, for details, see plot : "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c") for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.

ids	A logical value indicating whether text labels should be used as plotting symbols (the variable used for these symbols indicated by the idlab xpose data variable).
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See `xpose.plot.default` for details.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See `compute.cwres` for details.

Value

Returns an autocorrelation plot for conditional weighted population residuals (CWRES).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

`xyplot`, `xpose.prefs-class`, `compute.cwres`, `xpose.data-class`

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
autocorr.cwres(xpdb)

## A conditioning plot
autocorr.cwres(xpdb, dilution=TRUE)

## Custom heading and axis labels
autocorr.cwres(xpdb, main="My conditioning plot", ylb="|CWRES|", xlb="PRED")

## Custom colours and symbols, IDs
autocorr.cwres(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

autocorr.wres

Autocorrelation of weighted residuals for Xpose 4

Description

This is an autocorrelation plot of weighted residuals. Most of the options take their default values from the `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
autocorr.wres(object,
              type="p",
              smooth=TRUE,
              ids=F,
              main = "Default",
              ...)
```

```
autocorr.iwres(object,
               type="p",
               smooth=TRUE,
               ids=F,
               main = "Default",
               ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>smooth</code>	Logical value indicating whether a smooth should be superimposed.
<code>type</code>	1-character string giving the type of plot desired. The following values are possible, for details, see plot : "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c") for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
<code>ids</code>	A logical value indicating whether text labels should be used as plotting symbols (the variable used for these symbols indicated by the <code>idlab</code> <code>xpose</code> data variable).
<code>main</code>	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns an autocorrelation plot for weighted population residuals (WRES) or individual weighted residuals (IWRES).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins & Andrew Hooker

See Also

[xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
autocorr.wres(xpdb)

## A conditioning plot
autocorr.wres(xpdb, dilution=TRUE)

## Custom heading and axis labels
autocorr.wres(xpdb, main="My conditioning plot", ylb="|CWRES|", xlb="PRED")

## Custom colours and symbols, IDs
autocorr.wres(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)

## A vanilla plot with IWRES
autocorr.iwres(xpdb)
```

Description

This is a compound plot consisting of plots of observations (DV) vs population predictions (PRED), observations (DV) vs individual predictions (IPRED), absolute individual weighted residuals (IIWRES) vs IPRED, and weighted population residuals (CWRES) vs independent variable (IDV), a specific function in Xpose 4. WRES are also supported. It is a wrapper encapsulating arguments to the `dv.vs.pred`, `dv.vs.ipred`, `absval.iwres.vs.ipred` and `wres.vs.idv` functions.

Usage

```
basic.gof(object,  
          force.wres=FALSE,  
          main="Default",  
          use.log = FALSE,  
          ...)
```

Arguments

object	An xpose.data object.
force.wres	Should the plots use WRES? Values can be TRUE/FALSE. Otherwise the CWRES are used if present.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
use.log	Should we use log transformations in the plots?
...	Other arguments passed to xpose.plot.default .

Details

Four basic goodness-of-fit plots are presented side by side for comparison.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

`basic.gof.cwres` is just a wrapper for `basic.gof` with `use.cwres=TRUE`.

Value

Returns a compound plot comprising plots of observations (DV) vs population predictions (PRED), DV vs individual predictions (IPRED), absolute individual weighted residuals (IIWRES) vs IPRED, and weighted populations residuals (WRES) vs the independent variable (IDV).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[dv.vs.pred](#), [dv.vs.ipred](#), [absval.iwres.vs.ipred](#), [wres.vs.idv](#), [cwres.vs.idv](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [compute.cwres](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run 1
## in the current working directory
xpdb <- xpose.data(1)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
basic.gof(xpdb)

## Custom colours and symbols, IDs of individuals in study
basic.gof(xpdb, cex=0.6, pch=8, col=1, ids=TRUE)
```

basic.model.comp

Basic model comparison plots, for Xpose 4

Description

This creates a stack of four plots, comparing PRED, IPRED, WRES (or CWRES), and IWRES estimates for the two specified model fits.

Usage

```
basic.model.comp(object,
  object.ref = NULL,
  onlyfirst = FALSE,
  inclZeroWRES = FALSE,
  subset = xsubset(object),
  main="Default",
  force.wres=FALSE,
  ...)
```

Arguments

object	An xpose.data object.
object.ref	An xpose.data object. If not supplied, the user will be prompted.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot. The default is TRUE.
onlyfirst	Logical value indicating whether only the first row per individual is included in the plot.

subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
force.wres	Force function to use WRES?
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Four basic model comparison plots are displayed in sequence.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.wres](#) for details.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) for details.

Value

Returns a stack of plots comprising comparisons of PRED, IPRED, WRES (or CWRES) and IWRES for the two specified runs.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [compute.wres](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for runs
## 5 and 6 in the current working directory
xpdb5 <- xpose.data(5)
xpdb6 <- xpose.data(6)

## A vanilla plot, without prompts
basic.model.comp(xpdb5, xpdb6, prompt = FALSE)

## Custom colours and symbols, no user IDs
basic.model.comp.wres(xpdb5, xpdb6, cex=0.6, pch=8, col=1, ids=NULL)

## End(Not run)
```

boot.hist	<i>Function to create histograms of results from the bootstrap tool in PsN</i>
-----------	--

Description

Reads results from the bootstrap tool in [PsN](http://psn.sf.net) and then creates histograms.

Usage

```
boot.hist(results.file = "raw_results_run1.csv",
          incl.ids.file = "included_individuals1.csv",
          min.failed = FALSE, cov.failed = FALSE,
          cov.warnings = FALSE, boundary = FALSE,
          showOriginal = TRUE, showMean = FALSE,
          showMedian = FALSE, showPCTS = FALSE,
          PCTS = c(0.025, 0.975), excl.id = c(),
          layout = NULL, sort.plots = TRUE,
          main = "Default", ...)
```

Arguments

results.file	The location of the results file from the bootstrap tool in PsN
incl.ids.file	The location of the included ids file from the bootstrap tool in PsN
min.failed	Should NONMEM runs that had failed minimization be included? TRUE or FALSE
cov.failed	Should NONMEM runs that had a failed covariance step be included? TRUE or FALSE
cov.warnings	Should NONMEM runs that had covariance step warnings be included? TRUE or FALSE
boundary	Should NONMEM runs that had boundary warnings be included? TRUE or FALSE
showOriginal	Should we show the value from the original NONMEM run in the histograms? TRUE or FALSE
showMean	Should we show the mean of the histogram data? TRUE or FALSE
showMedian	Should we show the median of the histogram data? TRUE or FALSE
showPCTS	Should we show the percentiles of the histogram data? TRUE or FALSE
PCTS	the percentiles to show. Can be a vector of any length. For example, <code>c(0.05, 0.2, 0.5, 0.7)</code>
excl.id	Vector of id numbers to exclude.
layout	Layout of plots. A vector of number of rows and columns in each plot. <code>c(3, 3)</code> for example.
sort.plots	Should the plots be sorted based on type of parameter. Sorting on parameters, standard errors, shrinkage and eigenvalues.

main The title of the plot.
 ... Additional arguments that can be passed to [xpose.plot.histogram](#), [xpose.panel.histogram](#), [histogram](#) and other [lattice-package](#) functions.

Value

Plots or a list of plots are returned.

Author(s)

Andrew Hooker

References

[PsN](#)

See Also

[xpose.plot.histogram](#), [xpose.panel.histogram](#), [histogram](#) and other [lattice-package](#) functions.

Examples

```
## Not run:
boot.hist(results.file="./boot1/raw_results_run1.csv",
          incl.ids.file="./boot1/included_individuals1.csv")

## End(Not run)
```

bootgam.print

Print summary information for a bootgam or bootscm

Description

This functions prints some summary information for a bootgam performed in Xpose, or for a bootscm performed in PsN.

Usage

```
bootgam.print(bootgam.obj = NULL)
```

Arguments

bootgam.obj The bootgam or bootscm object.

Value

No value returned

Author(s)

Ron Keizer

Examples

```
## Not run:
bootgam.print(current.bootgam) # Print summary for the current Xpose bootgam object
bootgam.print(current.bootscm) # Print summary for the current Xpose bootscm object

## End(Not run)
```

```
bootscm.import      Import bootscm data into R/Xpose
```

Description

This function imports data generated by the PsN `boot_scm` function into the Xpose / R environment.

Usage

```
bootscm.import(scm.folder = NULL, silent = FALSE, n.bs = NULL,
cov.recoding = NULL, group.by.cov=NULL, skip.par.est.import=FALSE,
dofv.forward = 3.84, dofv.backward = 6.64)
```

Arguments

<code>scm.folder</code>	The folder in which the PsN-generated bootscm data are.
<code>silent</code>	Don't output any progress report. Default is FALSE.
<code>n.bs</code>	The number of bootstraps performed. Defaults to 100.
<code>cov.recoding</code>	For categorical covariates that are recoded to dichotomous covariates within the bootscm configuration file, a list can be specified containing data frames for recoding. See the example below for details.
<code>group.by.cov</code>	Group inclusion frequencies by covariate, instead of calculating them per parameter-covariates relationship. Default is NULL, which means that the user will be asked to make a choice.
<code>skip.par.est.import</code>	Skip the import of all parameter estimates (in each final model in all scm's, as well as parameter estimates in first step of each scm). These data are required to make plot that show inclusion bias and correlation in parameter estimates. Importing these data takes a bit of time (may take a minute or so), so if you don't intend to make these plots anyhow this step can be skipped. Default is FALSE.
<code>dofv.forward</code>	dOFV value used in forward step of scm.
<code>dofv.backward</code>	dOFV value used in backward step of scm.

Author(s)

Ron Keizer

cat.dv.vs.idv.sb	<i>Categorical observations vs. independent variable using stacked bars.</i>
------------------	--

Description

Categorical observations vs. independent variable using stacked bars.

Usage

```
cat.dv.vs.idv.sb(object,
  dv=xvardef("dv",object),
  idv=xvardef("idv",object),
  by=NULL,
  groups=dv,
  force.by.factor = FALSE,
  recur=F,
  xlb=idv,
  ylb="Proportion",
  subset=NULL,
  vary.width=T,
  level.to.plot=NULL,
  refactor.levels=TRUE,
  main=xpose.create.title.text(idv,dv,
    "Proportions of",object,subset=subset,...),
  stack=TRUE,
  horizontal=FALSE,

  strip = function(...)
  strip.default(...,strip.names=c(TRUE,TRUE)),
  scales      = list(),
  inclZeroWRES = TRUE,
  onlyfirst   = FALSE,
  samp        = NULL,
  aspect      = object@Prefs@Graph.prefs$aspect,
  auto.key    = "Default",

  ## mirror stuff
  mirror      = FALSE,
  mirror.aspect="fill",
  pass.plot.list=FALSE,
  x.cex=NULL,
  y.cex=NULL,
  main.cex=NULL,
  mirror.internal=list(strip.missing=missing(strip)),
  ...)
```

Arguments

<code>object</code>	Xpose data object.
<code>dv</code>	The dependent variable (e.g. "DV" or "CP".)
<code>idv</code>	The indenpent variable (e.g. "TIME".)
<code>by</code>	Conditioning variable
<code>groups</code>	How we should group values in each conditional plot.
<code>force.by.factor</code>	Should we force the data to be treated as factors?
<code>recur</code>	Not used.
<code>xlb</code>	A string giving the label for the x-axis. NULL if none.
<code>ylb</code>	A string giving the label for the y-axis. NULL if none.
<code>subset</code>	Subset of data.
<code>vary.width</code>	Should we vary the width of the bars to match amount of information?
<code>level.to.plot</code>	Which levels of the DV to plot.
<code>refactor.levels</code>	Should we refactor the levels?
<code>main</code>	The title of the plot.
<code>stack</code>	Should we stack the bars?
<code>horizontal</code>	Should the bars be horizontal?
<code>strip</code>	Defining how the strips should appear in the conditioning plots.
<code>scales</code>	Scales argument to xyplot .
<code>inclZeroWRES</code>	Include rows with WRES=0?
<code>onlyfirst</code>	Only include first data point for each individual?
<code>samp</code>	Sample to use in mirror plot (a number).
<code>aspect</code>	Aspect argument to xyplot .
<code>auto.key</code>	Make a legend.
<code>mirror</code>	Mirror can be FALSE, TRUE, 1 or 3.
<code>mirror.aspect</code>	Aspect for mirror.
<code>pass.plot.list</code>	Should the plot list be passsed back to user?
<code>x.cex</code>	Size of x axis label.
<code>y.cex</code>	Size of Y axis label.
<code>main.cex</code>	Size of Title.
<code>mirror.internal</code>	Internal stuff.
<code>...</code>	Other arguments passed to function.

Author(s)

Andrew Hooker

Examples

```
## Not run:
## read in table files
runno <- 45
xpdb <- xpose.data(runno)

## make some stacked bar plots
cat.dv.vs.idv.sb(xpdb, idv=NULL, stack=F)
cat.dv.vs.idv.sb(xpdb, idv=NULL, stack=F, by="DOSE")
cat.dv.vs.idv.sb(xpdb, idv="DOSE")
cat.dv.vs.idv.sb(xpdb, idv=NULL, stack=F, by="TIME")
cat.dv.vs.idv.sb(xpdb, idv="TIME")
cat.dv.vs.idv.sb(xpdb, idv="CAVH")
cat.dv.vs.idv.sb(xpdb, idv="TIME", by="DOSE", scales=list(x=list(rot=45)))

## make some mirror plots
cat.dv.vs.idv.sb(xpdb, idv="DOSE", mirror=1)
cat.dv.vs.idv.sb(xpdb, idv="CAVH", mirror=1, auto.key=F)

## End(Not run)
```

cat.pc

Categorical (visual) predictive check.

Description

Categorical (visual) predictive check plots.

Usage

```
cat.pc(object,
       dv=xvardef("dv", object),
       idv=xvardef("idv", object),
       level.to.plot=NULL,
       subset=NULL,
       histo=T,
       median.line=F,
       PI.lines=F,
       xlb=if(histo){
         paste("Proportion of ", dv)
       } else {
         paste(idv)
       },
       ylb=if(histo){
         paste("Percent of Total")
       } else {
         paste("Proportion of Total")
       },
```



```
main=xpose.create.title.text(NULL,dv,  
  "Predictive check of",object,subset=subset,...),  
strip="Default",  
...)
```

Arguments

object	Xpose data object.
dv	The dependent variable (e.g. "DV" or "CP".)
idv	The indenpent variable (e.g. "TIME".)
level.to.plot	The levels to plot.
subset	Subset of data.
histo	If FALSE then a VPC is created, given that idv is defined.
median.line	Make a median line?
PI.lines	Make prediction interval lines?
xlb	Label for x axis.
ylb	label for y axis.
main	Main title.
strip	Defining how the strips should appear in the conditioning plots.
...	Extra arguments passed to the function.

Author(s)

Andrew C. Hooker

Examples

```
## Not run:  
## read in table files  
runno <- 45  
xpdb <- xpose.data(runno)  
  
## create proportion (visual) predictive check  
cat.pc(xpdb,idv=NULL)  
cat.pc(xpdb,idv="DOSE")  
cat.pc(xpdb,idv="DOSE",histo=F)  
cat.pc(xpdb,idv="TIME",histo=T,level.to.plot=1)  
  
## End(Not run)
```

`change.graphical.parameters`*Functions changing variable definitions in Xpose 4*

Description

These functions allow customization of Xpose's graphics settings.

Usage

```
change.ab.graph.par(object,  
                    classic = FALSE)
```

```
change.bw.graph.par(object,  
                    classic = FALSE)
```

```
change.cond.graph.par(object,  
                      classic = FALSE)
```

```
change.dil.graph.par(object,  
                    classic = FALSE)
```

```
change.label.par(object,  
                classic = FALSE)
```

```
change.lm.graph.par(object,  
                   classic = FALSE)
```

```
change.misc.graph.par(object,  
                     classic = FALSE)
```

```
change.pi.graph.par(object,  
                   classic = FALSE)
```

```
change.smooth.graph.par(object,  
                       classic = FALSE)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>classic</code>	A logical operator specifying whether the function should assume the classic menu system. This is an internal option and need never be called from the command line.

Details

These functions are used to customize graphics settings, the way Xpose draws its graphs. `change.misc.graph.par` sets basic graphics parameters, including plot type, point type and size, colour, line type, and line width. `change.ab.graph.par` is used to change settings for the line of identity, `change.lm.graph.par` is responsible for linear regression lines, `change.smooth.graph.par` sets preferences for loess smooths, `change.bw.graph.par` sets preferences for box-and-whisker plots, and `change.label.par`, `change.dil.graph.par` and `change.pi.graph.par` functions are responsible for labelling, dilution and prediction interval plotting preferences, respectively. `change.cond.graph.par` sets preferences for conditioning.

Settings can be saved and loaded using `export.graph.par` and `import.graph.par`, respectively.

Value

An `xpose.data` object (`classic == FALSE`) or null (`classic == TRUE`).

Author(s)

Niclas Jonsson & Justin Wilkins

See Also

`xpose.plot.default`, `xpose.panel.default`, `xpose.plot.bw`, `xpose.panel.bw`, `xpose.plot.default`, `import.graph.p`, `export.graph.par`, `plot.default`, `par`, `import.graph.par`, `panel.abline`, `panel.lmline`, `lm`, `panel.loess`, `loess.smooth`, `loess`, `panel.bwplot`, `shingle`, `reorder.factor`

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Change default miscellaneous graphic preferences
xpdb5 <- change.misc.graph.par(xpdb5)

## Change default linear regression line preferences, creating a new
## object
xpdb5.a <- change.lm.graph.par(xpdb5)

## Change conditioning preferences
xpdb5 <- change.cond.graph.par(xpdb5)

## End(Not run)
```

`change.misc.parameters`*Functions changing miscellaneous parameter settings in Xpose 4*

Description

These functions allow viewing and changing of settings relating to subsets, categorical threshold values, documentation and numbers indicating missing data values.

Usage

```
change.cat.levels(object,  
                  classic = FALSE,  
                  cat.limit=NULL,  
                  ...)
```

```
change.cat.levels(object,  
                  classic = FALSE,  
                  ...) <- value
```

```
change.dv.cat.levels(object,  
                     classic = FALSE,  
                     dv.cat.limit=NULL,  
                     ...)
```

```
change.dv.cat.levels(object,  
                     classic = FALSE,  
                     ...) <- value
```

```
change.cat.cont(object,  
                listall=TRUE,  
                classic=FALSE,  
                to.cat.vec=NULL,  
                to.cont.vec=NULL,  
                change.type.vec=NULL,  
                ...)
```

```
change.cat.cont(object,  
                listall=TRUE,  
                classic=FALSE,  
                to.cat.vec=NULL,  
                to.cont.vec=NULL,  
                ...) <- value
```

```
change.miss(object,
```

```

        classic = FALSE)

change.subset(object,
              classic = FALSE)

set.doc(object,
         classic = FALSE)

get.doc(object,
         classic = FALSE)

```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>classic</code>	A logical operator specifying whether the function should assume the classic menu system. This is an internal option and need never be called from the command line.
<code>dv.cat.limit</code>	The limit for which we treat DV as categorical. If there are <code>dv.cat.limit</code> or less unique <code>dv</code> values then <code>dv</code> is treated as categorical.
<code>cat.limit</code>	The limit for which we treat a list of values as categorical. If there are <code>cat.limit</code> or less unique values then the list is treated as categorical.
<code>listall</code>	A logical operator specifying whether the items in the database should be listed.
<code>to.cat.vec</code>	A vector of strings specifying the names of the categorical variables that should be transformed to continuous.
<code>to.cont.vec</code>	A vector of strings specifying the names of the continuous variables that should be transformed to categorical.
<code>change.type.vec</code>	A vector of strings specifying the names of the variables that should be transformed to/from continuous/categorical.
<code>...</code>	arguments passed to other functions.
<code>value</code>	This is the value that will be replaced in the <code>xpose</code> data object <code>object</code> . <code>value</code> is used in the “replacement function” version of these functions. That is the form where we have <code>function.name(object) <- value</code> . If <code>value</code> is <code>NULL</code> then the functions prompt the user for a value. For <code>change.cat.levels</code> , <code>value</code> is the categorical limit <code>cat.limit</code> . For <code>change.dv.cat.levels</code> , <code>value</code> is the DV categorical limit <code>dv.cat.limit</code> . For <code>change.cat.cont</code> , <code>value</code> is the <code>change.type.vec</code> . See the examples below.

Details

These functions are used to change settings for the number of unique data values required in a variable in order to define it as continuous (`change.cat.levels` and `change.dv.cat.levels` for ordinary variables and the dependent variable, respectively), the value to use as ‘missing’ (`change.miss`), and viewing (`get.doc`) and setting (`set.doc`) the documentation field in the `Xpose` data object.

change.cat.cont allows interchange between categorical and continuous data formats within the Xpose database. This in turn affects how plots are drawn.

change.subset is used for setting the data item's subset field. To specify a subset of the data to process, you use the variable names and the regular R selection operators. To combine a subset over two or more variables, the selection expressions for the two variables are combined using R's unary logical operators.

The variable names are those that are specified in the NONMEM table files (e.g. PRED, TIME, SEX).

The selection operators are: == (equal) != (not equal) || (or) > (greater than) < (less than)

For example, to specify that TIME less than 24 should be processed, you type the expression: TIME < 24.

The unary logical operators are: & (and) | (or)

For example, to specify TIME less than 24 and males (SEX equal to 1), you type the expression: TIME < 24 & SEX == 1

This subset selection scheme works on all variables, including ID numbers.

The subset selection is not entirely stable. For example, there is no check that the user enters a valid expression, nor that the user specifies existing variable names. An erroneous expression will not become evident until a plot is attempted and the expression takes effect.

Value

An [xpose.data](#) object, except view.doc, which returns the value of object@Doc.

Author(s)

Andrew Hooker, Niclas Jonsson & Justin Wilkins

See Also

[Data](#), [SData](#), [subset](#), [xpose.data](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Change default subset
xpdb5 <- change.subset(xpdb5)

## Set documentation field
xpdb5 <- set.doc(xpdb5)
## View it
view.doc(xpdb5)

## change the categorical limit for the dv variable
```

```
change.dv.cat.levels(xpdb5) <- 10

## change the categorical limit for non DV variables
change.cat.levels(xpdb5) <- 2
## or
xpdb5 <- change.cat.levels(xpdb5,cat.levels=2)

## chnage variables from categorical to continuous
xpdb5 <- change.cat.cont(xpdb5,to.cat.vec=c("AGE"),to.cont.vec=c("SEX"))
xpdb5 <- change.cat.cont(xpdb5,change.type.vec=c("AGE","SEX"))
change.cat.cont(xpdb5) <- c("AGE","SEX")

## End(Not run)
```

change.parm	<i>Change parameter scope.</i>
-------------	--------------------------------

Description

Function to change the parameter scope.

Usage

```
change.parm(object, listall = TRUE, classic = FALSE)
```

Arguments

object	The xpose data object.
listall	whether we should list all the current parameters.
classic	true if used in the classic menu system (for internal use).

Value

If classic then return nothing. Otherwise return the new data object.

Author(s)

Andrew C. Hooker

change.var.name	<i>Changes the name of an Xpose data item</i>
-----------------	---

Description

This function allows the names of data items in the Xpose database to be changed.

Usage

```
change.var.name(object,  
                classic = FALSE)
```

Arguments

object	An xpose.data object.
classic	A logical operator specifying whether the function should assume the classic menu system. This is an internal option and need never be called from the command line.

Details

This function facilitates the changing of data item names in the object@Data slot.

Value

An [xpose.data](#) object.

Author(s)

Niclas Jonsson & Justin Wilkins

See Also

[Data](#), [SData](#), [xpose.data](#)

Examples

```
## Not run:  
## xpdb5 is an Xpose data object  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
xpdb5 <- change.var.name(xpdb5)  
  
## End(Not run)
```

change.xlabel	<i>Changes the label of an Xpose data item</i>
---------------	--

Description

This function allows the labels of data items in the Xpose database to be changed.

Usage

```
change.xlabel(object,  
              listall = TRUE,  
              classic = FALSE)
```

Arguments

object	An xpose.data object.
listall	A logical operator specifying whether the items in the database should be listed.
classic	A logical operator specifying whether the function should assume the classic menu system. This is an internal option and need never be called from the command line.

Details

This function facilitates the changing of data item labels in the object@Prefs@Labels slot.

Value

An [xpose.data](#) object.

Author(s)

Justin Wilkins

See Also

[Data,SData,xpose.data](#)

Examples

```
## Not run:  
## xpdb5 is an Xpose data object  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
xpdb5 <- change.xlabel(xpdb5)  
  
## End(Not run)
```

change.xvardef *Change Xpose variable definitions.*

Description

These functions allow for the changing of Xpose variable definitions like "idv" and "dv". These variable definitions are used to refer to columns of the observed data in a generic way, so that generic plotting functions can be created.

Usage

```
change.xvardef(object,var = ".ask", def = ".ask", listall = TRUE,
               classic = FALSE, check.var = FALSE, ...)
```

```
change.xvardef(object,
               var,
               listall=FALSE,
               classic=FALSE,
               check.var=FALSE,
               ...) <- value
```

Arguments

object	An xpose.data object.
var	The Xpose variable you would like to change or add to the current object. A one-element character vector (e.g. "idv"). If ".ask" the user will be prompted to input a value.
def	A vector of column names from NONMEM table files (names(object@Data)) that should be associated with this variable (e.g. c("TIME")). Mutiple values are allowed. If ".ask" the user will be prompted to input values.
value	Same as def defined above.
listall	Should the function list the database values?
classic	Is the function being used from the classic interface. This is an internal option.
check.var	Should the variables be checked against the current variables in the object?
...	Items passed to functions within this function.

Value

If called from the the command line then this function returns an xpose database. If called from the classic interface this function updates the current xpose database (.cur.db).

Additional arguments

The default xpose variables are:

id Individual identifier column in dataset

idlab values used for plotting ID values on data points in plots

occ The occasion variable

dv The dv variable

pred The pred variable

ipred The ipred variable

wres The wres variable

cwres The cwres variable

res The res variable

parms The parameters in the database

covariates The covariates in the database

ranpar The random parameters in the database

Acknowledgements

Thanks to Sebastien Bihorel for input which helped create this function.

Author(s)

Andrew Hooker

See Also

[xvardef](#), [xpose.data](#)

Examples

```
## Not run:
# Change the "id" variable to point to "PRED" in the xpose object
xpdb <- change.xvardef(xpdb,var="id",def="PRED")

# Check the value of the "id" variable
xvardef("id",xpdb)

# Change the "idv" variable
change.xvardef(xpdb,var="idv") <- "TIME"

# Change the covariate scope
change.xvardef(xpdb,var="covariates") <- c("SEX","AGE","WT")

# Use the interactive capabilities of the function
xpdb <- change.xvardef(xpdb)

## End(Not run)
```

`classic`*Classic menu functions for Xpose 4*

Description

These functions drive the classic Xpose menu system.

Usage`xpose4()`**Details**

The classic Xpose menu system is invoked by using the command `xpose4()`. With the exception of `xpose4`, these are internal Xpose functions, not intended for direct use.

Value

Functions for the classic Xpose menu system.

Author(s)

Andrew Hooker

Examples

```
## Not run:  
xpose4()  
  
## End(Not run)
```

`compute.cwres`*Compute the Conditional Weighted Residuals*

Description

This function computes the conditional weighted residuals (CWRES) from a NONMEM run. CWRES are an extension of the weighted residuals (WRES), but are calculated based on the first-order with conditional estimation (FOCE) method of linearizing a pharmacometric model (WRES are calculated based on the first-order (FO) method). The function requires a NONMEM table file and an extra output file that must be explicitly asked for when running NONMEM, see details below.

Usage

```
compute.cwres(run.number,
              tab.prefix="cwtab",
              sim.suffix="",
              est.tab.suffix=".est",
              deriv.tab.suffix=".deriv",
              old.file.convention=FALSE,
              id="ALL",
              printToOutfile=TRUE,
              onlyNonZero=TRUE,
              ...)

xpose.calculate.cwres(object,
                      cwres.table.prefix = "cwtab",
                      tab.suffix = "",
                      sim.suffix = "sim",
                      est.tab.suffix=".est",
                      deriv.tab.suffix=".deriv",
                      old.file.convention=FALSE,
                      id = "ALL",
                      printToOutfile = TRUE,
                      onlyNonZero = FALSE,
                      classic = FALSE,
                      ...)
```

Arguments

run.number	The run number of the NONMEM from which the CWRES are to be calculated.
tab.prefix	The prefix to two NONMEM file containing the needed values for the computation of the CWRES, described in the details section.
sim.suffix	The suffix ,before the ".", of the NONMEM file containing the needed values for the computation of the CWRES, described in the details section. For example, the table files might be named cwtab1sim.est and cwtab1sim.deriv, in which case sim.suffix="sim".
est.tab.suffix	The suffix, after the ".", of the NONMEM file containing the estimated parameter values needed for the CWRES calculation.
deriv.tab.suffix	The suffix, after the ".", of the NONMEM file containing the derivatives of the model with respect to the random parameters needed for the CWRES calculation.
old.file.convention	For backwards compatibility. Use this if you are using the previous file convention for CWRES (table files named cwtab1, cwtab1.50, cwtab1.51, ... , cwtab.58 for example).
id	Can be either "ALL" or a number matching an ID label in the datasetname. Value is fixed to "ALL" for xpose.calculate.cwres.

<code>printToOutfile</code>	Logical (TRUE/FALSE) indicating whether the CWRES values calculated should be appended to a copy of the <code>datasetname</code> . Only works if <code>id="ALL"</code> . If chosen the resulting output file will be <code>datasetname.cwres</code> . Value is fixed to TRUE for <code>xpose.calculate.cwres</code> .
<code>onlyNonZero</code>	Logical (TRUE/FALSE) indicating if the return value (the CWRES values) of <code>compute.cwres</code> should include the zero values associated with non-measurement lines in a NONMEM data file.
<code>object</code>	An <code>xpose.data</code> object.
<code>cwres.table.prefix</code>	The prefix to the NONMEM table file containing the derivative of the model with respect to the <code>etas</code> and <code>epsilons</code> , described in the details section.
<code>tab.suffix</code>	The suffix to the NONMEM table file containing the derivative of the model with respect to the <code>etas</code> and <code>epsilons</code> , described in the details section.
<code>classic</code>	Indicates if the function is to be used in the classic menu system.
<code>...</code>	Other arguments passed to basic functions in code.

Details

`compute.cwres` This function is the computational 'brains' of the CWRES computation and does not require an Xpose data object to work. The function simply reads in the following two files:

```
paste(tab.prefix,run.number,sim.suffix,est.tab.suffix,sep="")
paste(tab.prefix,run.number,sim.suffix,deriv.tab.suffix,sep="")
```

Which might be for example:

```
cwtab1.est
cwtab1.deriv
```

and (depending on the input values to the function) returns the CWRES in vector form as well as creating a new table file named:

```
paste(tab.prefix,run.number,sim.suffix,sep="")
```

Which might be for example:

```
cwtab1
```

`xpose.calculate.cwres` This function is a wrapper around the function `compute.cwres`. It computes the CWRES for the model file associated with the Xpose data object input to the function. If possible it also computes the CWRES for any simulated data associated with the current Xpose data object. If you have problems with this function try using `compute.cwres` and then rereading your dataset into Xpose.

Value

compute.cwres Returns a vector containing the values of the CWRES.

xpose.calculate.cwres
Returns an Xpose data object that contains the CWRES. If simulated data is present, then the CWRES will also be calculated for that data.

Setting up the NONMEM model file

In order for this function to calculate the CWRES, NONMEM must be run while requesting certain tables and files to be created. How these files are created differs depending on if you are using \$PRED or ADVAN as well as the version of NONMEM you are using. These procedures are known to work for NONMEM VI but may be different for NONMEM V. We have attempted to indicate where NONMEM V may be different, but this has not been extensively tested!

This procedure can be done automatically using Perl Speaks NONMEM (PsN) and we highly recommend using PsN for this purpose. After installing PsN just type 'execute [modelName] -cwres'. See <http://psn.sourceforge.net> for more details. PsN does not currently do this computation for NONMEM 7.

There are five main insertions needed in your NONMEM control file:

1. \$ABB COMRES=X.

Insert this line directly after your \$DATA line. The value of X is the number of ETA() terms plus the number of EPS() terms in your model. For example for a model with three ETA() terms and two EPS() terms the code would look like this:

```
$DATA temp.csv IGNORE=@
$ABB COMRES=5
$INPUT ID TIME DV MDV AMT EVID
$SUB ADVAN2 TRANS2
```

2. Verbatim code.

- Using ADVAN.

If you are using ADVAN routines in your model, then Verbatim code should be inserted directly after the \$ERROR section of your model file. The length of the code depends again on the number of ETA() terms and EPS() terms in your model. For each ETA(y) in your model there is a corresponding term G(y,1) that you must assign to a COM() variable. For each EPS(y) in your model, there is a corresponding HH(y,1) term that you must assign to a COM() variable.

For example for a model using ADVAN routines with three ETA() terms and two EPS() terms the code would look like this:

```
"LAST
" COM(1)=G(1,1)
" COM(2)=G(2,1)
" COM(3)=G(3,1)
" COM(4)=HH(1,1)
" COM(5)=HH(2,1)
```

- Using PRED.

If you are using \$PRED, the verbatim code should be inserted directly after the \$PRED section of your model file. For each ETA(y) in your model there is a corresponding term G(y,1) that you must assign to a COM() variable. For each EPS(y) in your model, there is a corresponding H(y,1) term that you must assign to a COM() variable. The code would look like this for three ETA() terms and two EPS() terms:

```
"LAST
" COM(1)=G(1,1)
" COM(2)=G(2,1)
" COM(3)=G(3,1)
" COM(4)=H(1,1)
" COM(5)=H(2,1)
```

3. INFN routine.

- Using ADVAN with NONMEM VI and higher.

If you are using ADVAN routines in your model, then an \$INFN section should be placed directly after the \$PK section using the following code. In this example we are assuming that the model file is named something like 'run1.mod', thus the prefix to these file names 'cwtab' has the same run number attached to it (i.e. 'cwtab1'). This should be changed for each new run number.

```
$INFN
IF (ICALL.EQ.3) THEN
  OPEN(50,FILE='cwtab1.est')
  WRITE(50,*) 'ETAS'
  DO WHILE(DATA)
    IF (NEWIND.LE.1) WRITE (50,*) ETA
  ENDDO
  WRITE(50,*) 'THETAS'
  WRITE(50,*) THETA
  WRITE(50,*) 'OMEGAS'
  WRITE(50,*) OMEGA(BLOCK)
  WRITE(50,*) 'SIGMAS'
  WRITE(50,*) SIGMA(BLOCK)
ENDIF
```

- Using ADVAN with NONMEM V.

If you are using ADVAN routines in your model, then you need to use an INFN subroutine. If we call the INFN subroutine 'myinfn.for' then the \$SUBS line of your model file should include the INFN option. That is, if we are using ADVAN2 and TRANS2 in our model file then the \$SUBS line would look like:

```
$SUB ADVAN2 TRANS2 INFN=myinfn.for
```

The 'myinfn.for' routine for 4 thetas, 3 etas and 1 epsilon is shown below. If your model has different numbers of thetas, etas and epsilons then the values of NTH, NETA, and NEPS, should be changed respectively. These values are found in the DATA statement of the subroutine. additionally, in this example we are assuming that the model file is named something like 'run1.mod', thus the prefix to the output file names ('cwtab') in this

subroutine has the same run number attached to it (i.e. 'cwtab1'). This number should be changed for each new run number (see the line beginning with 'OPEN'). Please note that the 4th and 5th lines of code should be one line with the '...' removed from each line, reading: COMMON /ROCM6/ THETAF(40), OMEGAF(30, 30), SIGMAF(30, 30).

```

SUBROUTINE INFN(ICALL, THETA, DATREC, INDXS, NEWIND)
DIMENSION THETA(*), DATREC(*), INDXS(*)
DOUBLE PRECISION THETA
COMMON /ROCM6/ ...
... THETAF(40), OMEGAF(30, 30), SIGMAF(30, 30)
COMMON /ROCM7/ SETH(40), SEOM(30, 30), SESIG(30, 30)
COMMON /ROCM8/ OBJECT
COMMON /ROCM9/ IERE, IERC
DOUBLE PRECISION THETAF, OMEGAF, SIGMAF
DOUBLE PRECISION OBJECT
REAL SETH, SEOM, SESIG
DOUBLE PRECISION ETA(10)
INTEGER J, I
INTEGER IERE, IERC
INTEGER MODE
INTEGER NTH, NETA, NEPS
DATA NTH, NETA, NEPS/4, 3, 1/
IF (ICALL.EQ.0) THEN
C   open files here, if necessary
    OPEN(50, FILE='cwtab1.est')
ENDIF
IF (ICALL.EQ.3) THEN
    MODE=0
    CALL PASS(MODE)
    MODE=1
WRITE(50, *) 'ETAS'
20   CALL PASS(MODE)
    IF (MODE.EQ.0) GO TO 30
    IF (NEWIND.NE.2) THEN
        CALL GETETA(ETA)
        WRITE (50, 97) (ETA(I), I=1, NETA)
    ENDIF
    GO TO 20
30   CONTINUE
    WRITE (50, *) 'THETAS'
    WRITE (50, 99) (THETAF(J), J=1, NTH)
WRITE(50, *) 'OMEGAS'
    DO 7000 I=1, NETA
7000   WRITE (50, 99) (OMEGAF(I, J), J=1, NETA)
WRITE(50, *) 'SIGMAS'
    DO 7999 I=1, NEPS
7999   WRITE (50, 99) (SIGMAF(I, J), J=1, NEPS)
ENDIF
99   FORMAT (20E15.7)

```

```

98  FORMAT (2I8)
97  FORMAT (10E15.7)
    RETURN
    END

```

- Using \$PRED with NONMEM VI and higher.

If you are using \$PRED, then the following code should be placed at the end of the \$PRED section of the model file (together with the verbatim code). In this example we are assuming that the model file is named something like 'run1.mod', thus the prefix to these file names 'cwtab' has the same run number attached to it (i.e. 'cwtab1'). This should be changed for each new run number.

```

IF (ICALL.EQ.3) THEN
  OPEN(50,FILE='cwtab1.est')
  WRITE(50,*) 'ETAS'
  DO WHILE(DATA)
    IF (NEWIND.LE.1) WRITE (50,*) ETA
  ENDDO
  WRITE(50,*) 'THETAS'
  WRITE(50,*) THETA
  WRITE(50,*) 'OMEGAS'
  WRITE(50,*) OMEGA(BLOCK)
  WRITE(50,*) 'SIGMAS'
  WRITE(50,*) SIGMA(BLOCK)
ENDIF

```

- Using \$PRED with NONMEM V.

If you are using \$PRED with NONMEM V, then you need to add verbatim code immediately after the \$PRED command. In this example we assume 4 thetas, 3 etas and 1 epsilon. If your model has different numbers of thetas, etas and epsilons then the values of NTH, NETA, and NEPS, should be changed respectively. These values are found in the DATA statement below. Please note that the 3rd and 4th lines of code should be one line with the '...' removed from each line, reading: \"

```

$PRED
"FIRST
"      COMMON /ROCM6/ ...
... THETA(40), OMEGA(30,30), SIGMA(30,30)
"      COMMON /ROCM7/ SETH(40), SEOM(30,30), SESIG(30,30)
"      COMMON /ROCM8/ OBJECT
"      DOUBLE PRECISION THETA, OMEGA, SIGMA
"      DOUBLE PRECISION OBJECT
"      REAL SETH, SEOM, SESIG
"      INTEGER J, I
"      INTEGER MODE
"      INTEGER NTH, NETA, NEPS
"      DATA NTH, NETA, NEPS/4, 3, 1/

```

After this verbatim code you add all of the abbreviated code needed for the \$PRED routine in your model file. After the abbreviated code more verbatim code is needed. This verbatim code should be added before the verbatim code discussed above under point 2. In the example below we are assuming that the model file is named something like 'run1.mod', thus the prefix to the output file names ('cwtab') has the same run number attached to it (i.e. 'cwtab1'). This number should be changed for each new run number (see the line beginning with 'OPEN').

```

"      IF (ICALL.EQ.0) THEN
"C      open files here, if necessary
"      OPEN(50,FILE='cwtab1.est')
"      ENDIF
"      IF (ICALL.EQ.3) THEN
"      MODE=0
"      CALL PASS(MODE)
"      MODE=1
"      WRITE(50,*) 'ETAS'
" 20    CALL PASS(MODE)
"      IF (MODE.EQ.0) GO TO 30
"      IF (NEWIND.NE.2) THEN
"      CALL GETETA(ETA)
"      WRITE (50,97) (ETA(I),I=1,NETA)
"      ENDIF
"      GO TO 20
" 30    CONTINUE
"      WRITE (50,*) 'THETAS'
"      WRITE (50,99) (THETAF(J),J=1,NTH)
"      WRITE (50,*) 'OMEGAS'
"      DO 7000 I=1,NETA
" 7000  WRITE (50,99) (OMEGAF(I,J),J=1,NETA)
"      WRITE (50,*) 'SIGMAS'
"      DO 7999 I=1,NEPS
" 7999  WRITE (50,99) (SIGMAF(I,J),J=1,NEPS)
"      ENDIF
" 99   FORMAT (20E15.7)
" 98   FORMAT (2I8)
" 97   FORMAT (10E15.7)

```

4. cwtab*.deriv table file.

A special table file needs to be created to print out the values contained in the COMRES variables. In addition the ID, IPRED, MDV, DV, PRED and RES data items are needed for the computation of the CWRES. The following code should be added to the NONMEM model file. In this example we continue to assume that we are using a model with three ETA() terms and two EPS() terms, extra terms should be added for new ETA() and EPS() terms in the model file. We also assume the model file is named something like 'run1.mod', thus the prefix to these file names 'cwtab' has the same run number attached to it (i.e. 'cwtab1'). This should be changed for each new run number.

```
$TABLE ID COM(1)=G11 COM(2)=G21 COM(3)=G31
```

```
COM(4)=H11 COM(5)=H21
IPRED MDV NOPRINT ONEHEADER FILE=cwtab1.deriv
```

5. \$ESTIMATION.

To compute the CWRES, the NONMEM model file must use (at least) the FO method with the POSTHOC step. If the FO method is used and the POSTHOC step is not included then the CWRES values will be equivalent to the WRES. The CWRES calculations are based on the FOCE approximation, and consequently give an idea of the ability of the FOCE method to fit the model to the data. If you are using another method of parameter estimation (e.g. FOCE with interaction), the CWRES will not be calculated based on the same model linearization procedure.

Author(s)

Andrew Hooker

References

Hooker A, Staats CE, Karlsson MO. *Conditional weighted residuals, an improved model diagnostic for the FO/FOCE methods*. PAGE 15 (2006) Abstr 1001 [<http://www.page-meeting.org/?abstract=1001>].

Examples

```
## Not run:
## Capture CWRES from cwtab5.est and cwtab5.deriv
cwres <- compute.cwres(5)
mean(cwres)
var(cwres)

## Capture CWRES from cwtab1.est and cwtab1.deriv, do not print out, allow zeroes
cwres <- compute.cwres("1", printToFile = FALSE,
  onlyNonZero = FALSE)

## Capture CWRES for ID==1
cwres.1 <- compute.cwres("1", id=1)

## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Compare WRES, CWRES
xpdb5 <- xpose.calculate.cwres(xpdb5)
cwres.wres.vs.idv(xpdb5)

## End(Not run)
```

create.mirror	<i>Function to create mirror plots from the generic Xpose plotting commands</i>
---------------	---

Description

This function takes the generic plotting functions from Xpose 4 and calls them multiple times with the current arguments to the functions, changing the arguments as needed for mirror plotting.

Usage

```
create.mirror(fun,  
              arg.list,  
              mirror,  
              plotTitle,  
              fix.y.limits = TRUE,  
              fix.x.limits = TRUE,  
              ...)
```

Arguments

fun	The function name that we will call multiple times
arg.list	The arguments to that function
mirror	The type of mirror plots desired (1 or 3 mirror plots can be created)
plotTitle	The title for the plots
fix.y.limits	Should we fix all the y axes to be the same?
fix.x.limits	Should we fix all the x axes to be the same?
...	additional arguments passed to the function.

Details

mostly and internal function for Xpose

Value

a list of plots, or NULL.

Author(s)

Andrew Hooker

See Also

[xpose.plot.default](#), [xpose.plot.histogram](#), [xpose.plot.qq](#), [xpose.plot.splom](#)

```
create.xpose.plot.classes
```

Create xpose.multiple.plot class.

Description

Creates a class for viewing and plotting xpose plots with multiple plots on the same page or multiple pages.

Usage

```
create.xpose.plot.classes()
```

Author(s)

Niclas Jonsson and Andrew C. Hooker

```
createXposeClasses
```

This function creates the Xpose data classes ("xpose.data" and "xpose.prefs")

Description

This function defines and sets the Xpose data classes.

Usage

```
createXposeClasses(nm7=F)
```

Arguments

nm7 FALSE if not using NONMEM 7.

Note

All the default settings are defined in this function.

Author(s)

Niclas Jonsson and Andrew C. Hooker

See Also

[xpose.data-class](#), [xpose.prefs-class](#)

cwres.dist.hist	<i>Histogram of conditional weighted residuals (CWRES), for Xpose 4</i>
-----------------	---

Description

This is a histogram of the distribution of conditional weighted residuals (CWRES) in the dataset, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.histogram` function.

Usage

```
cwres.dist.hist(object,  
                ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>...</code>	Other arguments passed to <code>xpose.plot.histogram</code> .

Details

Displays a histogram of the conditional weighted residuals (CWRES).

Value

Returns a histogram of conditional weighted residuals (CWRES).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.histogram](#), [xpose.panel.histogram](#), [histogram](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
## End(Not run)  
  
## Here we load the example xpose database  
data(simpraz.xpdb)  
xpdb <- simpraz.xpdb
```

```
## A vanilla plot
cwres.dist.hist(xpdb)
```

cwres.dist.qq	<i>Quantile-quantile plot of conditional weighted residuals (CWRES), for Xpose 4</i>
---------------	--

Description

This is a QQ plot of the distribution of conditional weighted residuals (CWRES) in the dataset, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.qq` function.

Usage

```
cwres.dist.qq(object,
              ...)
```

Arguments

object	An <code>xpose.data</code> object.
...	Other arguments passed to <code>link{xpose.plot.qq}</code> .

Details

Displays a QQ plot of the conditional weighted residuals (CWRES).

Value

Returns a QQ plot of conditional weighted residuals (CWRES).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.qq](#), [xpose.panel.qq](#), [qqmath](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
```



```

data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
cwres.dist.qq(xpdb)

## A different plotting character
cwres.dist.qq(xpdb, pch=4)

```

cwres.vs.cov	<i>Conditional Weighted residuals (CWRES) plotted against covariates, for Xpose 4</i>
--------------	---

Description

This creates a stack of plots of conditional weighted residuals (CWRES) plotted against covariates, and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` and `xpose.plot.histogram` functions. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```

cwres.vs.cov(object,
             ylb = "CWRES",
             smooth=TRUE,
             type="p",
             main="Default",
             ...)

```

Arguments

object	An <code>xpose.data</code> object.
ylb	A string giving the label for the y-axis. NULL if none.
smooth	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
type	1-character string giving the type of plot desired. The following values are possible, for details, see 'plot': "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c" for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
...	Other arguments passed to <code>link{xpose.plot.default}</code> or <code>link{xpose.plot.histogram}</code> .

Details

Each of the covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$Covariates`, is evaluated in turn, creating a stack of plots.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling xyplots and histograms are available. See [xpose.plot.default](#) and [xpose.plot.histogram](#) for details.

Value

Returns a stack of xyplots and histograms of CWRES versus covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.plot.histogram](#), [xyplot](#), [histogram](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
cwres.vs.cov(xpdb)

## Custom colours and symbols, IDs
cwres.vs.cov(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)

## End(Not run)
```

cwres.vs.idv

Population conditional weighted residuals (CWRES) plotted against the independent variable (IDV) for Xpose 4

Description

This is a plot of population conditional weighted residuals (CWRES) vs the independent variable (IDV), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```

cwres.vs.idv(object,
              abline=c(0,0),
              smooth=TRUE,
              ...)

```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>abline</code>	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
<code>smooth</code>	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Conditional weighted residuals (CWRES) are plotted against the independent variable, as specified in `object@Prefs@Xvardef$idv`.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an xyplot of CWRES vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
cwres.vs.idv(xpdb)

## A conditioning plot
cwres.vs.idv(xpdb, by="HCTZ")

## Logarithmic Y-axis
cwres.vs.idv(xpdb, logy=TRUE)

## Custom colours and symbols, IDs labels
cwres.vs.idv(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

cwres.vs.idv.bw	<i>Box-and-whisker plot of conditional weighted residuals vs the independent variable for Xpose 4</i>
-----------------	---

Description

This creates a box and whisker plot of conditional weighted residuals (CWRES) vs the independent variable (IDV), and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.bw` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
cwres.vs.idv.bw(object,
  ...)
```

Arguments

object	An <code>xpose.data</code> object.
...	Other arguments passed to <code>link{xpose.plot.bw}</code> .

Details

This creates a box and whisker plot of conditional weighted residuals (CWRES) vs the independent variable (IDV), and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the [xpose.plot.bw](#) function. Most of the options take their default values from xpose.data object but may be overridden by supplying them as arguments.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling bwplots are available. See [xpose.plot.bw](#) and [xpose.panel.bw](#) for details.

Value

Returns a stack of box-and-whisker plots of CWRES vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.bw](#), [xpose.panel.bw](#), [bwplot](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
cwres.vs.idv.bw(xpdb)
```

cwres.vs.pred

Population conditional weighted residuals (CWRES) plotted against population predictions (PRED) for Xpose 4

Description

This is a plot of population conditional weighted residuals (cwres) vs population predictions (PRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the [xpose.plot.default](#) function. Most of the options take their default values from xpose.data object but may be overridden by supplying them as arguments.

Usage

```

cwres.vs.pred(object,
               abline=c(0,0),
               smooth = TRUE,
               ...)

```

Arguments

object	An xpose.data object.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
abline	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an xyplot of CWRES vs PRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xyplot](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```

## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
cwres.vs.pred(xpdb)

```

```
## A conditioning plot
cwres.vs.pred(xpdb, by="HCTZ")

## Logarithmic Y-axis
cwres.vs.pred(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
cwres.vs.pred(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

cwres.vs.pred.bw	<i>Box-and-whisker plot of conditional weighted residuals vs population predictions for Xpose 4</i>
------------------	---

Description

This creates a box and whisker plot of conditional weighted residuals (CWRES) vs population predictions (PRED), and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.bw` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
cwres.vs.pred.bw(object,
  ...)
```

Arguments

object	An <code>xpose.data</code> object.
...	Other arguments passed to <code>link{xpose.plot.bw}</code> .

Details

This creates a box and whisker plot of conditional weighted residuals (CWRES) vs population predictions (PRED), and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.bw` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling bwplots are available. See [xpose.plot.bw](#) and [xpose.panel.bw](#) for details.

Value

Returns a box-and-whisker plot of CWRES vs PRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.bw](#), [xpose.panel.bw](#), [bwplot](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
cwres.vs.pred.bw(xpdb)
```

cwres.wres.vs.idv	<i>Weighted residuals (WRES) and conditional WRES (CWRES) plotted against the independent variable (IDV) or the population predictions (PRED) for Xpose 4</i>
-------------------	---

Description

This function graphically compares WRES and CWRES as plotted against the independent variable or the population predictions.

Usage

```
cwres.wres.vs.idv(object,
  ylb = "Residuals",
  abline = c(0,0),
  smooth=TRUE,
  scales=list(),
  ...)

cwres.wres.vs.pred(object,
  ylb = "Residuals",
  abline = c(0,0),
  smooth=TRUE,
  scales=list(),
  ...)
```


Arguments

object	An xpose.data object.
ylb	A string giving the label for the y-axis. NULL if none.
abline	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
smooth	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
scales	scales is passed to <code>xpose.plot.default</code>
...	Other arguments passed to <code>link[lattice]{xyplot}</code> .

Details

This function creates plots of WRES and CWRES, presented side-by-side for comparison.

Conditional weighted residuals (CWRES) require some extra steps to calculate. See [compute.cwres](#) for details.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

A compound xyplot.

Author(s)

Niclas Jonsson & Andrew Hooker

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#), [compute.cwres](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run, table and data files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

cwres.wres.vs.idv(xpdb)
```

Data	<i>Extract data from an xpose.data object.</i>
------	--

Description

Extracts the data from the Data or SData slots in an "xpose.data" object.

Usage

```
Data(object,
      inclZeroWRES = FALSE,
      onlyfirst = FALSE,
      subset = NULL)
```

```
SData(object,
      inclZeroWRES = FALSE,
      onlyfirst = FALSE,
      subset = NULL,
      samp = NULL)
```

```
Data(object,quiet=TRUE,keep.structure=F) <- value
SData(object) <- value
```

Arguments

object	An "xpose.data" object
inclZeroWRES	Logical value indicating whether rows with WRES==0 should be included in the extracted data.
onlyfirst	Logical value indicating whether only the first line per individual should be included in the extracted data.
subset	Expression with which the extracted data should be subset (see xsubset)
samp	An integer between 1 and object@Nsim (see xpose.data-class) specifying which of the simulated data sets to extract from SData.
quiet	TRUE or FALSE if FALSE then some more information is printed out when adding data to an Xpose object.
keep.structure	TRUE or FALSE if FALSE then values are converted to continuous or categorical according to the rules set up by xpose using object@Prefs@Cat.levels, object@Prefs@DV.cat.levels and the values in the "catab" file.
value	An R data.frame.

Details

When using Data to assign a data.frame to the Data slot in the "xpose.data" object a number of things happen:

Each column in the `data.frame` is checked and set to factor if the number of unique values are less than the value of `Cat.levels` (see [xpose.prefs-class](#)).

It is checked which of the predefined `xpose` data variables that exists in the `data.frame`. The variable definitions that does not exist are set to `NULL`.

The column identified by the `dv` `xpose` variable definition, is checked and set to factor if the number of unique values are less than or equal to the `DV.Cat.levels` (see [xpose.prefs-class](#)).

Finally, each column name in the `data.frame` is checked for a label (see [xpose.prefs-class](#)). If it is non-existent, the label is set to the column name.

When `SData` is used to assign a `data.frame` to the `SData` slot it is first checked that the number of rows in the `SData` `data.frame` is an even multiple of the number of rown in `Data`. Next, each column in the `SData` `data.frame` is assigned the same class as the corresponding column in the `Data` `data.frame` (it is required that the columns are the same in `Data` and `SData`). Finally, an extra column, "iter", is added to `SData`, which indicates the iteration number that each row belongs to. At the same time, the `Nsim` slot of the "xpose.data" object is set to the number of iterations (see [nsim](#)).

Value

Returns a `data.frame` from the `Data` or `SData` slots, excluding rows as indicated by the arguments.

Author(s)

Niclas Jonsson

See Also

[xpose.data-class](#), [xpose.prefs-class](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Extract data
my.dataframe <- Data(xpdb5, subset="TIME < 24")

## Assign data
Data(xpdb5) <- my.dataframe

## Extract simulated data
my.simulated.dataframe <- SData(xpdb5)

## Assign simulated data
SData(xpdb5) <- my.simulated.dataframe

## End(Not run)
```

data.checkout	<i>Check through the source dataset to detect problems</i>
---------------	--

Description

This function graphically "checks out" the dataset to identify errors or inconsistencies.

Usage

```
data.checkout(obj=NULL,
              datafile = ".ask.",
              hlin = -99,
              dotcol = "black",
              dotpch = 16,
              dotcex = 1,
              idlab="ID",
              csv=NULL,
              main="Default",
              ...)
```

Arguments

obj	NULL or an xpose.data object.
datafile	A data file, suitable for import by read.table .
hlin	An integer, specifying the line number on which the column headers appear.
dotcol	Colour for the dots in the dotplot. If obj is an xpose data object then the default is to use the same value as defined for box-and-whisker plots.
dotpch	Plotting character for the dots in the dotplot. If obj is an xpose data object then the default is to use the same value as defined for box-and-whisker plots.
dotcex	Relative scaling for the dots in the dotplot. If obj is an xpose data object then the default is to use the same value as defined for box-and-whisker plots.
idlab	The ID column label in the dataset. Input as a text string.
csv	Is the data file in CSV format (comma separated values)? If the value is NULL then the user is asked at the command line. If supplied to the function the value can be TRUE/FALSE.
main	The title to the plot. "default" means that Xpose creates a title.
...	Other arguments passed to <code>link[lattice]{dotplot}</code> .

Details

This function creates a series of dotplots, one for each variable in the dataset, against individual ID. Outliers and clusters may easily be detected in this manner.

Value

A stack of dotplots.

Author(s)

Niclas Jonsson, Andrew Hooker & Justin Wilkins

See Also

[dotplot](#), [xpose.prefs-class](#), [read.table](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run, table and data files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

data.checkout(xpdb5, datafile = "mydata.dta")
data.checkout(datafile = "mydata.dta")

## End(Not run)
```

db.names	<i>Prints the contents of an Xpose data object</i>
----------	--

Description

These functions print a summary of the specified Xpose object to the R console.

Usage

```
db.names(object)

xpose.print(object,
            long = TRUE)
```

Arguments

object	An xpose.data object.
long	A logical operator specifying whether the function should print an abridged (long = FALSE) or full list of its contents.

Details

These functions return a detailed summary of the contents of a specified [xpose.data](#) object.

Value

A detailed summary of the contents of a specified `xpose.data` object.

Author(s)

Niclas Jonsson & Justin Wilkins

See Also

[xpose.data](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

xpose.print(xpdb5)

## End(Not run)
```

dist.hist

Plot the parameter or covariate distributions using a histogram

Description

These functions plot the parameter or covariate values stored in an Xpose data object using histograms.

Usage

```
parm.hist(object,
          onlyfirst = TRUE,
          main="Default",
          ...)
```

```
cov.hist(object,
         onlyfirst=TRUE,
         main="Default",
         ...)
```

```
ranpar.hist(object,
            onlyfirst=TRUE,
            main="Default",
            ...)
```

Arguments

object	An xpose.data object.
onlyfirst	Logical value indicating if only the first row per individual is included in the plot.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
...	Other arguments passed to <code>xpose.plot.histogram</code> .

Details

Each of the parameters or covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$params`, `object@Prefs@Xvardef$covariates` or `object@Prefs@Xvardef$ranpar` is evaluated in turn, creating a stack of histograms.

A wide array of extra options controlling histograms are available. See `xpose.plot.histogram` for details.

Value

Delivers a stack of histograms.

Author(s)

Andrew Hooker & Justin Wilkins

See Also

`xpose.plot.histogram`, `xpose.panel.histogram`, `histogram`, `xpose.data-class`, `xpose.prefs-class`

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## Not run:
## A stack of parameter histograms, with 3 mirror panes
parm.hist(xpdb, mirror=3)

## End(Not run)

## Covariate distribution, in green
cov.hist(xpdb, hicol=11, hidcol="DarkGreen", hiborder="White")
```

dist.qq	<i>Plot the parameter or covariate distributions using quantile-quantile (Q-Q) plots</i>
---------	--

Description

These functions plot the parameter or covariate values stored in an Xpose data object using Q-Q plots.

Usage

```
parm.qq(object,
        onlyfirst = TRUE,
        main="Default",
        ...)
```

```
ranpar.qq(object,
          onlyfirst = TRUE,
          main="Default",
          ...)
```

```
cov.qq(object,
        onlyfirst = TRUE,
        main="Default",
        ...)
```

Arguments

object	An xpose.data object.
onlyfirst	Logical value indicating if only the first row per individual is included in the plot.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
...	Other arguments passed to xpose.plot.qq .

Details

Each of the parameters or covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$params`, `object@Prefs@Xvardef$ranpar` or `object@Prefs@Xvardef$covariates`, is evaluated in turn, creating a stack of Q-Q plots.

A wide array of extra options controlling Q-Q plots are available. See [xpose.plot.qq](#) for details.

Value

Delivers a stack of Q-Q plots.

Author(s)

Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.qq](#), [xpose.panel.qq](#), [qqmath](#), [xpose.data-class](#), [xpose.prefs-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## Not run:
## A stack of parameter histograms, with 3 mirror panes
parm.qq(xpdb5, mirror=3)

## End(Not run)

## A stack of random parameter histograms
ranpar.qq(xpdb)

## Covariate distribution, in green with red line of identity
cov.qq(xpdb, col=11, ablcol=2)
```

dOFV.vs.cov

Change in individual objective function value vs. covariate value.

Description

Change in individual objective function value vs. covariate value.

Usage

```
dOFV.vs.cov(xpdb1,
            xpdb2,
            covariates=xvardef("covariates", xpdb1),
            ylb=expression(paste(Delta, OFV[i])),
            main="Default",
            smooth=TRUE,
            abline=c(0,0),
            ablcol="grey",
```

```

abllwd=2,
abllty="dashed",
max.plots.per.page=1,
...)
```

Arguments

xpdb1	Xpose data object for first NONMEM run
xpdb2	Xpose data object for second NONMEM run
covariates	Covariates to plot against
ylb	Label for Y axis.
main	Title of plot.
smooth	Should we have a smooth?
abline	abline description.
ablcol	color of abline
abllwd	line width of abline
abllty	type of abline
max.plots.per.page	Plots per page.
...	additional arguments to function

Author(s)

Andrew C. Hooker

Examples

```

## Not run:
## read in table files
xpdb8 <- xpose.data(8)
xpdb11 <- xpose.data(11)

## Make some plots
dOFV.vs.cov(xpdb8,xpdb11,"AGE")
dOFV.vs.cov(xpdb8,xpdb11,c("AGE","SECR"))

## End(Not run)
```

dOFV.vs.id

Change in Objective function value vs. removal of individuals.

Description

A plot showing the most and least influential individuals in determining a drop in OFV between two models.

Usage

```
dOFV.vs.id(xpdb1,
           xpdb2,
           sig.drop=-3.84,
           decrease.label.number=3,
           increase.label.number=3,
           id.lab.cex=0.6,
           id.lab.pos=2,
           type="o",
           xlb="Number of subjects removed",
           ylb=expression(paste(Delta,"OFV")),
           main="Default",
           sig.line.col = "red",
           sig.line.lty = "dotted",
           tot.line.col = "grey",
           tot.line.lty = "dashed",
           key=list(columns = 1,
                   lines = list(pch = c(super.sym$pch[1:2],NA,NA),
                                type=list("o","o","1","1"),
                                col = c(super.sym$col[1:2],sig.line.col,tot.line.col),
                                lty = c(super.sym$lty[1:2],sig.line.lty,tot.line.lty)
                                ),
                   text = list(c(
                                expression(paste(Delta, OFV[i] < 0)),
                                expression(paste(Delta, OFV[i] > 0)),
                                expression(paste("Significant ",Delta, OFV)),
                                expression(paste("Total ",Delta, OFV))
                                )),
                   corner=c(0.95,0.5),border=T
           ),
           ...)
```

Arguments

xpdb1	Xpose data object for first NONMEM run ("new" run)
xpdb2	Xpose data object for Second NONMEM run ("reference" run)

sig.drop	What is a significant drop of OFV?
decrease.label.number	How many points should bw labeled with ID values for those IDs with a drop in iOFV?
increase.label.number	How many points should bw labeled with ID values for those IDs with an increase in iOFV?
id.lab.cex	Size of ID labels.
id.lab.pos	ID label position.
type	Type of lines.
xlb	X-axis label.
ylb	Y-axis label.
main	Title of plot.
sig.line.col	Significant OFV drop line color.
sig.line.lty	Significant OFV drop line type.
tot.line.col	Total OFV drop line color.
tot.line.lty	Total OFV drop line type.
key	Legend for plot.
...	Additional arguments to function.

Author(s)

Andrew C. Hooker

Examples

```
## Not run:
library(xpose4)

## first make sure that the iofv values are read into xpose
cur.dir <- getwd()
setwd(paste(cur.dir, "/LAG_TIME", sep=""))
xpdb1 <- xpose.data(1)
setwd(paste(cur.dir, "/TRANSIT_MODEL", sep=""))
xpdb2 <- xpose.data(1)
setwd(cur.dir)

## then make the plot
dOFV.vs.id(xpdb1, xpdb2)

## End(Not run)
```

dOFV1 .vs. dOFV2	<i>Change in individual objective function value 1 vs. individual objective function value 2.</i>
------------------	---

Description

Change in individual objective function value 1 vs. individual objective

Usage

```
dOFV1 .vs. dOFV2(xpdb1,
                 xpdb2,
                 xpdb3,
                 ylb=expression(paste(Delta, OFV1[i])),
                 xlb=expression(paste(Delta, OFV2[i])),
                 main="Default",
                 smooth=NULL,
                 abline=c(0,1),
                 ablcol="grey",
                 abllwd=2,
                 abllty="dashed",
                 lmline=TRUE,
                 ...)
```

Arguments

xpdb1	Xpose data object for first NONMEM run
xpdb2	Xpose data object for second NONMEM run
xpdb3	Xpose data object for third NONMEM run
ylb	Label for Y axis.
xlb	Label for X axis.
main	Title of plot.
smooth	Should we have a smooth?
abline	abline description.
ablcol	color of abline
abllwd	line width of abline
abllty	type of abline
lmline	Linear regression line?
...	Additional arguments to function.

Author(s)

Andrew C. Hooker

Examples

```
## Not run:
## read in table files
xpdb8 <- xpose.data(8)
xpdb8 <- xpose.data(9)
xpdb11 <- xpose.data(11)

## Make the plot
dOFV.vs.cov(xpdb8, xpdb9, xpdb11)

## End(Not run)
```

dv.preds.vs.idv	<i>Observations (DV), individual predictions (IPRED) and population predictions (IPRED) plotted against the independent variable (IDV), for Xpose 4</i>
-----------------	---

Description

This is a compound plot consisting of plots of observations (DV), individual predictions (IPRED), and population predictions (PRED) against the independent variable (IDV), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function.

Usage

```
dv.preds.vs.idv(object,
  ylb = "Observations/Predictions",
  layout=c(3,1),
  smooth=TRUE,
  scales=list(),
  ...)
```

Arguments

object	An <code>xpose.data</code> object.
ylb	A string giving the label for the y-axis. NULL if none.
layout	A list controlling the number of columns and rows in a compound plot. The default is 2 columns and 1 row.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
scales	A list to be used for the scales argument in <code>xyplot</code> .
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling `xyplots` are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns a compound plot comprising plots of observations (DV), individual predictions (IPRED), and population predictions (PRED) against the independent variable (IDV).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[dv.vs.idv](#), [ipred.vs.idv](#), [pred.vs.idv](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.preds.vs.idv(xpdb)

## Custom colours and symbols, IDs
dv.preds.vs.idv(xpdb, cex=0.6, pch=8, col=1, ids=TRUE)
```

dv.vs.idv	<i>Observations (DV) plotted against the independent variable (IDV) for Xpose 4</i>
-----------	---

Description

This is a plot of observations (DV) vs the independent variable (IDV), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
dv.vs.idv(object,
           smooth = TRUE,
           ...)
```

Arguments

object	An xpose.data object.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling `xyplot` are available. See `xpose.plot.default` and `xpose.panel.default` for details.

Value

Returns an xyplot of DV vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

`xpose.plot.default`, `xpose.panel.default`, `xyplot`, `xpose.prefs-class`, `xpose.data-class`

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.vs.idv(xpdb)

## A conditioning plot
dv.vs.idv(xpdb, by="HCTZ")

## Logarithmic Y-axis
dv.vs.idv(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
dv.vs.idv(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

dv.vs.ipred	<i>Observations (DV) plotted against individual predictions (IPRED) for Xpose 4</i>
-------------	---

Description

This is a plot of observations (DV) vs individual predictions (IPRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
dv.vs.ipred(object,  
            abline=c(0,1),  
            smooth=TRUE,  
            ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>abline</code>	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
<code>smooth</code>	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling `xyplot` are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an `xyplot` of DV vs IPRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.vs.ipred(xpdb)

## A conditioning plot
dv.vs.ipred(xpdb, by="HCTZ")

## Logarithmic Y-axis
dv.vs.ipred(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
dv.vs.ipred(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

dv.vs.ipred.by.cov *Dependent variable vs individual predictions, conditioned on covariates, for Xpose 4*

Description

This is a plot of dependent variable (DV) vs individual predictions (IPRED) conditioned by covariates, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
dv.vs.ipred.by.cov(object,
  abline = c(0,1),
  smooth = TRUE,
  main="Default",
  ...)
```

Arguments

object	An <code>xpose.data</code> object.
abline	Vector of arguments to the panel.abline function. No abline is drawn if NULL.

smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Each of the covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$Covariates`, is evaluated in turn, creating a stack of plots.

A wide array of extra options controlling `xyplot` are available. See `xpose.plot.default` and `xpose.panel.default` for details.

Value

Returns a stack of `xyplots` of DV vs IPRED, conditioned on covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[dv.vs.ipred](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.vs.ipred.by.cov(xpdb)

## Custom axis labels
dv.vs.ipred.by.cov(xpdb, ylb="DV", xlb="PRED")

## Custom colours and symbols, IDs
dv.vs.ipred.by.cov(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)

## End(Not run)
```

dv.vs.ipred.by.idv	<i>Dependent variable vs individual predictions, conditioned on independent variable, for Xpose 4</i>
--------------------	---

Description

This is a plot of the dependent variable (DV) vs individual predictions (IPRED) conditioned by the independent variable, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
dv.vs.ipred.by.idv(object,  
  abline = c(0,1),  
  smooth = TRUE,  
  ...)
```

Arguments

object	An <code>xpose.data</code> object.
abline	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling `xyplot` are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns a stack of `xyplots` of DV vs IPRED, conditioned on the independent variable.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[dv.vs.ipred](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.vs.ipred.by.idv(xpdb)

## Custom axis labels
dv.vs.ipred.by.idv(xpdb, ylb="DV", xlb="PRED")

## Custom colours and symbols, IDs
dv.vs.ipred.by.idv(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

dv.vs.pred	<i>Observations (DV) plotted against population predictions (PRED) for Xpose 4</i>
------------	--

Description

This is a plot of observations (DV) vs population predictions (PRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
dv.vs.pred(object,
            abline=c(0,1),
            smooth=TRUE,
            ...)
```

Arguments

object	An <code>xpose.data</code> object.
abline	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an xyplot of DV vs PRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.vs.pred(xpdb)

## A conditioning plot
dv.vs.pred(xpdb, by="HCTZ")

## Logarithmic Y-axis
dv.vs.pred(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
dv.vs.pred(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

dv.vs.pred.by.cov

Dependent variable vs population predictions, conditioned on covariates, for Xpose 4

Description

This is a plot of the dependent variable (DV) vs population predictions (PRED) conditioned by covariates, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
dv.vs.pred.by.cov(object,
                  abline = c(0,1),
                  smooth = TRUE,
                  main="Default",
                  ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>abline</code>	Vector of arguments to the <code>panel.abline</code> function. No abline is drawn if NULL.
<code>smooth</code>	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
<code>main</code>	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Each of the covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$Covariates`, is evaluated in turn, creating a stack of plots.

A wide array of extra options controlling xyplots are available. See `xpose.plot.default` and `xpose.panel.default` for details.

Value

Returns a stack of xyplots of DV vs PRED, conditioned on covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[dv.vs.pred](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.vs.pred.by.cov(xpdb)

## Custom axis labels
dv.vs.pred.by.cov(xpdb, ylb="DV", xlb="PRED")

## Custom colours and symbols, IDs
dv.vs.pred.by.cov(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)

## End(Not run)
```

dv.vs.pred.by.idv	<i>Dependent variable vs population predictions, conditioned on independent variable, for Xpose 4</i>
-------------------	---

Description

This is a plot of the dependent variable (DV) vs population predictions (PRED) conditioned by the independent variable, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
dv.vs.pred.by.idv(object,
  abline = c(0,1),
  smooth = TRUE,
  ...)
```

Arguments

object	An <code>xpose.data</code> object.
abline	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns a stack of xyplots of DV vs PRED, conditioned on the independent variable.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[dv.vs.pred](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.vs.pred.by.idv(xpdb)

## Custom axis labels
dv.vs.pred.by.idv(xpdb, ylb="DV", xlb="PRED")

## Custom colours and symbols, IDs
dv.vs.pred.by.idv(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

dv.vs.pred.ipred

Observations (DV) are plotted against individual predictions (IPRED) and population predictions (PRED), for Xpose 4

Description

This is a compound plot consisting of plots of observations (DV) against individual predictions (IPRED) and population predictions (PRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function.

Usage

```
dv.vs.pred.ipred(object,
  xlb = "Predictions",
  layout=c(2,1),
  abline=c(0,1),
  lmline=TRUE,
  smooth=NULL,
  scales=list(),
  ...)
```

Arguments

object	An <code>xpose.data</code> object.
xlb	A string giving the label for the x-axis. NULL if none.
layout	A list giving the layout of the graphs on the plot, in columns and rows.
abline	Vector of arguments to the <code>panel.abline</code> function. No abline is drawn if NULL.
lmline	logical variable specifying whether a linear regression line should be superimposed over an <code>xyplot</code> . NULL ~ FALSE. ($y \sim x$)
scales	A list to be used for the scales argument in <code>xyplot</code> .
smooth	NULL or TRUE value indicating whether an x-y smooth should be superimposed.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Plots of DV vs PRED and IPRED are presented side by side for comparison.

A wide array of extra options controlling `xyplots` are available. See `xpose.plot.default` and `xpose.panel.default` for details.

Value

Returns a compound plot comprising plots of observations (DV) against individual predictions (IPRED) and population predictions (PRED).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[dv.vs.pred](#), [dv.vs.ipred](#), [xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
dv.vs.pred.ipred(xpdb)

## Custom colours and symbols, IDs
dv.vs.pred.ipred(xpdb, cex=0.6, pch=8, col=1, ids=TRUE)
```

export.graph.par	<i>Exports Xpose graphics settings to a file.</i>
------------------	---

Description

This function exports graphics settings for a specified Xpose data object to a file.

Usage

```
export.graph.par(object)

xpose.write(object,
            file = "xpose.ini")
```

Arguments

object	An xpose.data object.
file	The file to contain exported Xpose settings.

Details

This function exports the graphics settings (contents of `object@Prefs@Graph.prefs`) for a given `xpose.data` object to a file, typically 'xpose.ini'. It is a wrapper for `xpose.write`. Note that the file format is not the same as is used in [import.variable.definitions](#) and [export.variable.definitions](#).

Value

Null.

Author(s)

Niclas Jonsson & Justin Wilkins

See Also

[import.graph.par](#), [xpose.prefs-class](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## For a filename prompt
export.graph.par(xpdb5)

## Command-line driven
xpose.write(xpdb5, "c:/XposeSettings/mytheme.ini")

## End(Not run)
```

export.variable.definitions

Exports Xpose variable definitions to a file from an Xpose data object.

Description

This function exports variable definitions for a specified Xpose data object to a file.

Usage

```
export.variable.definitions(object)
```

Arguments

object An xpose.data object.

Details

This function exports variable definitions (contents of `object@Prefs@Xvardef`) for a given `xpose.data` object to a file, typically `'xpose.vardefs.ini'`. Note that file format is not the same as used for graphics settings. It is a wrapper for the R function [dput](#).

Value

Null.

Author(s)

Niclas Jonsson & Justin Wilkins

See Also

[import.variable.definitions](#), [xpose.prefs-class dput](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

export.variable.definitions(xpdb5)

## End(Not run)
```

functions

Data functions for Xpose 4

Description

These functions perform various tasks in managing Xpose data objects.

Usage

```
check.vars(vars,
           object,
           silent=FALSE)

is.readable.file(filename)

test.xpose.data(object)
```

Arguments

vars	List of variables to be checked.
object	An xpose.data object.
silent	A logical operator specifying whether output should be displayed.
filename	A filename to be checked for readability.

Details

These are internal Xpose functions, not intended for direct use.

Value

TRUE, FALSE or NULL.

Author(s)

Niclas Jonsson and Andrew Hooker

See Also

[xpose.prefs-class](#), [xvardef](#)

gam.functions

GAM functions for Xpose 4

Description

These are functions for running the generalized additive model within Xpose 4, summarizing its results, and plotting them.

Usage

```
xp.gam(object,  
  parnam=xvardef("parms", object),  
  covnams = xvardef("covariates", object),  
  wts.col=NULL,  
  ask.for.input=TRUE,  
  overwrite=TRUE,  
  ...)  
  
check.gamobj()  
  
xp.akaike.plot(title = NULL,  
  xlb = "Akaike value",  
  ylb = "Models",  
  gamobj=NULL,  
  ...)  
  
xp.check.scope(object,  
  covnam = xvardef("covariates", object),  
  nmods = object@Prefs@Gam.prefs$nmods,  
  smoother1 = object@Prefs@Gam.prefs$smoother1,  
  smoother2 = object@Prefs@Gam.prefs$smoother2,  
  smoother3 = object@Prefs@Gam.prefs$smoother3,  
  smoother4 = object@Prefs@Gam.prefs$smoother4,  
  arg1 = object@Prefs@Gam.prefs$arg1,  
  arg2 = object@Prefs@Gam.prefs$arg2,  
  arg3 = object@Prefs@Gam.prefs$arg3,
```

```
    arg4 = object@Prefs@Gam.prefs$arg4,  
    excl1 = object@Prefs@Gam.prefs$excl1,  
    excl2 = object@Prefs@Gam.prefs$excl2,  
    excl3 = object@Prefs@Gam.prefs$excl3,  
    excl4 = object@Prefs@Gam.prefs$excl4,  
    extra = object@Prefs@Gam.prefs$extra,  
    ...)  
  
xp.cook(gam.object)  
  
xp.ind.inf.fit(plot.ids = TRUE,  
              idsceex = 0.7,  
              ptsceex = 0.7,  
              title = NULL,  
              recur = FALSE,  
              xlb = NULL,  
              ylb = NULL,  
              gamobj=NULL,  
              ...)  
  
xp.ind.inf.terms(xlb = NULL,  
                ylb = NULL,  
                plot.ids = TRUE,  
                idsceex = 0.7,  
                ptsceex = 0.7,  
                prompt = TRUE,  
                gamobj=NULL,  
                ...)  
  
xp.ind.stud.res(title = NULL,  
               recur = FALSE,  
               xlb = NULL,  
               ylb = NULL,  
               gamobj=NULL)  
  
xp.plot(plot.ids = TRUE,  
        idsceex = 0.7,  
        ptsceex = 0.7,  
        prompt = TRUE,  
        gamobj=NULL,  
        ...)  
  
xp.summary(gamobj=NULL)
```

Arguments

object An xpose.data object.

title	A text string indicating plot title. If NULL, left blank.
x1b	A text string indicating x-axis legend. If NULL, left blank.
y1b	A text string indicating y-axis legend. If NULL, left blank.
covnam	A list of covariate variables to use in the GAM search.
wts.col	Column in the object@Data.firstonly to use as weights on the parnam values.
nmodels	The number of modelfits to use when setting GAM scope. The default is 3.
smoother1	Smoother for each model.
smoother2	Smoother for each model.
smoother3	Smoother for each model.
smoother4	Smoother for each model.
arg1	Argument for model 1.
arg2	Argument for model 2.
arg3	Argument for model 3.
arg4	Argument for model 4.
excl1	Covariate exclusion from model 1.
excl2	Covariate exclusion from model 2.
excl3	Covariate exclusion from model 3.
excl4	Covariate exclusion from model 4.
extra	Scope parameter for the GAM.
gam.object	A GAM object (see gam).
plot.ids	Logical, specifies whether or not ID numbers should be displayed.
idsceX	ID label size.
ptsceX	Point size.
recur	If dispersion should be used in the GAM object.
prompt	Specifies whether or not the user should be prompted to press RETURN between plot pages. Default is TRUE.
parnam	The parameter to run the GAM on.
covnams	The covariates to test on the parnam
ask.for.input	Should the program ask for input from the user? Can be TRUE or FALSE.
overwrite	Should we overwrite the gam object stored in memory if it exists already. Can be TRUE or FALSE.
gamobj	A GAM object to use in the plot. IF null then the user is asked to choose from a list of GAM objects in memory.
...	Other arguments passed to the GAM functions.

Details

Forthcoming.

Value

A GAM fit.

Author(s)

Niclas Jonsson & Andrew Hooker

See Also

[gam](#), [dotplot](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb <- xpose.data(5)

## Run a GAM
xp.gam(xpdb)

## Summarize GAM
xp.summary()

## An Akaike plot of the results
xp.akaike.plot()

## Studentized residuals
xp.ind.stud.res()

## Individual influence on GAM fit
xp.ind.inf.fit(plot.ids=xpdb@Prefs@Gam.prefs$plot.ids)

## Individual influence on GAM terms
xp.ind.inf.terms(plot.ids=xpdb@Prefs@Gam.prefs$plot.ids)

## End(Not run)
```

generic

Generic internal functions for Xpose 4

Description

These are internal functions relating to the Xpose generic functions.

Details

These are internal Xpose functions, for adding ID numbers, computing prediction intervals, randomization, stacking, and binning. They are not intended for direct use.

Value

Internal helper functions for the generic Xpose functions.

Author(s)

Justin Wilkins

gof

Structured goodness of fit diagnostics.

Description

This is a template function for creating structured goodness of fit diagnostics using the functions in the Xpose specific library.

Usage

```
gof(runno=NULL, save=FALSE, onefile=FALSE,
    saveType="pdf", pageWidth=7.6, pageHeight=4.9,
    structural = TRUE, residual=TRUE, covariate=FALSE,
    iiv=FALSE, iov=FALSE, all=FALSE, myTrace=xpPage)
```

Arguments

runno	The run number fo Xpose to identify the appropriate files to read. In addition runno is used to construct the file name to save plots in. runno can also be NULL for cases in which the function is used for non-Xpose based code.
save	Logical. TRUE if the plot(s) is to be saved in a file. FALSE if the plot(s) is to be displayed on screen. The plot(s) will be saved in a file named with the function name followed by the word 'run', the run number, an order number followed by a file name extension appropriate for the selected saveType. For example 'gofrun1-01.pdf' for the first plot file created by a script called gof based on output from run 1 and saveType='pdf'.
onefile	Logical. TRUE if plots are to be save in a single file and FALSE if each plot should be saved as a separate file. In the latter case, each file will be have an incremented order number (01-99).
saveType	The type of graphics file to produce if save=TRUE. Allowed values are 'pdf' (default), 'wmf' (only Windows) and 'png'.
pageWidth	The width of the graphics device in inches.
pageHeight	The height of the graphics device in inches.
structural	Logical. TRUE if the code in the structural model section (see below) should be executed and FALSE if not.
residual	Logical. TRUE if the code in the residual model section (see below) should be executed and FALSE if not.

covariate	Logical. TRUE if the code in the covariate model section (see below) should be executed and FALSE if not.
iiv	Logical. TRUE if the code in the IIV model section (see below) should be executed and FALSE if not.
iov	Logical. TRUE if the code in the IOV model section (see below) should be executed and FALSE if not.
all	Logical. TRUE if the code in all sections (see below) should be executed.
myTrace	NULL or the name of a function. The value of myTrace can used with the lattice page= argument to annotate plots for tracability.

Details

The `gof` function is provided as a template to facilitate the (structured) use of the functions in the Xpose specific library. Xpose specific is extensively described in the 'Xpose Bestiary'.

The function can be renamed so that multiple scripts can be used in paralell.

The function is set up to make it easy to display plots on screen as well as to save them in files. In the latter case, plots are save in a sub-directory called 'Plots'.

The arguments `structural`, `residual`, `covariate`, `iiv`, `iov` and `all` are just "switches" to different parts of the code (if-blocks). These blocks can be removed or the default values of the arguments changed to better suit the needs of the user.

It is also possible to add tracing information to the produced plots. This is done via the `myTrace` argument. A non-NULL value should be a function that returns a `panel.text` object. The default is the `xpPage` function that will put a string concatenated from the device name, function name, working directory and date, in small, faint grey, font at the bottom of each graph page. Note that the user need to add `page=myTrace` as an argument to the Xpose functions for this to have an effect.

The function calls a support function called `gofSetup`, which is responsible for setting up the graphics device and determining the file names for saved graphs.

Value

Does not return anything unless the user specify a return value.

Author(s)

E. Niclas Jonsson, Mats Karlsson and Andrew Hooker

See Also

[xpose4-package](#)

Examples

```
## Not run:  
## This is an example of how the function may be setup by a user.  
  
library(xpose4)  
mygof <- gof
```

```

fix(mygof)

mygof <- function (runno = NULL, save = FALSE, onefile = FALSE, saveType = "pdf",
  pageWidth = 7.6, pageHeight = 4.9, structural = TRUE, residual = TRUE,
  covariate = FALSE, iiv = FALSE, iov = FALSE, all = FALSE, myTrace=xpPage) {

  gofSetup(runno, save, onefile, saveType, pageWidth, pageHeight)
  xpdb <- xpose.data(runno)

  if (structural || all) {
    xplot <- dv.vs.pred.ipred(xpdb, page = myPage)
    print(xplot)
  }
  if (residual || all) {
    xplot <- absval.wres.vs.pred(xpdb, page = myPage)
    print(xplot)
  }
  if (covariate || all) {
  }
  if (iiv || all) {
  }
  if (iov || all) {
  }
  if (save) dev.off()
  invisible()
}

## The function can then be execute, e.g.:
mygof(1)

## End(Not run)

```

import.graph.par

Imports Xpose graphics settings from a file to an Xpose data object.

Description

This function imports graphics settings for a specified Xpose data object from a file.

Usage

```

import.graph.par(object, classic = FALSE)

xpose.read(object,
  file = "xpose.ini")

```

Arguments

object	An xpose.data object.
classic	A logical operator specifying whether the function should assume the classic menu system. This is an internal option and need never be called from the command line.
file	The file where default graphical settings are located.

Details

This function imports graphics settings (contents of object@Prefs@Graph.prefs) for a given xpose.data object from a file, typically 'xpose.ini'. It is a wrapper for xpose.read. It returns an xpose.data object. Note that the file format is not the same as is used in [import.variable.definitions](#) and [export.variable.definitions](#).

Value

An [xpose.data](#) object (classic = FALSE) or null (classic = TRUE).

Author(s)

Niclas Jonsson & Justin Wilkins

See Also

[export.graph.par](#), [xpose.prefs-class](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Import graphics preferences you saved earlier using export.graph.par
xpdb5 <- import.graph.par(xpdb5)

## Command-line driven
xpdb5 <- xpose.read(xpdb5, "c:/XposeSettings/mytheme.ini")

## End(Not run)
```

```
import.variable.definitions
```

Imports Xpose variable definitions from a file to an Xpose data object.

Description

This function imports variable definitions for a specified Xpose data object from a file.

Usage

```
import.variable.definitions(object, classic = FALSE)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>classic</code>	A logical operator specifying whether the function should assume the classic menu system. This is an internal option and need never be called from the command line.

Details

This function imports variable definitions (contents of `object@Prefs@Xvardef`) for a given `xpose.data` object from a file, typically `'xpose.vardefs.ini'`. It returns an `xpose.data` object. Note that file format is not the same as used for graphics settings. It is a wrapper for the R function [dget](#).

Value

An `xpose.data` object (`classic == FALSE`) or null (`classic == TRUE`).

Author(s)

Niclas Jonsson & Justin Wilkins

See Also

[export.variable.definitions](#), `xpose.prefs-class` [dget](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

xpdb5 <- import.variable.definitions(xpdb5)

## End(Not run)
```

ind.plots	<i>Observations (DV), individual predictions (IPRED) and population predictions (PRED) are plotted against the independent variable for every individual in the dataset, for Xpose 4</i>
-----------	--

Description

This is a compound plot consisting of plots of observations (DV), individual predictions (IPRED) and population predictions (PRED) against the independent variable for every individual in the dataset, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function.

Usage

```
ind.plots(object,
  y.vals = c(
    xvardef("dv", new.obj),
    xvardef("ipred", new.obj),
    xvardef("pred", new.obj)
  ),
  x.vals = xvardef("idv", new.obj),
  id.vals = xvardef("id", new.obj),
  key.text = y.vals,
  main = "Default",
  key="Default",
  xlb = xlabel(xvardef("idv", object), object),
  ylb = NULL,
  layout = c(4, 4),
  inclZeroWRES = FALSE,
  subset = xsubset(object),
  type = "o",
  grid=FALSE,
  col = c(1, 2, 4),
  lty = c(0, 1, 3),
  lwd = c(1, 1, 1),
  pch = c(21, 32, 32),
  cex = c(0.7, 0.7, 0.7),
  fill=c("lightgrey", 0, 0),
  prompt = FALSE,
  mirror=NULL,
  main.cex=0.9,
  max.plots.per.page=1,
  pch.ip.sp=c(21, 19, 18),
  cex.ip.sp=c(0.7, 0.4, 0.4),
  y.vals.subset=NULL,
  ...)
```

Arguments

object	An xpose.data object.
y.vals	The Y values to use.
x.vals	The X values to use.
id.vals	The ID values to use.
key.text	The text in the legend to use.
key	Create a legend.
grid	Should the plots have a grid in each plot?
fill	Fill the circles in the points?
max.plots.per.page	Maximum number of plots per page.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot. The default is TRUE.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
xlb	A string giving the label for the x-axis. NULL if none.
ylb	A string giving the label for the y-axis. NULL if none.
layout	A list giving the layout of the graphs on the plot, in columns and rows. The default is 4x4.
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
type	1-character string giving the type of plot desired. The default is "o", for over-plotted points and lines. See xpose.plot.default .
col	A list of three elements, giving plotting characters for observations, individual predictions, and population predictions, in that order. The default is black for DV, red for individual predictions, and blue for population predictions.
lty	A list of three elements, giving line types for observations, individual predictions, and population predictions, in that order.
lwd	A list of three elements, giving line widths for observations, individual predictions, and population predictions, in that order.
pch	A list of three elements, giving plotting characters for observations, individual predictions, and population predictions, in that order.
cex	A list of three elements, giving relative point size for observations, individual predictions, and population predictions, in that order. The default is c(0.7,0.7,0.7).
prompt	Specifies whether or not the user should be prompted to press RETURN between plot pages. Default is TRUE.
mirror	Mirror plots are not yet implemented in this function and this argument must contain a value of NULL
main.cex	The size of the title.

<code>pch.ip.sp</code>	If there is a panel with just one observation then this specifies the type of points for the DV, IPRED and PRED respectively.
<code>cex.ip.sp</code>	If there is a panel with just one observation then this specifies the size of the points for the DV, IPRED and PRED respectively.
<code>y.vals.subset</code>	Used to subset on the DV, IPRED and PRED variables separately. Either NULL or a vector of three strings, corresponding to the subset of DV, IPRED and PRED respectively. See examples below.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Matrices of individual plots are presented for comparison and closer inspection.

Value

Returns a stack of plots observations (DV) against individual predictions (IPRED) and population predictions (PRED).

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [strip.default](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
ind.plots(xpdb)

## Monochrome, suitable for manuscript or report
ind.plots(xpdb,
  subset="ID>40 & ID<57",
  col=c(1,1,1),
  lty=c(0,2,3),
```

```

strip=function(..., bg)
  strip.default(..., bg="grey"))

## Not run:
## IF we simulate in NONMEM using a dense grid of time points
## and all non-observed DV items are equal to zero.
ind.plots(xpdb,inclZeroWRES=TRUE,y.vals.subset=c("DV!=0","NULL","NULL"))

## End(Not run)

```

ind.plots.wres.hist *Histograms of weighted residuals for each individual in an Xpose data object, for Xpose 4*

Description

This is a compound plot consisting of histograms of the distribution of weighted residuals (any weighted residual available from NONMEM) for every individual in the dataset. It is a wrapper encapsulating arguments to the `xpose.plot.histogram` function.

Usage

```

ind.plots.wres.hist(object,
  main = "Default",
  wres="wres",
  ylb = NULL,
  layout=c(4,4),
  inclZeroWRES=FALSE,
  subset=xsubset(object),
  scales=list(cex=0.7,tck=0.5),
  aspect="fill",
  force.by.factor=TRUE,
  ids=F,
  as.table=TRUE,
  hicol = object@Prefs@Graph.prefs$hicol,
  hilty = object@Prefs@Graph.prefs$hilty,
  hilwd = object@Prefs@Graph.prefs$hilwd,
  hidcol = object@Prefs@Graph.prefs$hidcol,
  hidlty = object@Prefs@Graph.prefs$hidlty,
  hidlwd = object@Prefs@Graph.prefs$hidlwd,
  hiborder = object@Prefs@Graph.prefs$hiborder,
  prompt = FALSE,
  mirror=NULL,
  main.cex=0.9,
  max.plots.per.page=1,
  ...)

ind.plots.cwres.hist(object,

```

```
wres="cwres",
...)
```

Arguments

object	An xpose.data object.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
wres	Which weighted residual should we plot? Defaults to the WRES.
ylb	A string giving the label for the y-axis. NULL if none.
layout	A list giving the layout of the graphs on the plot, in columns and rows. The default is 4x4.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot. The default is FALSE.
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
scales	see xpose.plot.histogram
aspect	see xpose.plot.histogram
force.by.factor	see xpose.plot.histogram
ids	see xpose.plot.histogram
as.table	see xpose.plot.histogram
hiborder	the border colour of the histogram - an integer or string. The default is black (see histogram).
hicol	the fill colour of the histogram - an integer or string. The default is blue (see histogram).
hilty	the border line type of the histogram - an integer. The default is 1 (see histogram).
hilwd	the border line width of the histogram - an integer. The default is 1 (see histogram).
hidcol	the fill colour of the density line - an integer or string. The default is black (see histogram).
hidlty	the border line type of the density line - an integer. The default is 1 (see histogram).
hidlwd	the border line width of the density line - an integer. The default is 1 (see histogram).
prompt	Specifies whether or not the user should be prompted to press RETURN between plot pages. Default is FALSE.
mirror	Mirror plots are not yet implemented in this function and this argument must contain a value of NULL
main.cex	The size of the title.
max.plots.per.page	Maximum number of plots per page
...	Other arguments passed to xpose.plot.histogram .

Details

Matrices of histograms of weighted residuals in each included individual are displayed. `ind.plots.cwres.hist` is just a wrapper for `ind.plots.wres.hist(object, wres="cwres")`.

Value

Returns a compound plot comprising histograms of weighted residual conditioned on individual.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins & Andrew Hooker

See Also

[xpose.plot.histogram](#), [xpose.panel.histogram](#), [histogram](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
ind.plots.wres.hist(xpdb)

## Different plotting character
ind.plots.wres.hist(xpdb, pch=3)

## Different plotting character
ind.plots.wres.hist(xpdb, pch=3)

##with a grid
ind.plots.wres.hist(xpdb, grid=TRUE)

## subsets
ind.plots.wres.hist(xpdb, subset="ID<10")
ind.plots.wres.hist(xpdb, subset="ID<10 | ID>45",grid=TRUE)

## plot the CWRES instead
ind.plots.wres.hist(xpdb, wres="CWRES")
ind.plots.cwres.hist(xpdb)
ind.plots.cwres.hist(xpdb, subset="ID<10 | ID>45",grid=TRUE)

## End(Not run)
```

ind.plots.wres.qq	<i>Quantile-quantile plots of weighted residuals for each individual in an Xpose data object, for Xpose 4</i>
-------------------	---

Description

This is a compound plot consisting of QQ plots of the distribution of weighted residuals (any weighted residual produced by NONMEM) for every individual in the dataset. The function is a wrapper encapsulating arguments to the [xpose.plot.qq](#) function.

Usage

```
ind.plots.wres.qq(object,
  main = "Default",
  wres="wres",
  layout=c(4,4),
  inclZeroWRES=FALSE,
  subset=xsubset(object),
  scales=list(cex=0.7,tck=0.5),
  aspect="fill",
  force.by.factor=TRUE,
  ids=F,
  as.table=TRUE,
  type="o",
  pch=object@Prefs@Graph.prefs$pch,
  col=object@Prefs@Graph.prefs$col,
  cex=object@Prefs@Graph.prefs$cex,
  abllty = object@Prefs@Graph.prefs$abllty,
  abllwd = object@Prefs@Graph.prefs$abllwd,
  ablcol = object@Prefs@Graph.prefs$ablcol,
  prompt = FALSE,
  main.cex=0.9,
  mirror=NULL,
  max.plots.per.page=1,
  ...)

ind.plots.cwres.qq(object,
  wres="cwres",
  ...)
```

Arguments

object	An xpose.data object.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.

wres	Which weighted residual should we plot? Defaults to the WRES.
layout	A list giving the layout of the graphs on the plot, in columns and rows. The default is 4x4.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot. The default is FALSE.
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
scales	See xpose.plot.qq .
aspect	See xpose.plot.qq .
force.by.factor	See xpose.plot.qq .
ids	See xpose.plot.qq .
as.table	See xpose.plot.qq .
type	1-character string giving the type of plot desired. The following values are possible, for details, see 'plot': "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c") for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
col	The color for lines and points. Specified as an integer or a text string. A full list is obtained by the R command <code>colours()</code> . The default is blue (<code>col=4</code>).
pch	The plotting character, or symbol, to use. Specified as an integer. See R help on points . The default is an open circle.
cex	The amount by which plotting text and symbols should be scaled relative to the default. 'NULL' and 'NA' are equivalent to '1.0'.
abllwd	Line width of the line of identity.
ablty	Line type of the line of identity.
ablcol	Line colour of the line of identity.
prompt	Specifies whether or not the user should be prompted to press RETURN between plot pages. Default is FALSE.
mirror	Mirror plots are not yet implemented in this function and this argument must contain a value of NULL
main.cex	The size of the title.
max.plots.per.page	Maximum number of plots per page
...	Other arguments passed to <code>link{xpose.plot.qq}</code> .

Details

Matrices of QQ plots of weighted residuals in each included individual are displayed.

A wide array of extra options controlling QQ plots are available. See [xpose.plot.qq](#) for details.

Value

Returns a compound plot comprising QQ plots of weighted residuals conditioned on individual.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins & Andrew Hooker

See Also

[xpose.plot.qq](#), [xpose.panel.qq](#), [qqplot](#), [qqmath](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
ind.plots.wres.qq(xpdb)

## Custom colours
ind.plots.wres.qq(xpdb, hicol=5, hidcol=2)

## with a grid
ind.plots.wres.qq(xpdb, grid=TRUE)

## Subset on ID value
ind.plots.wres.qq(xpdb, grid=TRUE, subset="ID<10")

## Use CWRES instead
ind.plots.wres.qq(xpdb, grid=TRUE, subset="ID<10", wres="cwres")
ind.plots.cwres.qq(xpdb, grid=TRUE, subset="ID<10")

## Use NPDEs instead
ind.plots.wres.qq(xpdb, grid=TRUE, subset="ID<10", wres="NPDE")

## End(Not run)
```

internal.functions *Internal functions for the VPC*

Description

Internal functions for the VPC

Usage

```
find.right.table(object, inclZeroWRES, onlyfirst,
                samp, PI.subset, subscripts,
                PI.bin.table, panel.number, ...)
get.polygon.regions(PPI, PI.mirror, ...)
setup.PPI(PIlimits, PI.mirror, tmp.table, ...)
```

Arguments

object	Xpose object
inclZeroWRES	Include row sof data with WRES=0
onlyfirst	Use only first data for each individual
samp	sample number
PI.subset	Prediction interval subset
subscripts	subscripts
PI.bin.table	prediction interval binning table
panel.number	panel number
PPI	Plot prediction intervals
PI.mirror	Prediction interval mirror
PIlimits	Prediction interval limits
tmp.table	temporary table
...	Extra options passed to arguments

Value

Returned to [xpose.VPC](#)

ipred.vs.idv	<i>Individual predictions (IPRED) plotted against the independent variable (IDV) for Xpose 4</i>
--------------	--

Description

This is a plot of Individual predictions (IPRED) vs the independent variable (IDV), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
ipred.vs.idv(object,
             smooth = TRUE,
             ...)
```


Arguments

object	An xpose.data object.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an xyplot of IPRED vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
ipred.vs.idv(xpdb)

## A conditioning plot
ipred.vs.idv(xpdb, by="HCTZ")

## Logarithmic Y-axis
ipred.vs.idv(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
ipred.vs.idv(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

iwres.dist.hist *Histogram of individual weighted residuals (IWRES), for Xpose 4*

Description

This is a histogram of the distribution of individual weighted residuals (IWRES) in the dataset, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.histogram` function.

Usage

```
iwres.dist.hist(object,  
                ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>...</code>	Other arguments passed to <code>xpose.plot.histogram</code> .

Details

Displays a histogram of the individual weighted residuals (IWRES).

Value

Returns a histogram of individual weighted residuals (IWRES).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.histogram](#), [xpose.panel.histogram](#), [histogram](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
## End(Not run)  
  
## Here we load the example xpose database  
data(simpraz.xpdb)  
xpdb <- simpraz.xpdb  
  
## A vanilla plot
```

```
iwres.dist.hist(xpdb)
```

iwres.dist.qq	<i>Quantile-quantile plot of individual weighted residuals (IWRES), for Xpose 4</i>
---------------	---

Description

This is a QQ plot of the distribution of individual weighted residuals (IWRES) in the dataset, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.qq` function.

Usage

```
iwres.dist.qq(object,
              ...)
```

Arguments

object	An <code>xpose.data</code> object.
...	Other arguments passed to <code>link{xpose.plot.qq}</code> .

Details

Displays a QQ plot of the individual weighted residuals (IWRES).

Value

Returns a QQ plot of individual weighted residuals (IWRES).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.qq](#), [xpose.panel.qq](#), [qqmath](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
```

```
xpdb <- simpraz.xpdb

## A vanilla plot
iwres.dist.qq(xpdb)

## A different plotting character
iwres.dist.qq(xpdb, pch=4)
```

iwres.vs.idv	<i>Individual weighted residuals (IWRES) plotted against the independent variable (IDV) for Xpose 4</i>
--------------	---

Description

This is a plot of individual weighted residuals (IWRES) vs the independent variable (IDV), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
iwres.vs.idv(object,
             abline=c(0,0),
             smooth=TRUE,
             ...)
```

Arguments

object	An <code>xpose.data</code> object.
abline	Vector of arguments to the panel.abline function. No abline is drawn if NULL. Here, the default is <code>c(0,0)</code> , specifying a horizontal line at <code>y=0</code> .
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an xyplot of IWRES vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
iwres.vs.idv(xpdb)

## A conditioning plot
iwres.vs.idv(xpdb, by="HCTZ")

## Logarithmic Y-axis
iwres.vs.idv(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
iwres.vs.idv(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

kaplan.plot

Kaplan-Meier plots of (repeated) time-to-event data

Description

Kaplan-Meier plots of (repeated) time-to-event data. Includes VPCs.

Usage

```
kaplan.plot(x="TIME",y="DV",id="ID",
            data=NULL,
            evid="EVID",
            by=NULL,
            xlab="Time",
            ylab="Default",
            object=NULL,
            events.to.plot="All",
            sim.data=NULL,
            sim.zip.file=NULL,
            VPC = FALSE,
```

```

nsim.lab="simNumber",
sim.evct.lab="counter",
probs=c(0.025,0.975),
add.baseline=T,
add.last.area=T,
subset=NULL,
##subset.real=NULL,
main="Default",
main.sub="Default",
main.sub.cex = 0.8,
nbins=NULL,
real.type="l",
real.lty=1,
real.lwd=1,
real.col="blue",
real.se= if(!is.null(sim.data)) F else T,
real.se.type="l",
real.se.lty=2,
real.se.lwd=0.5,
real.se.col="red",
cens.type="l",
cens.lty=1,
cens.col="black",
cens.lwd=1,
inclZeroWRES=TRUE,
onlyfirst=FALSE,
samp=NULL,
poly.alpha=1,
poly.fill="lightgreen",
poly.line.col="darkgreen",
poly.lty=2,
censor.lines=TRUE,
ylim=c(-5,105),
cov = NULL,
cov.fun = "mean",
...)
```

Arguments

x	The independent variable.
y	The dependent variable. event (>0) or no event (0).
id	The ID variable in the dataset.
data	A dataset can be used instead of the data in an Xpose object. Must have the same form as an xpose data object xpdb@Data.
evid	The EVID data item. If not present then all rows are considered events (can be censored or an event). Otherwise, EVID!=0 are dropped from the data set.
by	A vector of conditioning variables.

xlab	X-axis label
ylab	Y-axis label
object	An Xpose object. Needed if no data is supplied.
events.to.plot	Vector of events to be plotted. "All" means that all events are plotted.
sim.data	The simulated data file. Should be a table file with one header row and have, at least, columns with headers coesponding to x, y, id, by (if used), nsim.lab and sim.evct.lab.
sim.zip.file	The sim.data can be in \.zip format and xpose will unzip the file before reading in the data. Must have the same structure as described above in sim.data.
VPC	TRUE or FALSE. If TRUE then Xpose will search for a zipped file with name paste("simtab",object@Runno,".zip",sep=""), for example "simtab42.zip".
nsim.lab	The column header for sim.data that contains the simulation number for that row in the data.
sim.evct.lab	The column header for sim.data that contains the individual event counter information. For each individual the event counter should increase by one for each event (or censored event) that occurs.
probs	The probabilities (non-parametric percentiles) to use in computation of the prediction intervals for the simulated dats.
add.baseline	Should a (x=0,y=1) baseline measurement be added to each individual in the dataset. Otherwise each plot will begin at the first event in the dataset.
add.last.area	Should an area be added to the VPC extending the last PI?
subset	The subset of the data and sim.data to use.
main	The title of the plot. Can also be NULL or "Default".
main.sub	The title of the subplots. Must be a list, the same length as the number of subplots (actual graphs), or NULL or "Default".
main.sub.cex	The size of the title of the subplots.
nbins	The number of bins to use in the VPC. If NULL, the the number of unique x values in sim.data is used.
real.se	Should the standard errors of the real (non simulated) data be plotted? Calculated using survfit .
real.se.type	Type for the standard errors.
real.type	Type for the real data.
real.lwd	Line width (lwd) for the real data.
real.lty	Line type (lty) for the curve of the original (or real) data.
real.col	Color for the curve of the original (or real) data.
real.se.lty	Line type (lty) for the standard error lines.
real.se.lwd	Line width (lwd) for the standard error lines.
real.se.col	Color for the standard error lines.
cens.type	Type for the censored lines.
cens.lty	Line type (lty) for the censored lines.

cens.col	Color for the censored lines.
cens.lwd	Line width for the censored lines.
cov	The covariate in the dataset to plot instead of the survival curve.
cov.fun	The summary function for the covariate in the dataset to plot instead of the survival curve.
inclZeroWRES	Include WRES=0 rows from the real data set in the plots?
onlyfirst	Include only the first measurement for the real data in the plots?
samp	Simulated data in the xpose data object can be used as the "real" data. samp is a number selecting which simulated data set to use.
poly.alpha	The transparency of the VPC shaded region.
poly.fill	The fill color of the VPC shaded region.
poly.line.col	The line colors for the VPC region.
poly.lty	The line type for the VPC region.
sensor.lines	Should censored observations be marked on the plot?
ylim	Limits for the y-axes
...	Additional arguments passed to the function.

Value

returns an object of class "xpose.multiple.plot".

Author(s)

Andrew C. Hooker

See Also

[survfit](#), [Surv](#), [xpose.multiple.plot](#).

Examples

```
## Not run:
library(xpose4)

## Read in the data
runno <- "57"
xpdb <- xpose.data(runno)

#####
# here are the real data plots
#####

kaplan.plot(x="TIME",y="DV",object=xpdb)
kaplan.plot(x="TIME",y="DV",object=xpdb,
            events.to.plot=c(1,2),
            by=c("DOSE==0","DOSE!=0"))
kaplan.plot(x="TIME",y="DV",object=xpdb,
```



```

        events.to.plot=c(1,2),
        by=c("DOSE==0", "DOSE==10",
            "DOSE==50", "DOSE==200"))

## make a PDF of the plots
pdf(file=paste("run",runno,"_kaplan.pdf", sep=""))
kaplan.plot(x="TIME", y="DV", object=xpdb,
            by=c("DOSE==0", "DOSE==10",
                "DOSE==50", "DOSE==200"))
dev.off()

#####
## VPC plots
#####

kaplan.plot(x="TIME", y="DV", object=xpdb, VPC=T, events.to.plot=c(1))
kaplan.plot(x="TIME", y="DV", object=xpdb, VPC=T,
            events.to.plot=c(1, 2, 3),
            by=c("DOSE==0", "DOSE!=0"))
kaplan.plot(x="TIME", y="DV", object=xpdb, VPC=T,
            events.to.plot=c(1),
            by=c("DOSE==0", "DOSE==10", "DOSE==50", "DOSE==200"))

## make a PDF of all plots
pdf(file=paste("run",runno,"_kaplan.pdf", sep=""))
kaplan.plot(x="TIME", y="DV", object=xpdb, VPC=T,
            by=c("DOSE==0", "DOSE==10", "DOSE==50", "DOSE==200"))
dev.off()

## End(Not run)

```

make.sb.data

Make stacked bar data set.

Description

Function to make stacked bar data set for categorical data plots.

Usage

```
make.sb.data(data, idv, dv, nbins = 6, by = NULL, by.nbins = 6, ...)
```

Arguments

data	Data set to transform.
idv	the independent variable.
dv	the dependent variable.
nbins	the number of bins.
by	Conditioning variable.

```
by.nbins      by.nbins.
...           additional arguments.
```

Author(s)

The Xpose team.

npc.coverage

Function to plot the coverage of the Numerical Predictive Check

Description

This function takes the output from the npc command in Perl Speaks NONMEM (PsN) and makes a coverage plot. A coverage plot for the NPC looks at different prediction intervals (PIs) for each data point and calculates the total number of data points in the data set lying outside of these PIs. The plot shows the relative amount of data points outside of their PI compared to the expected amount at that PI. In addition a confidence interval around these values are computed based on the simulated data.

Usage

```
npc.coverage(npc.info="npc_results.csv",
             main = "Default",
             main.sub=NULL,
             main.sub.cex=0.85,
             ...)
```

Arguments

npc.info	The results file from the npc command in PsN. for example 'npc_results.csv', or if the file is in a separate directory './npc_dir1/npc_results.csv'.
main	A string giving the plot title or NULL if none. "Default" creates a default title.
main.sub	Used for names above each plot when using multiple plots. Should be a vector c("Group 1", "Group 2")
main.sub.cex	The size of the main.sub titles.
...	Other arguments passed to <code>xpose.multiple.plot.default</code> , <code>xyplot</code> and others. Please see these functions for more descriptions of what you can do.

Value

A list of plots

Additional arguments for the NPC coverage plots

Additional plot features

`CI = "both", "area" or "lines"` Specifies whether confidence intervals (as lines, a shaded area or both) should be added to the plot. NULL means no CI.

`mark.outside.data = TRUE or FALSE` Should the points outside the CI be marked in a different color to identify them.

`abline = TRUE` Should there be a line to mark the value of $y=1$? Possible values are TRUE, FALSE and NULL.

Line and area control. See [plot](#), [grid.polygon](#) and [xyplot](#) for more details.

`CI.area.col = "blue"` What color should the area for the CI be? Defaults to "blue".

`CI.area.alpha = 0.3` How much transparency should the `CI.area.col` have? Defaults to 0.3.

`ab.lwd=1` The width of the abline.

`ab.lty="dashed"` How should the abline look?

`CI.upper.lty="dotted"` What should the line at the upper edge of the CI look like when using `CI = "both" or "lines"`?

`CI.upper.col="brown"` What color should the line at the upper edge of the CI have when using `CI = "both" or "lines"`?

`CI.upper.lwd="2"` The line width of the line at the upper edge of the CI when using `CI = "both" or "lines"`?

`CI.lower.lty="dotted"` What should the line at the lower edge of the CI look like when using `CI = "both" or "lines"`?

`CI.lower.col="brown"` What color should the line at the lower edge of the CI have when using `CI = "both" or "lines"`?

`CI.lower.lwd="2"` The line width of the line at the lower edge of the CI when using `CI = "both" or "lines"`?

`obs.col="black"` The color of the observed values.

`obs.pch=19` The type of point to use for the observed values.

`obs.lty="solid"` The type of line to use for the observed values.

`obs.type="b"` The combination of lines and points to use for the observed values.

`obs.cex=1` The size of the points to use for the observed values.

`obs.lwd=1` The line width to use for the observed values.

`out.col="red"` The color of the observed values that lie outside of the CI. Only used if `mark.outside.data = TRUE`.

`out.pch=8` The type of point to use for the observed values that lie outside of the CI. Only used if `mark.outside.data = TRUE`.

`out.cex=1.3` The size of the points of the observed values that lie outside of the CI. Only used if `mark.outside.data = TRUE`.

`out.lwd=1` The line width of the observed values that lie outside of the CI. Only used if `mark.outside.data = TRUE`.

Author(s)

Andrew Hooker

See Also

[read.npc.vpc.results](#) [xpose.multiple.plot.default](#) [xyplot](#)

Examples

```
## Not run:
library(xpose4)

npc.coverage()

## to read files in a directory different than the current working directory
npc.file <- "../another_directory/npc_results.csv"
npc.coverage(npc.info=npc.file)

## End(Not run)
```

nsim

Extract or set the value of the Nsim slot.

Description

Extract or set the value of the Nsim slot of an "xpose.data" object.

Usage

```
nsim(object)
nsim(object) <- value
```

Arguments

object	An "xpose.data" object.
value	A value.

Author(s)

Niclas Jonsson

See Also

[xpose.data-class](#)


```

onlyfirst=TRUE,
smooth = TRUE,
lmline = NULL,
...)
```

Arguments

object	An xpose.data object.
main	A string giving the plot title or NULL if none.
varnames	A vector of strings containing labels for the variables in the scatterplot matrix.
onlyfirst	Logical value indicating if only the first row per individual is included in the plot.
lmline	logical variable specifying whether a linear regression line should be superimposed over an xyplot . NULL ~ FALSE. (y~x)
smooth	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
...	Other arguments passed to xpose.plot.histogram .

Details

The parameters or covariates in the Xpose data object, as specified in `object@Prefs@Xvardef$params`, `object@Prefs@Xvardef$ranpar` or `object@Prefs@Xvardef$covariates`, are plotted together as scatterplot matrices. `parm.splom` delivers parameters, `ranpar.splom` delivers random parameters, and `cov.splom` delivers covariates.

A wide array of extra options controlling scatterplot matrices are available. See [xpose.plot.splom](#) for details.

To control the appearance of the labels and names in the scatterplot matrix plots you can try `varname.cex=0.5` and `axis.text.cex=0.5` (this changes the tick labels and the variable names to be half as large as normal).

Value

Delivers a scatterplot matrix.

Author(s)

Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.splom](#), [xpose.panel.splom](#), [splom](#), [xpose.data-class](#), [xpose.prefs-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A scatterplot matrix of parameters, grouped by sex
parm.splom(xpdb, groups="SEX")

## A scatterplot matrix of ETAs, grouped by sex
ranpar.splom(xpdb, groups="SEX")

## Covariate scatterplots, with text customization
cov.splom(xpdb, varname.cex=0.4, axis.text.cex=0.4, smooth=NULL, cex=0.4)
```

 parm.vs.cov

Parameters plotted against covariates, for Xpose 4

Description

This creates a stack of plots of Bayesian parameter estimates plotted against covariates, and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
parm.vs.cov(object,
             onlyfirst=TRUE,
             smooth=TRUE,
             type="p",
             main="Default",
             ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>onlyfirst</code>	Logical value indicating whether only the first row per individual is included in the plot.
<code>smooth</code>	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.

type	The plot type - defaults to points only.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function <code>xpose.multiple.plot.title</code> is used.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Each of the parameters in the Xpose data object, as specified in `object@Prefs@Xvardef$parms`, is plotted against each covariate present, as specified in `object@Prefs@Xvardef$covariates`, creating a stack of plots.

A wide array of extra options controlling xyplots are available. See `xpose.plot.default` and `xpose.panel.default` for details.

Value

Returns a stack of xyplots and histograms of parameters against covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

`xpose.plot.default`, `xpose.plot.histogram`, `xyplot`, `histogram`, `xpose.prefs-class`, `xpose.data-class`

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb <- xpose.data(5)

## A vanilla plot
parm.vs.cov(xpdb)

## Custom colours and symbols, IDs
parm.vs.cov(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)

## End(Not run)
```

parm.vs.parm *Plot parameters vs other parameters*

Description

This function plots the parameter values stored in an Xpose data object versus each other in a series of graphs. The mirror functionality is available for this function.

Usage

```
parm.vs.parm(object,
              onlyfirst=TRUE,
              abline= FALSE,
              smooth=TRUE,
              type="p",
              main="Default",
              ...)
```

Arguments

object	An xpose.data object.
onlyfirst	Logical value indicating whether only the first row per individual is included in the plot.
abline	Allows for a line of identity.
smooth	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
type	The plot type - defaults to points only.
main	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
...	Other arguments passed to <code>xpose.plot.default</code> .

Details

Each of the parameters in the Xpose data object, as specified in `object@Prefs@Xvardef$parms`, is plotted against the rest, creating a stack of plots.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns a stack of xyplots and histograms of parameters against parameters.

Author(s)

Andrew Hooker

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb <- xpose.data(5)

parm.vs.parm(xpdb)

parm.vs.parm(xpdb,mirror=3)

## End(Not run)
```

pred.vs.idv	<i>Population predictions (PRED) plotted against the independent variable (IDV) for Xpose 4</i>
-------------	---

Description

This is a plot of population predictions (PRED) vs the independent variable (IDV), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
pred.vs.idv(object,
             smooth = TRUE,
             ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>smooth</code>	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an xyplot of PRED vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
pred.vs.idv(xpdb)

## A conditioning plot
pred.vs.idv(xpdb, by="HCTZ")

## Logarithmic Y-axis
pred.vs.idv(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
pred.vs.idv(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

```
print.xpose.multiple.plot
```

Print an Xpose multiple plot object.

Description

Print an Xpose multiple plot object, which is the output from the function [xpose.multiple.plot](#).

Usage

```
## S3 method for class 'xpose.multiple.plot'
print(x, ...)
```

Arguments

x Output object from the function [xpose.multiple.plot](#).
... Additional options passed to function.

Details

Print method for a plot class.

Author(s)

Niclas Jonsson and Andrew C. Hooker

See Also

[xpose.multiple.plot](#).

randtest.hist	<i>Function to create a histogram of results from the randomization test tool (randtest) in R</i> http://psn.sf.net <i>PsN</i>
---------------	--

Description

Reads results from the randtest tool in [PsN](#) and then creates a histogram.

Usage

```
randtest.hist(results.file = "raw_results_run1.csv",
              df = 1, p.val = 0.05, main = "Default",
              xlim = NULL, PCTS1col = "black",
              vlcol = c("red", "orange"), ...)
```

Arguments

results.file	The location of the results file from the randtest tool in PsN
df	The degrees of freedom between the full and reduced model used in the randomization test.
p.val	The p-value you would like to use.
main	The title of the plot.
xlim	The limits of the x-axis
PCTS1col	Color of the empirical line
vlcol	Colors of the original and nominal line
...	Additional arguments that can be passed to xpose.plot.histogram , xpose.panel.histogram , histogram and other lattice-package functions.

Value

Returns a plot object.

Author(s)

Andrew Hooker

References

PsN

See Also[xpose.plot.histogram](#), [xpose.panel.histogram](#), [histogram](#) and other [lattice-package](#) functions.**Examples**

```
## Not run:
randtest.hist(results.file="randtest_dir1/raw_results_run1.csv",df=2)

## End(Not run)
```

ranpar.vs.cov

Random parameters plotted against covariates, for Xpose 4

Description

This creates a stack of plots of Bayesian random parameter estimates plotted against covariates, and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
ranpar.vs.cov(object,
              onlyfirst=TRUE,
              smooth=TRUE,
              type="p",
              main="Default",
              ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>onlyfirst</code>	Logical value indicating whether only the first row per individual is included in the plot.
<code>smooth</code>	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
<code>type</code>	The plot type - defaults to points only.
<code>main</code>	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Each of the random parameters (ETAs) in the Xpose data object, as specified in `object@Prefs@Xvardef$ranpar`, is plotted against each covariate present, as specified in `object@Prefs@Xvardef$covariates`, creating a stack of plots.

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns a stack of xyplots and histograms of random parameters against covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.plot.histogram](#), [xyplot](#), [histogram](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb <- xpose.data(5)

## A vanilla plot
ranpar.vs.cov(xpdb)

## Custom colours and symbols, IDs
ranpar.vs.cov(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)

## End(Not run)
```

Read NONMEM output files

Read NONMEM output files into Xpose 4

Description

These are functions that read in a NONMEM output file (a '*.lst' file) and then format the input for various [xpose4specific-package](#) functions.

Usage

```

fix.wrapped.lines(par.mat)

read.lst(filename)

read.lst(filename)
read.lst7(filename)
read.lst6(filename)

read.phi(phi.file=NULL,
         phi.prefix="run",
         runno=NULL,
         phi.suffix=".phi",
         quiet=TRUE,
         nm7=TRUE,
         directory="",
         ...)

create.parameter.list(listfile)

calc.npar(object)

```

Arguments

par.mat	A parameter matrix.
filename	A NONMEM output file.
phi.file	A NONMEM .phi file
phi.prefix	prefix of a NONMEM .phi file
runno	NONMEM run number
phi.suffix	suffix of a NONMEM .phi file
quiet	Quiet or not
nm7	NM7 or not
directory	directory in which the NONMEM output files is
...	Items passed to functions within this function.
listfile	A NONMEM output file.
object	The return value of read.lst(filename)

Details

These are internal Xpose functions used to read a NONMEM output file.

fix.wrapped.lines unwraps matrix lines that NONMEM wraps in table files. Assumes no more than 60 ETAs.

read.lst parses information out of NONMEM output, specified by the filename argument, for use in run summaries.

calc.npar calculates the number and type of parameters included in a NONMEM output file

Value

Internal helper functions for the generic Xpose functions.

Author(s)

Niclas Jonsson, Andrew Hooker & Justin Wilkins

read.nm.tables	<i>Reading NONMEM table files</i>
----------------	-----------------------------------

Description

Reads one or more NONMEM table files, removes duplicated columns and merges the data into a data.frame.

Usage

```
read.nm.tables(table.files = NULL,
              runno = NULL,
              tab.suffix = "",
              table.names = c("sdtab", "mutab", "patab",
                             "catab", "cotab", "mytab",
                             "extra", "xptab"),
              cwres.name = c("cwtab"),
              cwres.suffix = "",
              quiet = FALSE,
              ...)
```

Arguments

table.files	Exact names of table files to read. If not provided then the exact names are created using the other arguments to the function.
runno	Run-number to identify sets of table files.
tab.suffix	Table file name suffix.
table.names	Vector of template table file names to read.
cwres.name	Vector of CWRES table file names to read.
cwres.suffix	CWRES table file name suffix.
quiet	Logical value to indicate whether some warnings should be quiet or not.
...	Additional arguments passed to this function

Details

Reads one or more table files, removes duplicate columns and merges the data. The function also checks to see if the table files are of the same length (required).

If there are header lines in the table files (for example if your data are simulated with NSUB>1), these are removed.

The table file names to read are constructed from the file name templates of `table.names`. The `runno` and `tab.suffix` are appended to the file name template before checking if the file is readable.

Xpose expects, by default, to find the following NONMEM tables in the working directory to be able to create an Xpose data object (using a run number of 5 as an example):

`sdtab5`: The 'standard' parameters, including IWRE, IPRE, TIME, and the NONMEM default items (DV, PRED, RES and WRES) that are added when NOAPPEND is not present in the `\$TABLE` record.

```
$TABLE ID TIME IPRE IWRE                                NOPRINT ONEHEADER FILE=sdtab5
```

`patab5`: The empirical Bayes estimates of individual model parameter values, or posthoc estimates. These are model parameters, such as CL, V2, ETA1, etc.

```
$TABLE ID CL V2 KA K F1 ETA1 ETA2 ETA3                    NOPRINT NOAPPEND ONEHEADER FILE=patab5
```

`catab5`: Categorical covariates, e.g. SEX, RACE.

```
$TABLE ID SEX HIV GRP                                    NOPRINT NOAPPEND ONEHEADER FILE=catab5
```

`cotab5`: Continuous covariates, e.g. WT, AGE.

```
$TABLE ID WT AGE BSA HT GGT HB                          NOPRINT NOAPPEND ONEHEADER FILE=cotab5
```

`mutab5`, `mytab5`, `extra5`, `xptab5`: Additional variables of any kind. These might be useful if there are more covariates than can be accommodated in the covariates tables, for example, or if you have other variables that should be added, e.g. CMAX, AUC.

Value

A data.frame.

Author(s)

Niclas Jonsson, Andrew Hooker

See Also

[xpose.data-class](#), [compute.cwres](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory, and that the table files have
## a suffix of '.dat', e.g. sdtab5.dat

my.dataframe <- read.nm.tables(5, tab.suffix = ".dat")

## End(Not run)
```

read.npc.vpc.results *Read the results file from a Numerical or Visual Predictive Check run in PsN*

Description

This function reads in a results file from running either the PsN command vpc or npc. The function then parses the file and passes the result to plotting functions.

Usage

```
read.npc.vpc.results(vpc.results = NULL, npc.results = NULL,
                    verbose = FALSE, ...)
```

Arguments

vpc.results	The name of the results file from running the PsN command vpc. Often this is named 'vpc_results.csv'. If the file is in a directory different then the working directory then you can define a relative or absolute path to the file by, for example, './vpc_strat_WT_4_mirror_5/vpc_results.csv'.
npc.results	The name of the results file from running the PsN command npc. Often this is named 'npc_results.csv'. relative or absolute paths to the file are allowed as for vpc.results.
verbose	Text messages passed to screen or not.
...	arguments passed to other functions.

Details

One of vpc.results or npc.results are necessary. If both or none are defined then the function does nothing and a NULL is returned from the function.

Value

A list of values is returned.

model.file	The model file that PsN ran either the npc or vpc with
dv.var	The dependent variable used in the calculations.
idv.var	The independent variable used in the calculations. NULL if npc.results is used.
num.tables	The number of separate tables in the results file.
by.interval	The conditioning interval for the stratification variable, only returned if vpc.results is used.
result.tables	The results tables from the results file. this is a list.

Author(s)

Andrew Hooker

See Also[xpose.VPC npc.coverage](#)

read.TTE.sim.data	<i>Read (repeated) time-to-event simulation data files.</i>
-------------------	---

Description

Read (repeated) time-to-event simulation data files.

Usage

```
read.TTE.sim.data(sim.file,  
                  subset = NULL,  
                  headers = c("REP", "ID", "DV", "TIME", "FLAG2", "DOSE"),  
                  xpose.table.file = FALSE,  
                  ...)
```

Arguments

sim.file	Name of the simulated file.
subset	subset to extract.
headers	headers in file.
xpose.table.file	xpose table files.
...	Extra arguments passed to function.

Author(s)

Andrew C. Hooker

 read.vpctab

Read the vpctab file from PsN into Xpose

Description

This function read in the vpctab file created from PsN and gathers the information needed to make a vpc plot.

Usage

```
read.vpctab(vpctab=NULL,
            object=NULL,
            vpc.name="vpctab",
            vpc.suffix="",
            tab.suffix="",
            inclZeroWRES=FALSE,
            verbose = FALSE,
            ...)
```

Arguments

vpctab	The vpctab file from a 'vpc' run in PsN.
object	An xpose data object. Created from xpose.data . One of object or vpctab is required. If both are present then the information from the vpctab will over-ride the xpose data object object (i.e. the values from the vpctab will replace any matching values in the object@Data portion of the xpose data object). If only object is present then the function will look for a vpctab with the same run number as the one associated with the object.
vpc.name	The default name of the vpctab file. Used if only object is supplied.
vpc.suffix	The suffix of the vpctab file. Used if only object is supplied.
tab.suffix	The table suffix of the vpctab file. Used if only object is supplied. Final order of the file would be then paste(vpc.name, object@Runno, vpc.suffix, tab.suffix)
inclZeroWRES	If there are no zero valued weighted residuals in the object then this should be TRUE.
verbose	Text messages passed to screen or not.
...	Other arguments passed to other functions.

Value

Returned is an xpose data object with vpctab information included.

Author(s)

Andrew Hooker

See Also[xpose.VPC](#)

reset*Resets Xpose variable definitions to factory settings*

Description

Function to reset Xpose's graphics parameters definitions to the default.

Usage

```
reset.graph.par(object,  
                classic = FALSE)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>classic</code>	A logical operator specifying whether the function should assume the classic menu system. This is an internal option and need never be called from the command line.

Details

This functions is used to reset Xpose's graphic settings definitions to their default values. Graphical settings are read from the file 'xpose.ini' in the root of the 'xpose4' package.

Value

An [xpose.data](#) object (`classic == FALSE`) or null (`classic == TRUE`).

Author(s)

Niclas Jonsson & Justin Wilkins

See Also[xpose.prefs-class](#), [import.graph.par](#), [change.xvardef](#)**Examples**

```
## Not run:  
## xpdb5 is an Xpose data object  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
## Import graphics preferences you saved earlier using export.graph.par
```

```
xpdb5 <- import.graph.par(xpdb5)

## Reset to default values
xpdb5 <- reset.graph.par(xpdb5)

## Change WRES definition
xpdb5 <- change.wres(xpdb5)

## End(Not run)
```

runsum

Print run summary in Xpose 4

Description

Function to build Xpose run summaries.

Usage

```
runsum(object,
        dir="",
        modfile=paste(dir,"run",object@Runno,".mod",sep=""),
        listfile=paste(dir,"run",object@Runno,".lst",sep=""),
        main=NULL,
        subset=xsubset(object),
        show.plots=TRUE,
        txt.cex=0.7,
        txt.font=1,
        show.ids=FALSE,
        param.table=TRUE,
        txt.columns=2,
        force.wres=FALSE,
        ...)
```

Arguments

object	An xpose.data object.
dir	The directory to look for the model and output file of a NONMEM run.
modfile	The name of the NONMEM control stream associated with the current run.
listfile	The name of the NONMEM output file associated with the current run.
main	A string giving the main heading. NULL if none.
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
show.plots	Logical indicating if GOF plots should be shown in the run summary.
txt.cex	Number indicating the size of the txt in the run summary.
txt.font	Font of the text in the run summary.
show.ids	Logical indicating if IDs should be plotted in the plots for the run summary.
param.table	Logical indicating if the parameter table should be shown in the run summary.
txt.columns	The number of text columns in the run summary.
force.wres	Plot the WRES even if other residuals are available.
...	Other arguments passed to the various functions.

Value

A compound plot containing an Xpose run summary is created.

Author(s)

Niclas Jonsson and Andrew Hooker

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

runsum(xpdb)
```

simpraz.xpdb

Simulated prazosin Xpose database.

Description

Xpose database from the NONMEM output of a model for prazosin using simulated data (and NONMEM 7.2).

Usage

```
data(simpraz.xpdb)
```

Format

The format is: Formal class 'xpose.data' [package ".GlobalEnv"] with 8 slots ..@ Data : 'data.frame': 640 obs. of 42 variables: ..\$ ID : num [1:640] 1 1 1 1 1 1 1 1 1 1\$ TIME : num [1:640] 0 1 2 3 4 5 6 7 9 11\$ IPRED : num [1:640] 0 69.2 80.2 75.3 66.9\$ IWRES : num [1:640] 0 0.0368 -0.0944 0.1683 -0.206\$ DV : num [1:640] 0 71.7 72.6 88 53.1\$ PRED : num [1:640] 0 86.4 89 75.5 61\$ RES : num [1:640] 0 -14.68 -16.4 12.54 -7.91\$ WRES : num [1:640] 0 -0.105 -0.759 1.208 -1.539\$ CL : num [1:640] 13.6 13.6 13.6 13.6 13.6\$ V : num [1:640] 93.6 93.6 93.6 93.6 93.6\$ KA : num [1:640] 1.22 1.22 1.22 1.22 1.22\$ ETA1 : num [1:640] -0.268 -0.268 -0.268 -0.268 -0.268\$ ETA2 : num [1:640] 0.198 0.198 0.198 0.198\$ ETA3 : num [1:640] -0.164 -0.164 -0.164 -0.164 -0.164\$ SEX : Factor w/ 2 levels "1","2": 2 2 2 2 2 2 2 2 2 2\$ RACE : Factor w/ 3 levels "1","2","3": 2 2 2 2 2 2 2 2 2 2\$ SMOK : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1\$ HCTZ : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2\$ PROP : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2\$ CON : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2\$ OCC : Factor w/ 1 level "0": 1 1 1 1 1 1 1 1 1 1\$ AGE : num [1:640] 55 55 55 55 55 55 55 55 55 55\$ HT : num [1:640] 154 154 154 154 154 154 154 154 154 154\$ WT : num [1:640] 81 81 81 81 81


```

..$ SECR : num [1:640] 1 1 1 1 1 1 1 1 1 1 ... ..$ NPRED : num [1:640] 0 86.4 89 75.5 61 ... ..$
NRES : num [1:640] 0 -14.68 -16.4 12.54 -7.91 ... ..$ NWRES : num [1:640] 0 -0.0997 -0.7782
1.2737 -1.6358 ... ..$ PREDI : num [1:640] 0 86.4 89 75.5 61 ... ..$ RESI : num [1:640] 0 -14.68
-16.4 12.54 -7.91 ... ..$ WRESI : num [1:640] 0 -0.105 -0.759 1.208 -1.539 ... ..$ CPRED :
num [1:640] 0 85.8 91.3 79.3 65.3 ... ..$ CRES : num [1:640] 0 -14.03 -18.67 8.73 -12.16 ... ..
..$ CWRES : num [1:640] 0 -0.0646 -0.9411 1.1911 -1.5154 ... ..$ CPREDI: num [1:640] 0 85.8
91.3 79.3 65.3 ... ..$ CRESI : num [1:640] 0 -14.03 -18.67 8.73 -12.16 ... ..$ CWRESI: num
[1:640] 0 -0.101 -0.825 1.071 -1.504 ... ..$ EPRED : num [1:640] 0 80.9 80.5 69.4 57.7 ... ..$
ERES : num [1:640] 0 -9.14 -7.84 18.63 -4.57 ... ..$ EWRES : num [1:640] 0 -0.162 -0.626 1.505
-1.485 ... ..$ NPDE : num [1:640] 0 -0.00836 -0.54367 1.5548 -1.8339 ... ..$ ECWRES: num
[1:640] 0 -0.156 -0.65 1.457 -1.336 ... ..@ SData : NULL ..@ Data.firstonly : 'data.frame': 64 obs.
of 12 variables: .. ..$ SUBJECT_NO: int [1:64] 1 2 3 4 5 6 7 8 9 10 ... ..$ ID : int [1:64] 1 2
3 4 5 6 7 8 9 10 ... ..$ ETA.1. : num [1:64] -0.2677 -0.7097 -0.4762 0.0996 -0.3529 ... ..$
ETA.2. : num [1:64] 0.198 0.186 0.202 -0.429 0.098 ... ..$ ETA.3. : num [1:64] -0.164 0.737
0.436 0.151 0.524 ... ..$ ETC.1.1. : num [1:64] 0.00412 0.00646 0.00401 0.00321 0.00328 ... ..
..$ ETC.2.1. : num [1:64] -0.002413 -0.002923 -0.001677 0.003449 -0.000594 ... ..$ ETC.2.2.
: num [1:64] 0.00971 0.00622 0.00661 0.00605 0.00691 ... ..$ ETC.3.1. : num [1:64] -0.00947
-0.01828 -0.01274 -0.00658 -0.01295 ... ..$ ETC.3.2. : num [1:64] 0.01757 0.00998 0.01247
0.00237 0.01117 ... ..$ ETC.3.3. : num [1:64] 0.0821 0.3496 0.1956 0.0754 0.2295 ... ..$ OBJ
: num [1:64] 54.04 60.73 9.13 27.62 25.53 ... ..@ SData.firstonly: NULL ..@ Runno : num 5 ..@
Nsim : NULL ..@ Doc : NULL ..@ Prefs :Formal class 'xpose.prefs' [package ".GlobalEnv"] with
8 slots .. ..@ Xvardef :List of 14 .. ..$ id : chr "ID" .. ..$ idlab : chr "ID" .. ..$ idv
: chr "TIME" .. ..$ occ : chr "OCC" .. ..$ dv : chr "DV" .. ..$ pred : chr "PRED" ..
.. ..$ ipred : chr "IPRED" .. ..$ iwres : chr "IWRES" .. ..$ wres : chr "WRES" .. ..
..$ cwres : chr "CWRES" .. ..$ res : chr "RES" .. ..$ parms : chr [1:6] "CL" "V" "KA"
"ETA1" ... ..$ covariates: chr [1:11] "SEX" "RACE" "SMOK" "HCTZ" ... ..$ ranpar
: chr [1:3] "ETA1" "ETA2" "ETA3" .. ..@ Labels :List of 43 .. ..$ OCC : chr "Occasion"
.. ..$ TIME : chr "Time" .. ..$ PRED : chr "Population predictions" .. ..$ IPRED :
chr "Individual predictions" .. ..$ WRES : chr "Weighted residuals" .. ..$ CWRES : chr
"Conditional weighted residuals" .. ..$ IWRES : chr "Individual weighted residuals" .. ..$
DV : chr "Observations" .. ..$ RES : chr "Residuals" .. ..$ CL : chr "Clearance" .. ..$
V : chr "Volume" .. ..$ TAD : chr "Time after dose" .. ..$ ID : chr "ID" .. ..$ KA : chr
"KA" .. ..$ ETA1 : chr "ETA1" .. ..$ ETA2 : chr "ETA2" .. ..$ ETA3 : chr "ETA3" ..
.. ..$ SEX : chr "SEX" .. ..$ RACE : chr "RACE" .. ..$ SMOK : chr "SMOK" .. ..$
HCTZ : chr "HCTZ" .. ..$ PROP : chr "PROP" .. ..$ CON : chr "CON" .. ..$ AGE : chr
"AGE" .. ..$ HT : chr "HT" .. ..$ WT : chr "WT" .. ..$ SECR : chr "SECR" .. ..$
NPRED : chr "NPRED" .. ..$ NRES : chr "NRES" .. ..$ NWRES : chr "NWRES" .. ..
..$ PREDI : chr "PREDI" .. ..$ RESI : chr "RESI" .. ..$ WRESI : chr "WRESI" .. ..$
CPRED : chr "CPRED" .. ..$ CRES : chr "CRES" .. ..$ CPREDI: chr "CPREDI" .. ..$
CRESI : chr "CRESI" .. ..$ CWRESI: chr "CWRESI" .. ..$ EPRED : chr "EPRED" .. ..
..$ ERES : chr "ERES" .. ..$ EWRES : chr "EWRES" .. ..$ NPDE : chr "NPDE" .. ..$
ECWRES: chr "ECWRES" .. ..@ Graph.prefs :List of 102 .. ..$ type : chr "b" .. ..$ pch :
num 1 .. ..$ cex : num 0.8 .. ..$ lty : num 1 .. ..$ lwd : num 1 .. ..$ col : num 4 ..
.. ..$ fill : chr "lightblue" .. ..$ grid : logi FALSE .. ..$ aspect : num 1 .. ..$ condvar :
NULL .. ..$ byordfun : chr "median" .. ..$ ordby : NULL .. ..$ shingnum : num 6 ..
.. ..$ shingol : num 0.5 .. ..$ abline : NULL .. ..$ abllwd : num 1 .. ..$ ablcol : num 1 ..
.. ..$ abllty : num 1 .. ..$ smlwd : num 2 .. ..$ smcol : num 2 .. ..$ smlty : num 1 ..
.. ..$ smspan : num 0.667 .. ..$ smdegr : num 1 .. ..$ lmline : NULL .. ..$ lmlwd : num

```

```

2 .. .. .$ lmcoll : num 2 .. .. .$ lmlty : num 1 .. .. .$ suline : NULL .. .. .$ sulwd : num 2 ..
.. .. .$ sucoll : num 3 .. .. .$ sulty : num 1 .. .. .$ suspan : num 0.667 .. .. .$ sudegr : num 1
.. .. .$ ids : logi FALSE .. .. .$ idsmode : NULL .. .. .$ idsext : num 0.05 .. .. .$ idscex
: num 0.7 .. .. .$ idmdir : chr "both" .. .. .$ dilfrac : num 0.7 .. .. .$ diltype : NULL .. ..
.$ dilci : num 0.95 .. .. .$ PIuplty : num 2 .. .. .$ PIDolty : num 2 .. .. .$ PImelty : num 1
.. .. .$ PIuptyp : chr "I" .. .. .$ PIDotyp : chr "I" .. .. .$ PImety : chr "I" .. .. .$ PIupcol
: chr "black" .. .. .$ PIDocol : chr "black" .. .. .$ PImecol : chr "black" .. .. .$ PIuplwd :
num 2 .. .. .$ PIDolwd : num 2 .. .. .$ PImelwd : num 2 .. .. .$ PIupltyR : num 1 .. .. .$
PIDoltyR : num 1 .. .. .$ PImeltyR : num 2 .. .. .$ PIuptypR : chr "I" .. .. .$ PIDotypR : chr
"I" .. .. .$ PImetyR : chr "I" .. .. .$ PIupcolR : chr "blue" .. .. .$ PIDocolR : chr "blue" ..
.. .. .$ PImecolR : chr "blue" .. .. .$ PIuplwdR : num 2 .. .. .$ PIDolwdR : num 2 .. .. .$
PImelwdR : num 2 .. .. .$ PIupltyM : num 1 .. .. .$ PIDoltyM : num 1 .. .. .$ PImeltyM :
num 2 .. .. .$ PIuptypM : chr "I" .. .. .$ PIDotypM : chr "I" .. .. .$ PImetyM : chr "I" .. ..
.. .. .$ PIupcolM : chr "darkgreen" .. .. .$ PIDocolM : chr "darkgreen" .. .. .$ PImecolM : chr
"darkgreen" .. .. .$ PIuplwdM : num 0.5 .. .. .$ PIDolwdM : num 0.5 .. .. .$ PImelwdM :
num 0.5 .. .. .$ PIarcol : chr "lightgreen" .. .. .$ PIlimits : num [1:2] 0.025 0.975 .. .. .$
bwhoriz : logi FALSE .. .. .$ bwratio : num 1.5 .. .. .$ bwvarwid : logi FALSE .. .. .$
bwdotpch : num 16 .. .. .$ bwdotcol : chr "black" .. .. .$ bwdotcex : num 1 .. .. .$ bwreccol
: chr "blue" .. .. .$ bwreccol : chr "transparent" .. .. .$ bwrecty : num 1 .. .. .$ bwrectwd :
num 1 .. .. .$ bwumbcol : chr "blue" .. .. .$ bwumbly : num 1 .. .. .$ bwumbld : num 1 ..
.. .. .$ bwoutcol : chr "blue" .. .. .$ bwoutcex : num 0.8 .. .. .$ bwoutpch : num 1 .. .. .$
hicol : num 5 .. .. .$ hiborder : num 1 .. .. .$ hilty : num 1 .. .. .$ hilwd : num 1 .. .. .$
[list output truncated] .. .. .$ Miss : num -99 .. .. .$ Cat.levels : num 4 .. .. .$ DV.Cat.levels : num 7
.. .. .$ Subset : NULL .. .. .$ Gam.prefs : List of 21 .. .. .$ onlyfirst : logi TRUE .. .. .$ wts
: logi FALSE .. .. .$ start.mod : NULL .. .. .$ steppit : logi TRUE .. .. .$ disp : NULL .. ..
.. .. .$ nmods : num 3 .. .. .$ smoother1 : num 0 .. .. .$ smoother2 : num 1 .. .. .$ smoother3
: chr "ns" .. .. .$ smoother4 : chr "ns" .. .. .$ arg1 : NULL .. .. .$ arg2 : NULL .. .. .$
arg3 : chr "df=2" .. .. .$ arg4 : chr "df=3" .. .. .$ excl1 : NULL .. .. .$ excl2 : NULL .. ..
.. .. .$ excl3 : NULL .. .. .$ excl4 : NULL .. .. .$ extra : NULL .. .. .$ plot.ids : logi TRUE .. ..
.. .. .$ medianNorm : logi TRUE

```

Details

The database can be used to test functions in Xpose 4. This database is slightly different than the database that is created when reading in the files created by `simprazExample` using `xpose.data`.

See Also

[simprazExample](#)

Examples

```

data(simpraz.xpdb)
str(simpraz.xpdb)

```

simprazExample	<i>Function to create files for the simulated Prazosin example in Xpose</i>
----------------	---

Description

Creates NONMEM data, model and output files for a model of Prazosin using simulated data.

Usage

```
simprazExample(overwrite=FALSE)
```

Arguments

overwrite	Logical. Should the function overwrite files with the same names already in the current working directory?
-----------	--

Details

Creates files in the current working directory named: run1.ext run1.lst run1.mod simpraz.dta xptab1

Author(s)

Niclas Jonsson and Andrew Hooker

Examples

```
simprazExample()
```

summary	<i>Summarize individual parameter values and covariates</i>
---------	---

Description

These functions produce tables, printed to the screen, summarizing the individual parameter values or covariates from a dataset in Xpose 4.

Usage

```
cov.summary(object,  
             onlyfirst=TRUE,  
             subset=xsubset(object),  
             inclZeroWRES=FALSE,  
             out.file=".screen",  
             main="Default",  
             fill = "gray",  
             values.to.use=xvardef("covariates",object),
```

```

        value.name="Covariate",
        ...)

parm.summary(object,
             onlyfirst=TRUE,
             subset=xsubset(object),
             inclZeroWRES=FALSE,
             out.file=".screen",
             main="Default",
             fill = "gray",
             values.to.use=xvardef("parms",object),
             value.name="Parameter",
             max.plots.per.page=1,
             ...)

```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>onlyfirst</code>	Logical value indicating if only the first row per individual is included in the plot.
<code>inclZeroWRES</code>	Logical value indicating whether rows with <code>WRES=0</code> are included in the plot. The default is <code>FALSE</code> .
<code>out.file</code>	Where the results should be output to. Can be <code>".screen"</code> , <code>".ask"</code> , <code>".graph"</code> or a filename in quotes.
<code>subset</code>	A string giving the subset expression to be applied to the data before plotting. See xsubset .
<code>main</code>	The title of the plot. If <code>"Default"</code> then a default title is plotted. Otherwise the value should be a string like <code>"my title"</code> or <code>NULL</code> for no plot title. For <code>"Default"</code> the function xpose.multiple.plot.title is used.
<code>fill</code>	The color to fill the boxes in the table if the table is printed to <code>".graph"</code>
<code>values.to.use</code>	Which values should be summarized
<code>value.name</code>	The name of the values
<code>max.plots.per.page</code>	Maximum plots per page.
<code>...</code>	Other arguments passed to <code>Data</code> and <code>SData</code> .

Value

Returned is the matrix of values from the table. `parm.summary` and `cov.summary` produce summaries of parameters and covariates, respectively. `parm.summary` produces less attractive output but supports mirror functionality.

`parm.summary` and `cov.summary` utilize [print.char.matrix](#) to print the information to the screen.

Author(s)

Andrew Hooker & Justin Wilkins

See Also

[Data](#), [SData](#), [xpose.data-class](#), [print.char.matrix](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

parm.summary(xpdb)

cov.summary(xpdb)
```

tables

Generic table functions for Xpose 4

Description

These are internal table functions relating to the Xpose summary functions.

Details

These are internal Xpose functions for outputting summary tables. They are not intended for direct use.

Value

Internal helper functions for the generic Xpose summary functions.

Author(s)

Niclas Jonsson & Justin Wilkins

tabulate.parameters *Tabulate the population parameter estimates*

Description

This function provides a summary of the model's parameter estimates and precision.

Usage

```
tabulate.parameters(object, prompt=FALSE, outfile = NULL, dir="")
```

Arguments

object	An xpose.data object.
prompt	Ask before printing.
outfile	file to output to (NULL means screen).
dir	Wich directory is the NONMEM output file located. "" means the current working directory getwd().

Value

A table summarizing the parameters and their precision.

Author(s)

Niclas Jonsson, Andrew Hooker & Justin Wilkins

Examples

```
## Not run:  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
tabulate.parameters(xpdb)  
  
## End(Not run)
```

wres.dist.hist	<i>Histogram of weighted residuals (WRES), for Xpose 4</i>
----------------	--

Description

This is a histogram of the distribution of weighted residuals (WRES) in the dataset, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.histogram` function.

Usage

```
wres.dist.hist(object,  
              ...)
```

Arguments

object	An <code>xpose.data</code> object.
...	Other arguments passed to <code>xpose.plot.histogram</code> .

Details

Displays a histogram of the weighted residuals (WRES).

Value

Returns a histogram of weighted residuals (WRES).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.histogram](#), [xpose.panel.histogram](#), [histogram](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
## End(Not run)  
  
## Here we load the example xpose database  
data(simpraz.xpdb)  
xpdb <- simpraz.xpdb  
  
## A vanilla plot  
wres.dist.hist(xpdb)
```

`wres.dist.qq`*Quantile-quantile plot of weighted residuals (WRES), for Xpose 4*

Description

This is a QQ plot of the distribution of weighted residuals (WRES) in the dataset, a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.qq` function.

Usage

```
wres.dist.qq(object,  
             ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.qq}</code> .

Details

Displays a QQ plot of the weighted residuals (WRES).

Value

Returns a QQ plot of weighted residuals (WRES).

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.qq](#), [xpose.panel.qq](#), [qqmath](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
## End(Not run)  
  
## Here we load the example xpose database  
data(simpraz.xpdb)  
xpdb <- simpraz.xpdb  
  
## A vanilla plot  
wres.dist.qq(xpdb)
```



```
## A different plotting character
wres.dist.qq(xpdb, pch=4)
```

wres.vs.cov

Weighted residuals (WRES) plotted against covariates, for Xpose 4

Description

This creates a stack of plots of weighted residuals (WRES) plotted against covariates, and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` and `xpose.plot.histogram` functions. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
wres.vs.cov(object,
            ylb = "WRES",
            smooth = TRUE,
            type = "p",
            main="Default",
            ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>ylb</code>	A string giving the label for the y-axis. NULL if none.
<code>smooth</code>	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
<code>type</code>	1-character string giving the type of plot desired. The following values are possible, for details, see 'plot': "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c") for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
<code>main</code>	The title of the plot. If "Default" then a default title is plotted. Otherwise the value should be a string like "my title" or NULL for no plot title. For "Default" the function xpose.multiple.plot.title is used.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> or <code>link{xpose.plot.histogram}</code> .

Details

Weighted residuals (WRES) are plotted against each covariate present, as specified in `object@Prefs@Xvardef$covariates`, creating a stack of plots.

A wide array of extra options controlling xyplots and histograms are available. See [xpose.plot.default](#) and [xpose.plot.histogram](#) for details.

Value

Returns a stack of xyplots and histograms of CWRES versus covariates.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.plot.histogram](#), [xyplot](#), [histogram](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
wres.vs.cov(xpdb)

## Custom colours and symbols, IDs
wres.vs.cov(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)

## End(Not run)
```

wres.vs.idv

Population weighted residuals (WRES) plotted against the independent variable (IDV) for Xpose 4

Description

This is a plot of population weighted residuals (WRES) vs the independent variable (IDV), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
wres.vs.idv(object,
             abline=c(0,0),
             smooth=TRUE,
             ...)
```

Arguments

object	An xpose.data object.
abline	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
smooth	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
...	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

Weighted residuals (WRES) are plotted against the independent variable, as specified in `object@Prefs@Xvardef$idv`. A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an xyplot of WRES vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xpose.panel.default](#), [xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
wres.vs.idv(xpdb)

## A conditioning plot
wres.vs.idv(xpdb, by="HCTZ")

## Logarithmic Y-axis
wres.vs.idv(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
wres.vs.idv(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

wres.vs.idv.bw	<i>Box-and-whisker plot of weighted residuals vs the independent variable for Xpose 4</i>
----------------	---

Description

This creates a box and whisker plot of weighted residuals (WRES) vs the independent variable (IDV), and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.bw` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
wres.vs.idv.bw(object,  
              ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.bw}</code> .

Details

This creates a box and whisker plot of weighted residuals (WRES) vs the independent variable (IDV), and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.bw` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

A wide array of extra options controlling bwplots are available. See [xpose.plot.bw](#) and [xpose.panel.bw](#) for details.

Value

Returns a stack of box-and-whisker plots of WRES vs IDV.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.bw](#), [xpose.panel.bw](#), [bwplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
wres.vs.idv.bw(xpdb)
```

wres.vs.pred	<i>Population weighted residuals (WRES) plotted against population predictions (PRED) for Xpose 4</i>
--------------	---

Description

This is a plot of population weighted residuals (WRES) vs population predictions (PRED), a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.default` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
wres.vs.pred(object,
              smooth = TRUE,
              abline=c(0,0),
              ...)
```

Arguments

<code>object</code>	An <code>xpose.data</code> object.
<code>smooth</code>	Logical value indicating whether an x-y smooth should be superimposed. The default is TRUE.
<code>abline</code>	Vector of arguments to the panel.abline function. No abline is drawn if NULL.
<code>...</code>	Other arguments passed to <code>link{xpose.plot.default}</code> .

Details

A wide array of extra options controlling xyplots are available. See [xpose.plot.default](#) and [xpose.panel.default](#) for details.

Value

Returns an xyplot of WRES vs PRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.default](#), [xyplot](#), [xpose.prefs-class](#), [compute.cwres](#), [xpose.data-class](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## End(Not run)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

## A vanilla plot
wres.vs.pred(xpdb)

## A conditioning plot
wres.vs.pred(xpdb, by="HCTZ")

## Logarithmic Y-axis
wres.vs.pred(xpdb, logy=TRUE)

## Custom colours and symbols, IDs
wres.vs.pred(xpdb, cex=0.6, pch=3, col=1, ids=TRUE)
```

wres.vs.pred.bw

*Box-and-whisker plot of weighted residuals vs population predictions
for Xpose 4*

Description

This creates a box and whisker plot of weighted residuals (WRES) vs population predictions (PRED), and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.bw` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

Usage

```
wres.vs.pred.bw(object,  
  ...)
```

Arguments

object	An xpose.data object.
...	Other arguments passed to <code>link{xpose.plot.bw}</code> .

Details

This creates a box and whisker plot of weighted residuals (WRES) vs population predictions (PRED), and is a specific function in Xpose 4. It is a wrapper encapsulating arguments to the `xpose.plot.bw` function. Most of the options take their default values from `xpose.data` object but may be overridden by supplying them as arguments.

A wide array of extra options controlling bwplots are available. See [xpose.plot.bw](#) and [xpose.panel.bw](#) for details.

Value

Returns a box-and-whisker plot of WRES vs PRED.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.bw](#), [xpose.panel.bw](#), [bwplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:  
## We expect to find the required NONMEM run and table files for run  
## 5 in the current working directory  
xpdb5 <- xpose.data(5)  
  
## End(Not run)  
  
## Here we load the example xpose database  
data(simpraz.xpdb)  
xpdb <- simpraz.xpdb  
  
## A vanilla plot  
wres.vs.pred.bw(xpdb)
```

xlabel	<i>Extract and set labels for Xpose data items.</i>
--------	---

Description

This function extracts and sets label definitions in Xpose data objects.

Usage

```
xlabel(x, object)
xlabel(object) <- value
```

Arguments

x	Name of the variable to assign a label to.
object	An xpose.data object.
value	A two element vector of which the first element is the name of the variable and the second the label

Details

x should be a string exactly matching the name of a column in the data.frame in the Data slot of an xpose.data object. The name of columns defined through xpose variable definitions (see [xpose.data](#)) can be extracted using the xvardef function and to be used in the xlabel function, e.g. xlabel(xvardef("dv", object), object), which would give the label for the dv variable.

Value

The label of the specified column.

Author(s)

Niclas Jonsson

See Also

[xpose.prefs-class](#), [xvardef](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Display label for dependent variable in the Xpose data object
```



```
xlabel(xvardef("dv", object), object)

## Set label for dependent variable
xlabel(xpdb5) <- c(xvardef("dv", object), "Concentration (mg/L)")

## End(Not run)
```

xp.boot.par.est

*Compare parameter estimates for covariate coefficients***Description**

This function creates a plot of the estimates for covariate coefficients, obtained from the first step (univariate testing) in each scm performed in the bootscm. When normalized for their standard deviation, these plots can be used to compare the strength of the covariate relationship. Coloring is based on the covariate being included in the final model (blue) not being included (red).

Optionally, estimated bias is plotted in the graph (as text). Bias is also shown by the difference in mean of parameter estimates when the covariate is included (blue diamond), as opposed to the mean of all parameter estimates (grey diamond)

Note: For dichotomous covariates, the default PsN implementation is to use the most common covariate value as base, while the effect of the other value, is estimated by a theta. Xpose (bootscm.import) however recalculates the estimated parameters, to the parametrization in which the lowest value of the dichotomous covariate is the base (e.g. 0), and the estimated THETA denotes the proportional change, when the covariate has the other value (e.g. 1).

Usage

```
xp.boot.par.est(bootgam.obj = NULL, sd.norm = TRUE, by.cov.type = FALSE,
abs.values = FALSE, show.data = TRUE, show.means = TRUE, show.bias =
TRUE, dotpch = c(1,19), labels= NULL, pch.mean="|", xlab=NULL,
ylab=NULL, col = c(rgb(.8, .5, .5), rgb(.2,
.2, .7), rgb(.2,.2,.7), rgb(.6,.6,.6)), ...)
```

Arguments

bootgam.obj	The object created using bootscm.import(), which hold the data for plotting.
sd.norm	Perform normalization of the covariate coefficients (default is TRUE). When TRUE, the estimated covariate coefficients will be multiplied by the standard deviation of the specific covariate (both for continuous and categorical covariates).
by.cov.type	Split the plot for continuous and dichotomous covariates. Default is FALSE.
abs.values	Show the covariate coefficient in absolute values. Default is FALSE.
show.data	Show the actual covariate coefficients in the plot. Default is TRUE.
show.means	Show the means of included covariates (blue) and all covariates (grey) in the plot. Default is TRUE.

show.bias	Show estimated bias as text in the plot. Default is TRUE.
dotpch	The character used for plotting.
labels	Custom labels for the parameter-covariate relationships, (character vector)
xlab	Custom x-axis label
ylab	Custom y-axis label
pch.mean	The character used for plotting the mean.
col	The color scheme.
...	Additional plotting arguments may be passed to this function.

Value

No value returned.

Author(s)

Ron Keizer

Examples

```
xp.boot.par.est()
```

xp.boot.par.est.corr *Correlations between covariate coefficients*

Description

This function creates a plot showing the correlations in estimates for covariate coefficients, obtained from the first step (univariate testing) in each scm performed in the bootscm.

Usage

```
xp.boot.par.est.corr(bootgam.obj = NULL, sd.norm = TRUE, by.cov.type =
FALSE, cov.plot = NULL, ask.covs = FALSE, dotpch=19, col = rgb(.2, .2,
.9, .75), ...)
```

Arguments

bootgam.obj	The object created using bootscm.import(), which hold the data for plotting.
sd.norm	Perform normalization of the covariate coefficients (default is TRUE). When TRUE, the estimated covariate coefficients will be multiplied by the standard deviation of the specific covariate (both for continuous and categorical covariates).
by.cov.type	Split the plot for continuous and dichotomous covariates. Default is FALSE.
cov.plot	A character vector which lists the covariates to include in the plot. If none are specified (NULL), all covariate coefficients will be included in the plot.

ask.covs	Ask the user which covariates to include in the plot. Default is FALSE.
dotpch	The character used for plotting.
col	The colors used for plotting.
...	Additional plotting arguments may be passed to this function.

Value

No value returned.

Author(s)

Ron Keizer

Examples

```
## Not run:
xp.boot.par.est.corr(current.bootscm, sd.norm = TRUE,
                    cov.plot = c("CLSEX", "VSEX", "CLWT"))

## End(Not run)
```

xp.distr.mod.size	<i>Plot of model size distribution for a bootgam or bootscm</i>
-------------------	---

Description

This function creates a kernel smoothed plot of the number of covariates included in the final model in each gam/scm in the bootgam/bootscm procedure.

Usage

```
xp.distr.mod.size(bootgam.obj = NULL, boot.type = NULL,
                 main = NULL, bw = 0.5, xlb = NULL, ...)
```

Arguments

bootgam.obj	The bootgam or bootscm object.
boot.type	Either "bootgam" or "bootscm". Default is NULL, which means the user will be asked to make a choice.
main	Plot title.
bw	The smoothing bandwidth to be used for the kernel.
xlb	The x-axis label.
...	Additional plotting parameter may be passed to this function.

Value

A lattice plot object will be returned.

Author(s)

Ron Keizer

xp.dofv.plot	<i>OFV difference (optimism) plot.</i>
--------------	--

Description

A plot of the difference in OFV between final bootscm models and the reference final scm model.

Usage

```
xp.dofv.plot(bootscm.obj = NULL, main = NULL,  
             xlb = "Difference in OFV", ylb = "Density", ...)
```

Arguments

bootscm.obj	The bootgam or bootscm object.
main	Plot title.
xlb	Label for x-axis.
ylb	Label for y-axis.
...	Additional plotting parameters.

Value

A lattice plot object is returned.

Author(s)

Ron Keizer

xp.inc.prob *Inclusion frequency plot*

Description

Plot the inclusion frequencies of covariates in the final models obtained in a bootgam or bootscm. Covariates are ordered by inclusion frequency.

Usage

```
xp.inc.prob(bootgam.obj = NULL, boot.type=NULL, main = NULL,
            col = "#6495ED", xlb = NULL, ylb = "Covariate", ...)
```

Arguments

bootgam.obj	The bootgam or bootscm object.
boot.type	Either "bootgam" or "bootscm". Default is NULL, which means the user will be asked to make a choice.
main	Plot title
col	Color used for the plot.
xlb	Label for x-axis.
ylb	Label for y-axis.
...	Additional plotting parameters.

Value

A lattice plot object will be returned.

Author(s)

Ron Keizer

xp.inc.prob.comb.2 *Inclusion frequency plot for combination of covariates.*

Description

Plot the inclusion frequency of the most common 2-covariate combinations.

Usage

```
xp.inc.prob.comb.2(bootgam.obj = NULL, boot.type=NULL, main = NULL, col
= "#6495ED", xlb = NULL, ylb = "Covariate combination", ...)
```

Arguments

bootgam.obj	The bootgam or bootscm object.
boot.type	Either "bootgam" or "bootscm". Default is NULL, which means the user will be asked to make a choice.
main	Plot title
col	Color used for plot.
xlb	Label for x-axis.
ylb	Label for y-axis.
...	Additional plotting parameters.

Value

A lattice plot object will be returned.

Author(s)

Ron Keizer

xp.inc.stab.cov

Inclusion stability plot

Description

A plot of the inclusion frequency of covariates vs bootgam/bootscm iteration number. This plot can be used to evaluate whether sufficient iterations have been performed.

Usage

```
xp.inc.stab.cov(bootgam.obj = NULL, boot.type=NULL, main = NULL,
                xlb = "Bootstrap replicate number",
                ylb = "Inclusion frequency", ...)
```

Arguments

bootgam.obj	The bootgam or bootscm object.
boot.type	Either "bootgam" or "bootscm". Default is NULL, which means the user will be asked to make a choice.
main	Plot title.
xlb	The label for the x-axis.
ylb	The label for the y-axis.
...	Additional plotting parameters

Value

A lattice plot object is returned.

Author(s)

Ron Keizer

xp.incl.index.cov *Plot of inclusion index of covariates.*

Description

Covariate inclusion indices show the correlation in inclusion of a covariate in the final model in a bootgam or bootscm.

Usage

```
xp.incl.index.cov(bootgam.obj = NULL, boot.type = NULL, main = NULL, xlb
= "Index", ylb = "Covariate", add.ci = FALSE, incl.range = NULL, ...)
```

Arguments

bootgam.obj	The bootgam or bootscm object.
boot.type	Either "bootgam" or "bootscm". Default is NULL, which means the user will be asked to make a choice.
main	Plot title.
xlb	Label for the x-axis.
ylb	Label for th y-axis.
add.ci	Add a confidence interval to the plotted data.
incl.range	Included range
...	Additional plotting information.

Value

A lattice plot object is returned.

Author(s)

Ron Keizer

xp.incl.index.cov.comp

Inclusion index individuals, compare between covariates.

Description

A plot showing the range of inclusion indices for individuals for all covariates. This plot can be used to evaluate whether there were covariates which were more influenced by the constituency of the bootstrapped dataset than others.

Usage

```
xp.incl.index.cov.comp(bootgam.obj = NULL, boot.type = NULL, main = NULL,
  xlb = "Individual inclusion index", ylb = "ID", ...)
```

Arguments

bootgam.obj	A bootgam or bootscm object.
boot.type	Either "bootgam" or "bootscm". Default is NULL, which means the user will be asked to make a choice.
main	The title of the plot.
xlb	The label for the x-axis.
ylb	The label for the y-axis.
...	Additional plotting parameters.

Value

A lattice plot object is returned.

Author(s)

Ron Keizer

xp.incl.index.cov.ind *Individual inclusion index*

Description

This function will generate a plot of individual inclusion indexes for a specific covariate, which can be used to identify influential individuals for inclusion of that covariate. The index for an individual is calculated as the observed number of inclusions of that individual when the specific covariate was included minus the expected number of inclusions (based on the total bootstrap inclusions), divided by expected.

Usage

```
xp.incl.index.cov.ind(bootgam.obj = NULL, boot.type=NULL, cov.name = NULL,
                      main = NULL, ylb = "ID",
                      xlb = "Individual inclusion index", ...)
```

Arguments

bootgam.obj	A bootgam or bootscm object.
boot.type	Either "bootgam" or "bootscm". Default is NULL, which means the user will be asked to make a choice.
cov.name	The name of the covariate for which to create the plot.
main	The title of the plot.
xlb	The label for the x-axis.
ylb	The label for the y-axis.
...	Additional plotting parameters.

Value

A lattice plot object is returned.

Author(s)

Ron Keizer

xpose.ask.for.filename

Function to ask the user for the name of a file

Description

Asks the user for the name of a file.

Usage

```
xpose.ask.for.filename(object,
                      listfile = paste("run",object@Runno,".lst",sep= ""),
                      modfile = paste("run",object@Runno,".mod",sep =""),
                      ...)
```

```
xpose.ask.for.lst(object,
                  listfile = paste("run",object@Runno,".lst",sep = ""),
                  ...)
```

```
xpose.ask.for.mod(object,
                  modfile = paste("run", object@Runno,".mod",sep =""),
                  ...)
```

Arguments

object	An <code>xpose.data</code> object.
listfile	A NONMEM output file
modfile	A NONMEM model file
...	Additional arguments passed to the function

Details

Function checks if the file exists, if it does then the filename is returned from the function.

Value

The name of the file if it exists, otherwise nothing is returned.

Author(s)

Niclas Jonsson, Justin Wilkins, Mats Karlsson and Andrew Hooker

`xpose.create.title` *Functions to create labels for plots*

Description

Functions to create labels for plots

Usage

```
xpose.create.title(x,y,object,subset=NULL,funx=NULL,funy=NULL,
                  no.runno=FALSE,...)
```

```
xpose.create.label(x, object, fun, logx,
                  autocorr.x = FALSE,
                  autocorr.y = FALSE, ...)
```

```
xpose.create.title.hist(x, object, subset, ...)
```

```
xpose.multiple.plot.title(object, plot.text,
                          subset = xsubset(object),
                          main = "Default",
                          no.runno = FALSE,
                          ...)
```

Arguments

x	Column name for x-variable
y	Column name for y variable
object	Xpose data object
subset	Subset used for plot
fun	Function applied to data
funx	Function applied to x data
funy	Function applied to y data
no.runno	should we include a run number in the label
logx	is the data log trasformed?
autocorr.x	Are we looking at an autocorrelation plot?
autocorr.y	Are we looking at an autocorrelation plot?
plot.text	Text of plot title
main	If "Default" then plot.text is the default title with the run number and subset expression added to the title. If the value is anything other than "Default" then that value is used as the plot title.
...	additional arguments passed to the function.

Value

Plot titles and labels.

Author(s)

Andrew Hooker

xpose.create.title.text

Create Xpose title text for plots.

Description

Create Xpose title text for plots.

Usage

```
xpose.create.title.text(x, y, text, object, subset, text2 = NULL, ...)
```

Arguments

x	The x-axis variable name.
y	The y-axis variable name.
text	Initial text in title.
object	Xpose data object <code>xpose.data</code> .
subset	Subset definition.
text2	Text at the end of the title.
...	Additional options passed to function.

Author(s)

Andrew C. Hooker

xpose.data	<i>Creates an Xpose data object</i>
------------	-------------------------------------

Description

Creates an `xpose.data` object.

Usage

```
xpose.data(runno,
           tab.suffix="",
           sim.suffix="sim",
           cwres.suffix="",
           directory="",
           quiet=TRUE,
           table.names=c("sdtab", "mutab", "patab", "catab",
                        "cotab", "mytab", "extra", "xptab", "cwtab"),
           cwres.name=c("cwtab"),
           mod.prefix="run",
           mod.suffix=".mod",
           phi.suffix=".phi",
           phi.file=NULL,
           nm7=NULL,
           ...)
```

Arguments

runno	Run number of the table files to read.
tab.suffix	Suffix to be appended to the table file names for the "real" data.
sim.suffix	Suffix to be appended to the table file names for any simulated data.
cwres.suffix	Suffix to be appended to the table file names for any CWRES data.

directory	Where the files are located.
quiet	A logical value indicating if more diagnostic messages should be printed when running this function.
table.names	Default text that Xpose looks for when searching for table files.
cwres.name	default text that xpose looks for when searching for CWRES table files.
mod.prefix	Start of model file name.
mod.suffix	End of model file name.
phi.suffix	End of \phi file name.
phi.file	The name of the \phi file. If not NULL then supercedes paste(mod.prefix,runno,phi.suffix,sep="")
nm7	T/F if table files are for NONMEM 7/6, NULL for undefined.
...	Extra arguments passed to function.

Details

Xpose expects, by default, to find the following NONMEM tables in the working directory to be able to create an Xpose data object (using a run number of 5 as an example):

sdtab5: The 'standard' parameters, including IWRE, IPRE, TIME, and the NONMEM default items (DV, PRED, RES and WRES) that are added when NOAPPEND is not present in the \$TABLE record.

```
$TABLE ID TIME IPRE IWRE                                NOPRINT ONEHEADER FILE=sdtab5
```

patab5: The empirical Bayes estimates of individual model parameter values, or posthoc estimates. These are model parameters, such as CL, V2, ETA1, etc.

```
$TABLE ID CL V2 KA K F1 ETA1 ETA2 ETA3                    NOPRINT NOAPPEND ONEHEADER FILE=patab5
```

catab5: Categorical covariates, e.g. SEX, RACE.

```
$TABLE ID SEX HIV GRP                                    NOPRINT NOAPPEND ONEHEADER FILE=catab5
```

cotab5: Continuous covariates, e.g. WT, AGE.

```
$TABLE ID WT AGE BSA HT GGT HB                            NOPRINT NOAPPEND ONEHEADER FILE=cotab5
```

mutab5, mytab5, extra5, xptab5: Additional variables of any kind. These might be useful if there are more covariates than can be accommodated in the covariates tables, for example, or if you have other variables that should be added, e.g. CMAX, AUC.

The default names for table files can be changed by changing the default values to the function. The files that Xpose looks for by default are:

```
paste(table.names, runno, tab.suffix, sep="")
```

The default CWRES table file name is called:

```
paste(cwres.name,runno,cwres.suffix,tab.suffix,sep="")
```

If there are simulation files present then Xpose looks for the files to be named:

```
paste(table.names, runno, sim.suffix, tab.suffix, sep="")
paste(cwres.name,runno,sim.suffix,cwres.suffix,tab.suffix,sep="")
```

This is basically a wrapper function for the `read.nm.tables`, `Data` and `SData` functions. See them for further information.

Also reads in the `\phi` file associated with the run (Individual OFVs, parameters, and variances of those parameters.)

Value

An `xpose.data` object.

Author(s)

Niclas Jonsson, Andrew Hooker

See Also

[xpose.data-class](#), [Data](#), [SData](#), [read.nm.tables](#), [compute.cwres](#)

Examples

```
## We expect to find the required NONMEM run and table files for run
## 1 in the current working directory, using default Xpose format
## Here we create files from an example NONMEM run
simprazExample(overwrite=FALSE)
xpdb <- xpose.data(1)

## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory, and that the table files have
## a suffix of '.dat', e.g. sdtab5.dat

xpdb5 <- xpose.data(5, tab.suffix = ".dat")

## End(Not run)
```

xpose.data-class *Class xpose.data*

Description

The `xpose.data` class is the fundamental data object in Xpose 4. It contains the data and preferences used in the creation of the Xpose plots and analyses.

Objects from the Class

Objects are most easily created by the `xpose.data` function, which reads the appropriate NONMEM table files and populates the slots of the object.

Slots

Runno: Object of class "character" or "numeric" The run number identifying the NONMEM run from which the data in the object comes from.

Data: Object of class "data.frame" or "NULL" This slot holds the (merged) data, in the form of a regular data.frame, from the NONMEM table files generated from the NONMEM run. It can be accessed and set using the [Data](#) function.

SData: Object of class "data.frame" or "NULL" A data.frame of any data simulated from the model. It can be accessed and set using the [SData](#) function. The number of rows should be an even multiple of the rows in Data and the column names should be exactly the same as in Data, except that one extra column should be present ("iter"). The extra column consists of integers that identifies the iteration the row belongs to.

Nsim: Object of class "numeric" or "NULL" A number that indicates the number of simulated data sets in SData.

Prefs: Object of class "xpose.prefs" An object that contains a number of data and graphical preferences for the xpose.data object.

Doc: Object of class "character" or "NULL" A short piece of text describing the model.

Methods

None yet.

Note

There is a validation function for the object called `test.xpose.data`.

Author(s)

Niclas Jonsson

See Also

[xpose.data](#), [Data](#), [SData](#) [read.nm.tables](#), [xpose.prefs-class](#)

xpose.dev.new

Create a new graphical device for an Xpose plot.

Description

The function uses the code from `dev.new()`. This is a function to make `dev.new()` back compatible with older versions of R (before 2.8.0).

Usage

```
xpose.dev.new(...)
```

Arguments

... Additional arguments to a new graphical device. see [dev.new](#).

Author(s)

Andrew Hooker

See Also

[dev.new](#).

xpose.gam

Stepwise GAM search for covariates on a parameter (Xpose 4)

Description

Function takes an Xpose object and performs a generalized additive model (GAM) stepwise search for covariates on a single model parameter.

Usage

```
xpose.gam(object,
  parnam = xvardef("parms", object)[1],
  covnams = xvardef("covariates", object),
  trace = TRUE,
  scope = NULL,
  disp = object@Prefs@Gam.prefs$disp,
  start.mod = object@Prefs@Gam.prefs$start.mod,
  family = "gaussian",
  wts.data = object@Data.firstonly,
  wts.col=NULL,
  steppit = object@Prefs@Gam.prefs$steppit,
  subset = xsubset(object),
```



```

onlyfirst = object@Prefs@Gam.prefs$onlyfirst,
medianNorm = object@Prefs@Gam.prefs$medianNorm,
nmods = object@Prefs@Gam.prefs$nmods,
smoother1 = object@Prefs@Gam.prefs$smoother1,
smoother2 = object@Prefs@Gam.prefs$smoother2,
smoother3 = object@Prefs@Gam.prefs$smoother3,
smoother4 = object@Prefs@Gam.prefs$smoother4,
arg1 = object@Prefs@Gam.prefs$arg1,
arg2 = object@Prefs@Gam.prefs$arg2,
arg3 = object@Prefs@Gam.prefs$arg3,
arg4 = object@Prefs@Gam.prefs$arg4,
excl1 = object@Prefs@Gam.prefs$excl1,
excl2 = object@Prefs@Gam.prefs$excl2,
excl3 = object@Prefs@Gam.prefs$excl3,
excl4 = object@Prefs@Gam.prefs$excl4,
extra = object@Prefs@Gam.prefs$extra,
...)
```

```

xp.get.disp(gamdata,
            parnam,
            covnams,
            family = "gaussian",
            ...)
```

Arguments

object	An xpose.data object.
parnam	ONE (and only one) model parameter name.
covnams	Covariate names to test on parameter.
trace	TRUE if you want GAM output to screen.
scope	Scope of the GAM search.
disp	If dispersion should be used in the GAM object.
start.mod	Starting model.
family	Assumption for the parameter distribution.
wts.data	Weights on the least squares fitting of parameter vs. covariate. Often one can use the variances of the individual parameter values as weights. This data frame must have column with name ID and any subset variable as well as the variable defined by the wts.col.
wts.col	Which column in the wts.data to use.
steppit	TRUE for stepwise search, false for no search.
subset	Subset on data.
onlyfirst	TRUE if only the first row of each individual's data is to be used.
medianNorm	Normalize to the median of parameter and covariates.
nmods	Number of models to examine.

smoother1	Smoother for each model.
smoother2	Smoother for each model.
smoother3	Smoother for each model.
smoother4	Smoother for each model.
arg1	Argument for model 1.
arg2	Argument for model 2.
arg3	Argument for model 3.
arg4	Argument for model 4.
excl1	Covariate exclusion from model 1.
excl2	Covariate exclusion from model 2.
excl3	Covariate exclusion from model 3.
excl4	Covariate exclusion from model 4.
extra	Extra exclusion criteria.
gamdata	Data for the GAM. A data frame.
...	Used to pass arguments to more basic functions.

Details

xpose.gam performs a stepwise GAM search for influential covariates. xp.get.disp is a helper function for calculating dispersion, and is not intended to be used by itself.

Value

Returned is a [step.gam](#) object

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xp.gam](#), [step.gam](#)

Examples

```
## Not run:
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb <- xpose.data(5)

## Here we load the example xpose database
data(simpraz.xpdb)
xpdb <- simpraz.xpdb

xpose.gam(xpdb, parnam="CL", covnams = xvardef("covariates", xpdb))

## End(Not run)
```

```
xpose.license.citation
```

Displays the Xpose license and citation information

Description

This function displays a copy of Xpose's end user license agreement (EULA).

Usage

```
xpose.license.citation()
```

Value

The EULA.

Author(s)

Andrew Hooker

Examples

```
xpose.license.citation()
```

```
xpose.multiple.plot
```

Create and object with class "xpose.multiple.plot".

Description

Create and object with class "xpose.multiple.plot".

Usage

```
xpose.multiple.plot(plotList,  
  plotTitle=NULL,  
  nm7 = TRUE,  
  prompt=FALSE,  
  new.first.window=FALSE,  
  max.plots.per.page=4,  
  title = list(  
    title.x = unit(0.5, "npc"),  
    title.y = unit(0.5, "npc"),  
    title.gp= gpar(cex=1.2, fontface="bold"), #, font=2),  
    title.just = c("center", "center")  
  ),  
  mirror=FALSE,
```

```
bql.layout=FALSE,
...)
```

Arguments

plotList	A list of lattice plots.
plotTitle	Main title for plots.
nm7	TRUE if we are using NONMEM 7
prompt	When printing should we prompt for each new page in plot?
new.first.window	TRUE or FALSE.
max.plots.per.page	A number. Max value is 9.
title	Title properties.
mirror	Are there mirror plots in plot list?
bql.layout	Should we use layout optimized for plots with BQL (below limit of quantification) measurements?
...	Additional options passed to function.

Value

An object of class "xpose.multiple.plot".

Author(s)

Niclas Jonsson and Andrew C. Hooker

See Also

[print.xpose.multiple.plot](#), [xpose.multiple.plot.default](#)

Examples

```
## Not run:
## See code for cwres.vs.cov() by typing

library(xpose4)
cwres.vs.cov

## End(Not run)
```

```
xpose.multiple.plot.default
      Xpose 4 generic function for plotting multiple lattice objects on one
      page
```

Description

Function takes a list of **lattice** plot objects and prints them in a multiple plot layout with a title.

Usage

```
xpose.multiple.plot.default(plotList,
                             plotTitle=NULL,
                             prompt=FALSE,
                             new.first.window=FALSE,
                             max.plots.per.page=4, # absolute max is 9
                             title = list(
                               title.x = unit(0.5, "npc"),
                               title.y = unit(0.5, "npc"),
                               title.gp= gpar(cex=1.2, fontface="bold"),
                               title.just = c("center", "center")
                             ),
                             mirror=FALSE,
                             bql.layout=FALSE,
                             page.numbers=TRUE,
                             ...)
```

Arguments

plotList	A list of lattice plot objects such that plot object <i>i</i> can be called with plotList[[<i>i</i>]]
plotTitle	The title used for the multiple plot layout
prompt	If more than one page is needed do you want a prompt at the command line before the next page is printed
new.first.window	Should the first page of this plot be in the already opened window or should a new window be created
max.plots.per.page	Maximum number of plots per page in the multiple layout
title	Look of title using grid .
mirror	if the list contains mirror plots
bql.layout	should we use layout optimized for BQL measurements?
page.numbers	Should we add page numbers to multiple page plots?
...	Other arguments passed to the code in this function

Details

Additional arguments:

title.x Where the title should be placed in the title **grid** region

title.y Where the title should be placed in the title **grid** region

title.just how the title should be justified

title.gp The par parameters for the title (see **grid**)

Value

returns nothing

Author(s)

Andrew Hooker

See Also

grid, [basic.gof](#), [parm.vs.parm](#), [parm.vs.cov](#),

xpose.panel.bw

Default box-and-whisker panel function for Xpose 4

Description

This is the box-and-whisker panel function for Xpose 4. This is not intended to be used outside the `xpose.plot.bw` function. Most of the arguments take their default values from `xpose.data` object but this can be overridden by supplying them as arguments to `xpose.plot.bw`.

Usage

```
xpose.panel.bw(x, y, object,
               subscripts,
               groups = NULL,
               inclZeroWRES = FALSE,
               onlyfirst = FALSE,
               samp = NULL,
               xvarnam = NULL,
               yvarnam = NULL,
               type = object@Prefs@Graph.prefs$type,
               col = object@Prefs@Graph.prefs$col,
               pch = object@Prefs@Graph.prefs$pch,
               cex = object@Prefs@Graph.prefs$cex,
               lty = object@Prefs@Graph.prefs$lty,
               fill = object@Prefs@Graph.prefs$col,
               ids = NULL,
```

```

idsmode = object@Prefs@Graph.prefs$idsmode,
idsext = object@Prefs@Graph.prefs$idsext,
idsceX = object@Prefs@Graph.prefs$idsceX,
idsdir = object@Prefs@Graph.prefs$idsdir,
bwhoriz = object@Prefs@Graph.prefs$bwhoriz,
bwratio = object@Prefs@Graph.prefs$bwratio,
bwvarwid = object@Prefs@Graph.prefs$bwvarwid,
bwdotpch = object@Prefs@Graph.prefs$bwdotpch,
bwdotcol = object@Prefs@Graph.prefs$bwdotcol,
bwdotcex = object@Prefs@Graph.prefs$bwdotcex,
bwreccol = object@Prefs@Graph.prefs$bwreccol,
bwrecfill = object@Prefs@Graph.prefs$bwrecfill,
bwreclty = object@Prefs@Graph.prefs$bwreclty,
bwreclwd = object@Prefs@Graph.prefs$bwreclwd,
bwumbcol = object@Prefs@Graph.prefs$bwumbcol,
bwumbly = object@Prefs@Graph.prefs$bwumbly,
bwumbld = object@Prefs@Graph.prefs$bwumbld,
bwoutcol = object@Prefs@Graph.prefs$bwoutcol,
bwoutcex = object@Prefs@Graph.prefs$bwoutcex,
bwoutpch = object@Prefs@Graph.prefs$bwoutpch,
grid = object@Prefs@Graph.prefs$grid,
logy = FALSE,
logx = FALSE,
force.x.continuous = TRUE,
binvar = NULL,
bins = 10,
...)
```

Arguments

x	Name(s) of the x-variable.
y	Name(s) of the y-variable.
object	An xpose.data object.
subscripts	The standard Trellis subscripts argument (see xyplot).
groups	Name of the variable used for superpose plots.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot.
onlyfirst	Logical value indicating whether only the first row per individual is included in the plot.
samp	An integer between 1 and object@Nsim (see xpose.data-class) specifying which of the simulated data sets to extract from SData.
xvarnam	Character string with the name of the x-variable.
yvarnam	Character string with the name of the y-variable.
type	Character value indicating the type of display to use: "l"=lines, "p"=points, "b"=both points and lines.
col	Colour of lines and plot symbols.

pch	Plot character to use.
cex	Size of the plot characters.
lty	Line type.
fill	Fill colour.
ids	Character value with the name of the variable to label data points with.
idsmode	Determines the way text labels are added to plots. NULL means that only extreme points are labelled. Non-NULL means all data points are labelled. (See link{xpose.plot.default})
idsxt	See link{xpose.plot.bw}
idscex	Size of text labels.
idsdir	A value of "both" (the default) means that both high and low extreme points are labelled while "up" and "down" labels the high and low extreme points respectively. See xpose.plot.bw
bwhoriz	logical value indicating whether box and whiskers should be horizontal or not. The default is FALSE.
bwratio	Ratio of box height to inter-box space. The default is 1.5. An argument for panel.bwplot .
bwvarwid	Logical. If TRUE, widths of boxplots are proportional to the number of points used in creating it. The default is FALSE. An argument for panel.bwplot .
bwdotpch	Graphical parameter controlling the dot plotting character 'bwdotpch=" "' is treated specially, by replacing the dot with a line. The default is 16. An argument for panel.bwplot .
bwdotcol	Graphical parameter controlling the dot colour - an integer or string. See 'col'. The default is black. An argument for panel.bwplot .
bwdotcex	The amount by which plotting text and symbols should be scaled relative to the default. 'NULL' and 'NA' are equivalent to '1.0'. An argument for panel.bwplot .
bwreccol	The colour to use for the box rectangle - an integer or string. The default is blue. See trellis.par.get and "box.rectangle".
bwrecfill	The colour to use for filling the box rectangle - an integer or string. The default is transparent (none). See trellis.par.get and "box.rectangle".
bwrectly	The line type for the box rectangle - an integer or string. The default is solid. See trellis.par.get and "box.rectangle".
bwreclwd	The width of the lines for the box rectangle - an integer. The default is 1. See trellis.par.get and "box.rectangle".
bwumbcol	The colour to use for the umbrellas - an integer or string. The default is blue. See trellis.par.get and "box.umbrella".
bwumbly	The line type for the umbrellas - an integer or string. The default is solid. See trellis.par.get and "box.umbrella".
bwumbld	the width of the lines for the umbrellas - an integer. The default is 1. See trellis.par.get and "box.umbrella".
bwoutcol	The colour to use for the outliers - an integer or string. The default is blue. See trellis.par.get and "box.symbol".

bwoutcex	The amount by which outlier points should be scaled relative to the default. 'NULL' and 'NA' are equivalent to '1.0'. The default is 0.8. See trellis.par.get and "box.symbol".
bwoutpch	The plotting character, or symbol, to use for outlier points. Specified as an integer. See R help on 'points'. The default is an open circle. See trellis.par.get and "box.symbol".
grid	logical value indicating whether a visual reference grid should be added to the graph. (Could use arguments for line type, color etc).
logy	Logical value indicating whether the y-axis should be logarithmic.
logx	Logical value indicating whether the x-axis should be logarithmic.
force.x.continuous	Logical value indicating whether x-values should be taken as continuous, even if categorical.
binvar	Variable to be used for binning.
bins	The number of bins to be used. The default is 10.
...	Other arguments that may be needed in the function.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.data-class](#), Cross-references above.

xpose.panel.default *Default panel function for Xpose 4*

Description

This is the panel function for Xpose 4. This is not intended to be used outside the `xpose.plot.default` function. Most of the arguments take their default values from `xpose.data` object but this can be overridden by supplying them as argument to `xpose.plot.default`.

Usage

```
xpose.panel.default(x, y, object,
  subscripts,
  groups = object@Prefs@Xvardef$id,
  grp.col = NULL,
  iplot = NULL,
  inclZeroWRES = FALSE,
  onlyfirst = FALSE,
  samp = NULL,
```

```

xvarnam = NULL,
yvarnam = NULL,

#####
## Prediction interval settings
#####
PI      = NULL,
PI.subset=NULL,
PI.bin.table=NULL,
PI.real=NULL, # can be NULL/TRUE
PI.mirror=NULL,
PI.ci = NULL,
PPI = NULL,
PI.mean = FALSE, # Should the mean be plotted in the VPCs
PI.delta.mean = FALSE, # Should the delta.mean be plotted in the VPCs

PI.limits= c(0.025, 0.975),#object@Prefs@Graph.prefs$PI.limits,

PI.arcol = "lightgreen",#object@Prefs@Graph.prefs$PI.arcol,

PI.up.lty = 2,#object@Prefs@Graph.prefs$PI.up.lty,
PI.up.type = "l",#object@Prefs@Graph.prefs$PI.up.type,
PI.up.col = "black",#object@Prefs@Graph.prefs$PI.up.col,
PI.up.lwd = 2,#object@Prefs@Graph.prefs$PI.up.lwd,

PI.down.lty = 2,#object@Prefs@Graph.prefs$PI.down.lty,
PI.down.type = "l",#object@Prefs@Graph.prefs$PI.down.type,
PI.down.col = "black",#object@Prefs@Graph.prefs$PI.down.col,
PI.down.lwd = 2,#object@Prefs@Graph.prefs$PI.down.lwd,

PI.med.lty = 1,#object@Prefs@Graph.prefs$PI.med.lty,
PI.med.type = "l",#object@Prefs@Graph.prefs$PI.med.type,
PI.med.col = "black",#object@Prefs@Graph.prefs$PI.med.col,
PI.med.lwd = 2,#object@Prefs@Graph.prefs$PI.med.lwd,

PI.mean.lty = 3,#object@Prefs@Graph.prefs$PI.med.lty,
PI.mean.type = "l",#object@Prefs@Graph.prefs$PI.med.type,
PI.mean.col = "black",#object@Prefs@Graph.prefs$PI.med.col,
PI.mean.lwd = 2,#object@Prefs@Graph.prefs$PI.med.lwd,

PI.delta.mean.lty = 3,#object@Prefs@Graph.prefs$PI.med.lty,
PI.delta.mean.type = "l",#object@Prefs@Graph.prefs$PI.med.type,
PI.delta.mean.col = "black",#object@Prefs@Graph.prefs$PI.med.col,
PI.delta.mean.lwd = 2,#object@Prefs@Graph.prefs$PI.med.lwd,

PI.real.up.lty = 2,#object@Prefs@Graph.prefs$PI.real.up.lty,
PI.real.up.type = "l",#object@Prefs@Graph.prefs$PI.real.up.type,

```

```
PI.real.up.col = "red",#object@Prefs@Graph.prefs$PI.real.up.col,  
PI.real.up.lwd = 2,#object@Prefs@Graph.prefs$PI.real.up.lwd,
```

```
PI.real.down.lty = 2,#object@Prefs@Graph.prefs$PI.real.down.lty,  
PI.real.down.type = "1",#object@Prefs@Graph.prefs$PI.real.down.type,  
PI.real.down.col = "red",#object@Prefs@Graph.prefs$PI.real.down.col,  
PI.real.down.lwd = 2,#object@Prefs@Graph.prefs$PI.real.down.lwd,
```

```
PI.real.med.lty = 1,#object@Prefs@Graph.prefs$PI.real.med.lty,  
PI.real.med.type = "1",#object@Prefs@Graph.prefs$PI.real.med.type,  
PI.real.med.col = "red",#object@Prefs@Graph.prefs$PI.real.med.col,  
PI.real.med.lwd = 2,#object@Prefs@Graph.prefs$PI.real.med.lwd,
```

```
PI.real.mean.lty = 3,#object@Prefs@Graph.prefs$PI.real.med.lty,  
PI.real.mean.type = "1",#object@Prefs@Graph.prefs$PI.real.med.type,  
PI.real.mean.col = "red",#object@Prefs@Graph.prefs$PI.real.med.col,  
PI.real.mean.lwd = 2,#object@Prefs@Graph.prefs$PI.real.med.lwd,
```

```
PI.real.delta.mean.lty = 3,#object@Prefs@Graph.prefs$PI.real.med.lty,  
PI.real.delta.mean.type = "1",#object@Prefs@Graph.prefs$PI.real.med.type,  
PI.real.delta.mean.col = "red",#object@Prefs@Graph.prefs$PI.real.med.col,  
PI.real.delta.mean.lwd = 2,#object@Prefs@Graph.prefs$PI.real.med.lwd,
```

```
PI.mirror.up.lty = 2,#object@Prefs@Graph.prefs$PI.mirror.up.lty,  
PI.mirror.up.type = "1",#object@Prefs@Graph.prefs$PI.mirror.up.type,  
PI.mirror.up.col = "darkgreen",#object@Prefs@Graph.prefs$PI.mirror.up.col,  
PI.mirror.up.lwd = 1,#object@Prefs@Graph.prefs$PI.mirror.up.lwd,
```

```
PI.mirror.down.lty = 2,#object@Prefs@Graph.prefs$PI.mirror.down.lty,  
PI.mirror.down.type = "1",#object@Prefs@Graph.prefs$PI.mirror.down.type,  
PI.mirror.down.col = "darkgreen",#object@Prefs@Graph.prefs$PI.mirror.down.col,  
PI.mirror.down.lwd = 1,#object@Prefs@Graph.prefs$PI.mirror.down.lwd,
```

```
PI.mirror.med.lty = 1,#object@Prefs@Graph.prefs$PI.mirror.med.lty,  
PI.mirror.med.type = "1",#object@Prefs@Graph.prefs$PI.mirror.med.type,  
PI.mirror.med.col = "darkgreen",#object@Prefs@Graph.prefs$PI.mirror.med.col,  
PI.mirror.med.lwd = 1,#object@Prefs@Graph.prefs$PI.mirror.med.lwd,
```

```
PI.mirror.mean.lty = 3,#object@Prefs@Graph.prefs$PI.mirror.med.lty,  
PI.mirror.mean.type = "1",#object@Prefs@Graph.prefs$PI.mirror.med.type,  
PI.mirror.mean.col = "darkgreen",#object@Prefs@Graph.prefs$PI.mirror.med.col,  
PI.mirror.mean.lwd = 1,#object@Prefs@Graph.prefs$PI.mirror.med.lwd,
```

```
PI.mirror.delta.mean.lty = 3,#object@Prefs@Graph.prefs$PI.mirror.med.lty,  
PI.mirror.delta.mean.type = "1",#object@Prefs@Graph.prefs$PI.mirror.med.type,  
PI.mirror.delta.mean.col = "darkgreen",  
PI.mirror.delta.mean.lwd = 1,#object@Prefs@Graph.prefs$PI.mirror.med.lwd,
```

```

PI.ci.up.arcol = "blue",
PI.ci.up.lty = 3,#object@Prefs@Graph.prefs$PIuplty,
PI.ci.up.type = "l",#object@Prefs@Graph.prefs$PIuptyp,
PI.ci.up.col = "darkorange",#object@Prefs@Graph.prefs$PI.up.col,
PI.ci.up.lwd = 2,#object@Prefs@Graph.prefs$PI.up.lwd,

PI.ci.down.arcol = "blue",
PI.ci.down.lty = 3,#object@Prefs@Graph.prefs$PIdolty,
PI.ci.down.type = "l",#object@Prefs@Graph.prefs$PIdotyp,
PI.ci.down.col = "darkorange",#object@Prefs@Graph.prefs$PI.down.col,
PI.ci.down.lwd = 2,#object@Prefs@Graph.prefs$PI.down.lwd,

PI.ci.med.arcol = "blue",
PI.ci.med.lty = 4,#object@Prefs@Graph.prefs$PImelty,
PI.ci.med.type = "l",#object@Prefs@Graph.prefs$PImetyp,
PI.ci.med.col = "darkorange",#object@Prefs@Graph.prefs$PI.med.col,
PI.ci.med.lwd = 2,#object@Prefs@Graph.prefs$PI.med.lwd,

PI.ci.mean.arcol = "purple",
PI.ci.mean.lty = 4,#object@Prefs@Graph.prefs$PImelty,
PI.ci.mean.type = "l",#object@Prefs@Graph.prefs$PImetyp,
PI.ci.mean.col = "darkorange",#object@Prefs@Graph.prefs$PI.med.col,
PI.ci.mean.lwd = 2,#object@Prefs@Graph.prefs$PI.med.lwd,

PI.ci.delta.mean.arcol = "purple",
PI.ci.delta.mean.lty = 4,#object@Prefs@Graph.prefs$PImelty,
PI.ci.delta.mean.type = "l",#object@Prefs@Graph.prefs$PImetyp,
PI.ci.delta.mean.col = "darkorange",#object@Prefs@Graph.prefs$PI.med.col,
PI.ci.delta.mean.lwd = 2,#object@Prefs@Graph.prefs$PI.med.lwd,

PI.ci.area.smooth=FALSE,
#####
## end of PI settings
#####

## Basic plot characteristics
type = object@Prefs@Graph.prefs$type,
col = object@Prefs@Graph.prefs$col,
pch = object@Prefs@Graph.prefs$pch,
cex = object@Prefs@Graph.prefs$cex,
lty = object@Prefs@Graph.prefs$lty,
lwd = object@Prefs@Graph.prefs$lwd,
fill = object@Prefs@Graph.prefs$fill,

## Text label setting
ids = NULL,
idsmode=object@Prefs@Graph.prefs$idsmode,

```

```
idsext =object@Prefs@Graph.prefs$idsext,
idsceX= object@Prefs@Graph.prefs$idsceX,
idsdir= object@Prefs@Graph.prefs$idsdir,

## abline settings
abline= object@Prefs@Graph.prefs$abline,
abllwd= object@Prefs@Graph.prefs$abllwd,
ablly= object@Prefs@Graph.prefs$ablly,
ablcol= object@Prefs@Graph.prefs$ablcol,

smooth= object@Prefs@Graph.prefs$smooth,
smlwd = object@Prefs@Graph.prefs$smlwd,
smlty = object@Prefs@Graph.prefs$smlty,
smcol = object@Prefs@Graph.prefs$smcol,
smspan= object@Prefs@Graph.prefs$smspan,
smdegr= object@Prefs@Graph.prefs$smdegr,
smooth.for.groups=NULL,

lmline= object@Prefs@Graph.prefs$lmline,
lmlwd = object@Prefs@Graph.prefs$lmlwd ,
lmlty = object@Prefs@Graph.prefs$lmlty ,
lmcol = object@Prefs@Graph.prefs$lmcol ,

suline = object@Prefs@Graph.prefs$suline,
sulwd = object@Prefs@Graph.prefs$sulwd ,
sulity = object@Prefs@Graph.prefs$sulity ,
sucol = object@Prefs@Graph.prefs$sucol ,
suspan = object@Prefs@Graph.prefs$suspan,
sudegr = object@Prefs@Graph.prefs$sudegr,

## Layout parameters
grid = object@Prefs@Graph.prefs$grid,
logy = FALSE,
logx = FALSE,

## Force x variables to be continuous
force.x.continuous = FALSE,

## Categorcal x-variable
bwhoriz = object@Prefs@Graph.prefs$bwhoriz,
bwratio = object@Prefs@Graph.prefs$bwratio,
bwvarwid = object@Prefs@Graph.prefs$bwvarwid,
bwdotpch = object@Prefs@Graph.prefs$bwdotpch,
bwdotcol = object@Prefs@Graph.prefs$bwdotcol,
bwdotcex = object@Prefs@Graph.prefs$bwdotcex,
bwreccol = object@Prefs@Graph.prefs$bwreccol,
bwrecfill= object@Prefs@Graph.prefs$bwrecfill,
bwreclty = object@Prefs@Graph.prefs$bwreclty,
```

```

bwreclwd = object@Prefs@Graph.prefs$bwreclwd,
bwumbcol = object@Prefs@Graph.prefs$bwumbcol,
bwumbly = object@Prefs@Graph.prefs$bwumbly,
bwumbld = object@Prefs@Graph.prefs$bwumbld,
bwoutcol = object@Prefs@Graph.prefs$bwoutcol,
bwoutcex = object@Prefs@Graph.prefs$bwoutcex,
bwoutpch = object@Prefs@Graph.prefs$bwoutpch,
autocorr=FALSE,

## vline settings
vline= NULL,#object@Prefs@Graph.prefs$abline,
vllwd= 3,#object@Prefs@Graph.prefs$abllwd,
vllty= 2,#object@Prefs@Graph.prefs$abllty,
vlcol= "grey",#object@Prefs@Graph.prefs$ablcol,

## hline settings
hline= NULL,#object@Prefs@Graph.prefs$abline,
hllwd= 3,#object@Prefs@Graph.prefs$abllwd,
hllty= 1,#object@Prefs@Graph.prefs$abllty,
hlcol= "grey",#object@Prefs@Graph.prefs$ablcol,

# ind.plots/group panels with single point per panel/group
pch.ip.sp=pch,
cex.ip.sp=cex,

...)
```

Arguments

x	Name(s) of the x-variable.
y	Name(s) of the y-variable.
object	An xpose.data object.
subscripts	The standard Trellis subscripts argument (see xyplot)
groups	Name of the variable used for superpose plots.
grp.col	Logical value indicating whether or not to use colour highlighting when groups are specified. NULL means no highlighting, while TRUE will identify group members by colour.
iplot	Is this an individual plots matrix? Internal use only.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot.
onlyfirst	Logical value indicating whether only the first row per individual is included in the plot.
samp	An integer between 1 and object@Nsim (see xpose.data-class) specifying which of the simulated data sets to extract from SData.
xvarnam	Character string with the name of the x-variable.
yvarnam	Character string with the name of the y-variable.

type	1-character string giving the type of plot desired. The following values are possible, for details, see 'plot': "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c" for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
col	The color for lines and points. Specified as an integer or a text string. A full list is obtained by the R command <code>colours()</code> . The default is blue (<code>col=4</code>).
pch	The plotting character, or symbol, to use. Specified as an integer. See R help on points . The default is an open circle.
cex	The amount by which plotting text and symbols should be scaled relative to the default. 'NULL' and 'NA' are equivalent to '1.0'.
lty	The line type. Line types can either be specified as an integer (0=blank, 1=solid, 2=dashed, 3=dotted, 4=dotdash, 5=longdash, 6=twodash) or as one of the character strings "blank", "solid", "dashed", "dotted", "dotdash", "longdash", or "twodash", where "blank" uses 'invisible lines' (i.e., doesn't draw them).
lwd	the width for lines. Specified as an integer. The default is 1.
fill	fill for areas in plot
ids	Logical value specifying whether to label data points.
idsmode	Determines the way text labels are added to plots. NULL means that only extreme points are labelled. Non-NULL means all data points are labelled. (See <code>link{xpose.plot.default}</code>)
idsext	specifies the extent of the extremes to be used in labelling points. The default is 0.05 (only the most extreme 5% of points are labelled).
idscex	the amount by which labels should be scaled relative to the default. 'NULL' and 'NA' are equivalent to '1.0'.
idsdir	a string indicating the directions of the extremes to include in labelling. Possible values are "up", "down" and "both".
abline	Vector of arguments to the <code>panel.abline</code> function. No abline is drawn if NULL.
abllwd	Line width of any abline.
ablty	Line type of any abline.
ablcol	Line colour of any abline.
lmline	logical variable specifying whether a linear regression line should be superimposed over an <code>xyplot</code> . NULL ~ FALSE. ($y \sim x$)
lmlwd	Line width of the lmline.
lmlty	Line type of the lmline.
lmlcol	Line colour of the lmline.
smooth	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
smlwd	Line width of the x-y smooth.
smlty	Line type of the x-y smooth.

smcol	Line color of the x-y smooth.
smspan	The smoothness parameter for the x-y smooth. The default is 0.667. An argument to panel.loess .
smdegr	The degree of the polynomials to be used for the x-y smooth, up to 2. The default is 1. An argument to panel.loess .
smooth.for.groups	Should a smooth for each group be drawn?
suline	A NULL value indicates that no superposed line should be added to the graph. If non-NULL then this should be the vector (the same length as y) of data points to be used for the smoothed superposed line.
sulwd	Line width of the superposed smooth.
sulty	Line type of the superposed smooth.
sucol	Line color of the superposed smooth.
suspan	The smoothness parameter. The default is 0.667. An argument to panel.loess .
sudegr	The degree of the polynomials to be used, up to 2. The default is 1. An argument to panel.loess .
grid	logical value indicating whether a visual reference grid should be added to the graph. (Could use arguments for line type, color etc).
logy	Logical value indicating whether the y-axis should be logarithmic.
logx	Logical value indicating whether the y-axis should be logarithmic.
force.x.continuous	Logical value indicating whether x-values should be taken as continuous, even if categorical.
bwhoriz	logical value indicating whether box and whiskers should be horizontal or not. The default is FALSE.
bwratio	Ratio of box height to inter-box space. The default is 1.5. An argument for panel.bwplot .
bwvarwid	Logical. If TRUE, widths of boxplots are proportional to the number of points used in creating it. The default is FALSE. An argument for panel.bwplot .
bwdotpch	Graphical parameter controlling the dot plotting character in boxplots. 'bwdotpch="l"' is treated specially, by replacing the dot with a line. The default is 16. An argument for panel.bwplot .
bwdotcol	Graphical parameter controlling the dot colour in boxplots - an integer or string. See 'col'. The default is black. An argument for panel.bwplot .
bwdotcex	The amount by which plotting text and symbols should be scaled relative to the default in boxplots. 'NULL' and 'NA' are equivalent to '1.0'. An argument for panel.bwplot .
bwreccol	The colour to use for the box rectangle in boxplots - an integer or string. The default is blue. See trellis.par.get and "box.rectangle".
bwrecfill	The colour to use for filling the box rectangle in boxplots - an integer or string. The default is transparent (none). See trellis.par.get and "box.rectangle".

bwrectly	The line type for the box rectangle in boxplots - an integer or string. The default is solid. See trellis.par.get and "box.rectangle".
bwreclwd	The width of the lines for the box rectangle in boxplots - an integer. The default is 1. See trellis.par.get and "box.rectangle".
bwumbcol	The colour to use for the umbrellas in boxplots - an integer or string. The default is blue. See trellis.par.get and "box.umbrella".
bwumbly	The line type for the umbrellas in boxplots - an integer or string. The default is solid. See trellis.par.get and "box.umbrella".
bwumbldw	the width of the lines for the umbrellas in boxplots - an integer. The default is 1. See trellis.par.get and "box.umbrella".
bwoutcol	The colour to use for the outliers in boxplots - an integer or string. The default is blue. See trellis.par.get and "box.symbol".
bwoutcex	The amount by which outlier points should be scaled relative to the default in boxplots. 'NULL' and 'NA' are equivalent to '1.0'. The default is 0.8. See trellis.par.get and "box.symbol".
bwoutpch	The plotting character, or symbol, to use for outlier points in boxplots. Specified as an integer. See R help on 'points'. The default is an open circle. See trellis.par.get and "box.symbol".
PI	Either "lines", "area" or "both" specifying whether prediction intervals (as lines, as a shaded area or both) should be computed from the data in SData and added to the display. NULL means no prediction interval.
PI.subset	The subset to be used for the PI.
PI.bin.table	The table used to create VPC plots. Has a specific format created by read.npc.vpc.results
PI.real	Plot the percentiles of the real data in the various bins. values can be NULL or TRUE. Note that for a bin with few actual observations the percentiles will be approximate. For example, the 95th percentile of 4 data points will always be the largest of the 4 data points.
PI.mirror	Plot the percentiles of one simulated data set in each bin. values allowed are NULL, TRUE or AN. INTEGER.VALUE. TRUE takes the first mirror from PI.bin.table and AN. INTEGER.VALUE can be 1, 2, ... n where n is the number of mirror's output in the PI.bin.table. Used mainly by xpose.VPC .
PI.ci	Plot the prediction interval of the simulated data's percentiles for each bin. Values can be "both", "area" or "lines" This can be thought of as a prediction interval about the PI.real or a confidence interval about the PI. However, note that with increasing number of simulations the CI will not go towards zero because the interval is also dependent on the size of the data set.
PPI	The plot prediction interval. Has a specific format that must be followed. See setup.PPI .
PI.mean	Should the mean be plotted in the VPCs? TRUE or FALSE.
PI.delta.mean	Should the delta mean be plotted in the VPCs? TRUE or FALSE.
PI.limits	A vector of two values that describe the limits of the prediction interval that should be displayed. For example $c(0.025, 0.975)$. These limits should be found in the 'PI.bin.table' table. These limits are also used as the percentages for the PI.real, PI.mirror and PI.ci. However, the confidence interval in PI.ci is always the one defined in the PI.bin.table.

PI.arcol	The color of the PI area
PI.up.lty	The upper line type. can be "dotted" or "dashed", etc.
PI.up.type	The upper type used for plotting. Defaults to a line.
PI.up.col	The upper line color
PI.up.lwd	The upper line width
PI.down.lty	The lower line type. can be "dotted" or "dashed", etc.
PI.down.type	The lower type used for plotting. Defaults to a line.
PI.down.col	The lower line color
PI.down.lwd	The lower line width
PI.med.lty	The median line type. can be "dotted" or "dashed", etc.
PI.med.type	The median type used for plotting. Defaults to a line.
PI.med.col	The median line color
PI.med.lwd	The median line width
PI.mean.lty	The mean line type. can be "dotted" or "dashed", etc.
PI.mean.type	The mean type used for plotting. Defaults to a line.
PI.mean.col	The mean line color
PI.mean.lwd	The mean line width
PI.delta.mean.lty	The delta.mean line type. can be "dotted" or "dashed", etc.
PI.delta.mean.type	The delta.mean type used for plotting. Defaults to a line.
PI.delta.mean.col	The delta.mean line color
PI.delta.mean.lwd	The delta.mean line width
PI.ci.up.arcol	The color of the upper PI.ci.
PI.ci.med.arcol	The color of the median PI.ci.
PI.ci.down.arcol	The color of the lower PI.ci.
PI.ci.up.lty	The upper line type. can be "dotted" or "dashed", etc.
PI.ci.up.type	The upper type used for plotting. Defaults to a line.
PI.ci.up.col	The upper line color
PI.ci.up.lwd	The upper line width
PI.ci.down.lty	The lower line type. can be "dotted" or "dashed", etc.
PI.ci.down.type	The lower type used for plotting. Defaults to a line.
PI.ci.down.col	The lower line color
PI.ci.down.lwd	The lower line width
PI.ci.med.lty	The median line type. can be "dotted" or "dashed", etc.

PI.ci.med.type The median type used for plotting. Defaults to a line.
PI.ci.med.col The median line color
PI.ci.med.lwd The median line width
PI.ci.mean.arcol
The color of the mean PI.ci.
PI.ci.mean.lty The mean line type. can be "dotted" or "dashed", etc.
PI.ci.mean.type
The mean type used for plotting. Defaults to a line.
PI.ci.mean.col The mean line color
PI.ci.mean.lwd The mean line width
PI.ci.delta.mean.arcol
The color of the delta.mean PI.ci.
PI.ci.delta.mean.lty
The delta.mean line type. can be "dotted" or "dashed", etc.
PI.ci.delta.mean.type
The delta.mean type used for plotting. Defaults to a line.
PI.ci.delta.mean.col
The delta.mean line color
PI.ci.delta.mean.lwd
The delta.mean line width
PI.real.up.lty The upper line type. can be "dotted" or "dashed", etc.
PI.real.up.type
The upper type used for plotting. Defaults to a line.
PI.real.up.col The upper line color
PI.real.up.lwd The upper line width
PI.real.down.lty
The lower line type. can be "dotted" or "dashed", etc.
PI.real.down.type
The lower type used for plotting. Defaults to a line.
PI.real.down.col
The lower line color
PI.real.down.lwd
The lower line width
PI.real.med.lty
The median line type. can be "dotted" or "dashed", etc.
PI.real.med.type
The median type used for plotting. Defaults to a line.
PI.real.med.col
The median line color
PI.real.med.lwd
The median line width
PI.real.mean.lty
The mean line type. can be "dotted" or "dashed", etc.

PI.real.mean.type
The mean type used for plotting. Defaults to a line.

PI.real.mean.col
The mean line color

PI.real.mean.lwd
The mean line width

PI.real.delta.mean.lty
The delta.mean line type. can be "dotted" or "dashed", etc.

PI.real.delta.mean.type
The delta.mean type used for plotting. Defaults to a line.

PI.real.delta.mean.col
The delta.mean line color

PI.real.delta.mean.lwd
The delta.mean line width

PI.mirror.up.lty
The upper line type. can be "dotted" or "dashed", etc.

PI.mirror.up.type
The upper type used for plotting. Defaults to a line.

PI.mirror.up.col
The upper line color

PI.mirror.up.lwd
The upper line width

PI.mirror.down.lty
The lower line type. can be "dotted" or "dashed", etc.

PI.mirror.down.type
The lower type used for plotting. Defaults to a line.

PI.mirror.down.col
The lower line color

PI.mirror.down.lwd
The lower line width

PI.mirror.med.lty
The median line type. can be "dotted" or "dashed", etc.

PI.mirror.med.type
The median type used for plotting. Defaults to a line.

PI.mirror.med.col
The median line color

PI.mirror.med.lwd
The median line width

PI.mirror.mean.lty
The mean line type. can be "dotted" or "dashed", etc.

PI.mirror.mean.type
The mean type used for plotting. Defaults to a line.

PI.mirror.mean.col
The mean line color

PI.mirror.mean.lwd
The mean line width

PI.mirror.delta.mean.lty	The delta.mean line type. can be "dotted" or "dashed", etc.
PI.mirror.delta.mean.type	The delta.mean type used for plotting. Defaults to a line.
PI.mirror.delta.mean.col	The delta.mean line color
PI.mirror.delta.mean.lwd	The delta.mean line width
PI.ci.area.smooth	Should the "area" for PI.ci be smoothed to match the "lines" argument? Allowed values are TRUE/FALSE. The "area" is set by default to show the bins used in the PI.ci computation. By smoothing, information is lost and, in general, the confidence intervals will be smaller than they are in reality.
autocorr	Is this an autocorrelation plot? Values can be TRUE/FALSE.
vline	Add a vertical line to the plot at the values specified.
vllwd	Width (lwd) of vertical line
vllty	Line type (lty) for vertical line
vlcol	Color (col) of vertical line
hline	Add a horizontal line to the plot at the values specified.
hllwd	Width (lwd) of horizontal line
hllty	Line type (lty) for horizontal line
hlcol	Color (col) of horizontal line
pch.ip.sp	If there is a panel with just one observation then this specifies the type of points for the DV, IPRED and PRED respectively.
cex.ip.sp	If there is a panel with just one observation then this specifies the size of the points for the DV, IPRED and PRED respectively.
...	Other arguments that may be needed in the function.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Justin Wilkins and Andrew Hooker

See Also

xpose.data-class, Cross-references above.

xpose.panel.histogram *Default histogram panel function for Xpose 4*

Description

This is the histogram panel function for Xpose 4. This is not intended to be used outside the xpose.plot.histogram function. Most of the arguments take their default values from xpose.data object but this can be overridden by supplying them as argument to xpose.plot.histogram.

Usage

```
xpose.panel.histogram(x, object,
```

```

breaks = NULL,
dens=TRUE, # density plot on top of histogram?
hidlty = object@Prefs@Graph.prefs$hidlty,
hidcol = object@Prefs@Graph.prefs$hidcol,
hidlwd = object@Prefs@Graph.prefs$hidlwd,
hiborder = object@Prefs@Graph.prefs$hiborder,
hilty = object@Prefs@Graph.prefs$hilty,
hicol = object@Prefs@Graph.prefs$hicol,
hilwd = object@Prefs@Graph.prefs$hilwd,
math.dens = NULL,
vline= NULL,
vllwd= 3,
vllty= 1,
vlcol= "grey",
hline= NULL,
hllwd= 3,
hllty= 1,
hlcol= "grey",
bins.per.panel.equal = TRUE,

showMean = FALSE,
meanllwd= 3,#object@Prefs@Graph.prefs$ablwd,
meanllty= 1,#object@Prefs@Graph.prefs$ablty,
meanlcol= "orange",#object@Prefs@Graph.prefs$ablcol,

showMedian = FALSE,
medianllwd= 3,#object@Prefs@Graph.prefs$ablwd,
medianllty= 1,#object@Prefs@Graph.prefs$ablty,
medianlcol= "black",#object@Prefs@Graph.prefs$ablcol,

```

```

showPCTS = FALSE,
PCTS = c(0.025,0.975),
PCTS1lwd= 2,#object@Prefs@Graph.prefs$abl1wd,
PCTS1lty= hidlty,#object@Prefs@Graph.prefs$abl1ty,
PCTS1col= "black",#object@Prefs@Graph.prefs$abl1col,

## vline settings different for each histogram
vdline= NULL,#object@Prefs@Graph.prefs$abline,
vd1lwd= 3,#object@Prefs@Graph.prefs$abl1wd,
vd1lty= 1,#object@Prefs@Graph.prefs$abl1ty,
vd1col= "red",#object@Prefs@Graph.prefs$abl1col,
... ,
groups)

```

Arguments

x	Name(s) of the x-variable.
object	An xpose.data object.
breaks	The breakpoints for the histogram.
dens	Density plot on top of histogram?
hidlty	Density line type.
hidcol	Color of density line.
hidlwd	Width of density line.
hiborder	Colour of the bar borders.
hilty	Line type for the bar borders.
hicol	Fill colour for the bars.
hilwd	Width for the bar borders.
math.dens	Should a density line be drawn. Values are NULL or TRUE.
vline	NULL or a vector of locations for the vertical lines to be drawn. For example, <code>vline=c(50,60)</code> will draw two vertical lines. The function <code>panel.abline</code> is used.
v1lwd	Line width of the vertical lines defined with <code>vline</code> . Can be a vector or a single value, for example <code>v1lwd=2</code> or <code>v1lwd=c(2,3)</code> .
v1lty	Line type of the vertical lines defined with <code>vline</code> . Can be a vector or a single value, for example <code>v1lty=1</code> or <code>v1lty=c(1,2)</code> .
vlcol	Line color of the vertical lines defined with <code>vline</code> . Can be a vector or a single value, for example <code>vlcol="red"</code> or <code>v1lty=c("red","blue")</code> .
hline	NULL or a vector of locations for the horizontal lines to be drawn. For example, <code>hline=c(50,60)</code> will draw two horizontal lines. The function <code>panel.abline</code> is used.
h1lwd	Line width of the horizontal lines defined with <code>hline</code> . Can be a vector or a single value, for example <code>h1lwd=2</code> or <code>h1lwd=c(2,3)</code> .
h1lty	Line type of the horizontal lines defined with <code>hline</code> . Can be a vector or a single value, for example <code>h1lty=1</code> or <code>h1lty=c(1,2)</code> .

<code>hlcol</code>	Line color of the horizontal lines defined with <code>hline</code> . Can be a vector or a single value, for example <code>hlcol="red"</code> or <code>hlty=c("red", "blue")</code> .
<code>bins.per.panel.equal</code>	Allow for different bins in different panels for continuous data? TRUE or FALSE.
<code>showMean</code>	Should the mean of the data in the histogram be shown?
<code>meanllwd</code>	Line width of mean line.
<code>meanllty</code>	The line type for the mean
<code>meanlcol</code>	Color for the mean line
<code>showMedian</code>	Should the median of the data for the histogram be shown as a vertical line?
<code>medianllwd</code>	line width of median line.
<code>medianllty</code>	line type of median line.
<code>medianlcol</code>	color of median line.
<code>showPCTS</code>	Should percentiles of the data for the histogram be shown?
<code>PCTS</code>	A vector of percentiles to show. Can be any length.
<code>PCTSllwd</code>	line width of percentiles. Can be a vector of same length as PCTS.
<code>PCTSllty</code>	Line type of the percentiles. Can be a vector of same length as PCTS.
<code>PCTSlcol</code>	Color of the percentiles. Can be a vector of same length as PCTS.
<code>vdline</code>	vertical line different for each histogram. Must be a vector.
<code>vdllwd</code>	line widths
<code>vdllty</code>	line types
<code>vdlcol</code>	line colors
<code>...</code>	Other arguments that may be needed in the function.
<code>groups</code>	used to pass the conditioning variable into this function.

Author(s)

Andrew Hooker, Mats Karlsson, Justin Wilkins & E. Niclas Jonsson

See Also

`xpose.data-class`, Cross-references above.

xpose.panel.qq *Default QQ panel function for Xpose 4*

Description

This is the QQ panel function for Xpose 4. This is not intended to be used outside the `xpose.plot.qq` function. Most of the arguments take their default values from `xpose.data` object but this can be overridden by supplying them as argument to `xpose.plot.qq`.

Usage

```
xpose.panel.qq(x, object,
               pch = object@Prefs@Graph.prefs$pch,
               col = object@Prefs@Graph.prefs$col,
               cex = object@Prefs@Graph.prefs$cex,
               abllty = object@Prefs@Graph.prefs$abllty,
               abllwd = object@Prefs@Graph.prefs$abllwd,
               ablcol = object@Prefs@Graph.prefs$ablcol,
               grid = object@Prefs@Graph.prefs$grid,
               ...)
```

Arguments

<code>x</code>	Name(s) of the x-variable.
<code>object</code>	An <code>xpose.data</code> object.
<code>col</code>	Colour of lines and plot symbols.
<code>pch</code>	Plot character to use.
<code>cex</code>	Amount to scale the plotting character by.
<code>abllty</code>	Line type.
<code>abllwd</code>	Line width.
<code>ablcol</code>	Line colour.
<code>grid</code>	logical value indicating whether a visual reference grid should be added to the graph. (Could use arguments for line type, color etc).
<code>...</code>	Other arguments that may be needed in the function.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.qq](#), [qqmath](#), [panel.qqmathline](#), [xpose.data-class](#)

xpose.panel.splom *Scatterplot matrix panel function for Xpose 4*

Description

This is the scatterplot matrix panel function for Xpose 4. This is not intended to be used outside the xpose.plot.splom function. Most of the arguments take their default values from xpose.data object but this can be overridden by supplying them as argument to xpose.plot.splom.

Usage

```
xpose.panel.splom(x, y, object,
                 subscripts,
                 onlyfirst = TRUE,
                 inclZeroWRES = FALSE,
                 type = "p",
                 col = object@Prefs@Graph.prefs$col,
                 pch = object@Prefs@Graph.prefs$pch,
                 cex = object@Prefs@Graph.prefs$cex,
                 lty = object@Prefs@Graph.prefs$lty,
                 lwd = object@Prefs@Graph.prefs$lwd,
                 smooth = TRUE,
                 smlwd = object@Prefs@Graph.prefs$smlwd,
                 smlty = object@Prefs@Graph.prefs$smlty,
                 smcol = object@Prefs@Graph.prefs$smcol,
                 smspan = object@Prefs@Graph.prefs$smspan,
                 smdegr = object@Prefs@Graph.prefs$smdegr,
                 lm1line = NULL,
                 lm1lwd = object@Prefs@Graph.prefs$lm1lwd,
                 lm1lty = object@Prefs@Graph.prefs$lm1lty,
                 lmcol = object@Prefs@Graph.prefs$lmcol,
                 grid = object@Prefs@Graph.prefs$grid,

                 groups = NULL,
                 ... )
```

Arguments

x	Name(s) of the x-variable.
y	Name(s) of the y-variable.
object	An xpose.data object.
subscripts	The standard Trellis subscripts argument (see xyplot)
groups	Name of the variable used for superpose plots.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot.

onlyfirst	Logical value indicating whether only the first row per individual is included in the plot.
type	1-character string giving the type of plot desired. The following values are possible, for details, see 'plot': "p" for points, "l" for lines, "o" for overplotted points and lines, "b", "c" for (empty if "c") points joined by lines, "s" and "S" for stair steps and "h" for histogram-like vertical lines. Finally, "n" does not produce any points or lines.
col	The color for lines and points. Specified as an integer or a text string. A full list is obtained by the R command <code>colours()</code> . The default is blue (<code>col=4</code>).
pch	The plotting character, or symbol, to use. Specified as an integer. See R help on points . The default is an open circle.
cex	The amount by which plotting text and symbols should be scaled relative to the default. 'NULL' and 'NA' are equivalent to '1.0'.
lty	The line type. Line types can either be specified as an integer (0=blank, 1=solid, 2=dashed, 3=dotted, 4=dotdash, 5=longdash, 6=twodash) or as one of the character strings "blank", "solid", "dashed", "dotted", "dotdash", "longdash", or "twodash", where "blank" uses 'invisible lines' (i.e., doesn't draw them).
lwd	the width for lines. Specified as an integer. The default is 1.
lmline	logical variable specifying whether a linear regression line should be superimposed over an xyplot . NULL ~ FALSE. ($y \sim x$)
lmlwd	Line width of the lmline.
lmlty	Line type of the lmline.
lmcol	Line colour of the lmline.
smooth	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
smlwd	Line width of the x-y smooth.
smlty	Line type of the x-y smooth.
smcol	Line color of the x-y smooth.
smspan	The smoothness parameter for the x-y smooth. The default is 0.667. An argument to panel.loess .
smdegr	The degree of the polynomials to be used for the x-y smooth, up to 2. The default is 1. An argument to panel.loess .
grid	logical value indicating whether a visual reference grid should be added to the graph. (Could use arguments for line type, color etc).
...	Other arguments that may be needed in the function.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.plot.splom](#), [xpose.data-class](#), [xyplot.splom](#), [panel.splom](#), [panel.pairs](#)

xpose.plot.bw

The generic Xpose functions for box-and-whisker plots

Description

This is a wrapper function for the lattice `bwplot` function.

Usage

```
xpose.plot.bw(x,y,object,
              inclZeroWRES = FALSE,
              onlyfirst     = FALSE,
              samp          = NULL,
              panel         = xpose.panel.bw,
              groups       = NULL,
              ids           = FALSE,
              logy          = FALSE,
              logx          = FALSE,
              aspect        = object@Prefs@Graph.prefs$aspect,
              funy          = NULL,
              funx          = NULL,

              ## Prediction interval settings
              PI            = FALSE,

              ## Conditioning settings
              by = object@Prefs@Graph.prefs$condvar,
              force.by.factor = FALSE,
              ordby      = object@Prefs@Graph.prefs$ordby,
              byordfun   = object@Prefs@Graph.prefs$byordfun,
              shingnum   = object@Prefs@Graph.prefs$shingnum,
              shingol    = object@Prefs@Graph.prefs$shingol,
              strip = function(...)
                strip.default(...,strip.names=c(TRUE,TRUE)),

              ## Subset stuff
              subset      = xsubset(object),

              ## Axes and titles
              main        = xpose.create.title(x,y,object,subset,funx,funy,...),
              xlb         = xpose.create.label(x,object,funx,logx,...),
              ylb         = xpose.create.label(y,object,funy,logy,...),
              scales      = list(),

              ## Superpose smooth
              suline       = object@Prefs@Graph.prefs$suline,
```

```

## bins
binvar      = NULL,
bins        = 10,

## mirror stuff
mirror      = FALSE,
max.plots.per.page=4,
mirror.aspect="fill",
pass.plot.list=FALSE,
x.cex=NULL,
y.cex=NULL,
main.cex=NULL,
mirror.internal=list(strip.missing=missing(strip)),
...)

```

Arguments

x	Name(s) of the x-variable.
y	Name(s) of the y-variable.
object	An xpose.data object.
inclZeroWRES	A logical value indicating whether rows with WRES=0 should be plotted.
onlyfirst	A logical value indicating whether only the first row per individual should be included in the plot.
samp	An integer between 1 and object@Nsim (see xpose.data-class) specifying which of the simulated data sets to extract from SData.
panel	The name of the panel function to use. This should in most cases be left as xpose.panel.bw .
groups	A string with the name of any grouping variable (used as the groups argument to panel.xyplot).
ids	A logical value indicating whether text labels should be used as plotting symbols (the variable used for these symbols indicated by the idlab Xpose data variable).
logy	Logical value indicating whether the y-axis should be logarithmic.
logx	Logical value indicating whether the x-axis should be logarithmic.
aspect	The aspect ratio of the display (see bwplot).
funy	String with the name of a function to apply to the y-variable before plotting, e.g. "abs".
funx	String with the name of a function to apply to the x-variable before plotting, e.g. "abs".
PI	Either "lines", "area" or "both" specifying whether prediction intervals (as lines, as a shaded area or both) should be computed from the data in SData and added to the display. NULL means no prediction interval.
by	A string or a vector of strings with the name(s) of the conditioning variables.

<code>force.by.factor</code>	Logical value. If TRUE, and <code>by</code> is not NULL, the variable specified by <code>by</code> is taken as categorical.
<code>ordby</code>	A string with the name of a variable to be used to reorder any factor conditioning variables (<code>by</code>). The variable is used in a call to the <code>reorder</code> function.
<code>byordfun</code>	The name of the function to be used when reordering a factor conditioning variable (see argument <code>ordby</code>).
<code>shingnum</code>	The number of shingles ("parts") a continuous conditioning variable should be divided into.
<code>shingol</code>	The amount of overlap between adjacent shingles (see argument <code>shingnum</code>)
<code>strip</code>	The name of the function to be used as the <code>strip</code> argument to the <code>bwplot</code> .
<code>main</code>	A string giving the plot title or NULL if none.
<code>xlb</code>	A string giving the label for the x-axis. NULL if none.
<code>ylb</code>	A string giving the label for the y-axis. NULL if none.
<code>subset</code>	A string giving the subset expression to be applied to the data before plotting. See <code>xsubset</code> .
<code>scales</code>	A list to be used for the <code>scales</code> argument in <code>bwplot</code> .
<code>suline</code>	A string giving the variable to be used to construct a smooth to superpose on the display. NULL if none. This argument is used if you want to add a superpose line of a variable not present in the <code>y</code> list of variables.
<code>binvar</code>	Variable to be used for binning.
<code>bins</code>	The number of bins to be used. The default is 10.
<code>mirror</code>	Should we create mirror plots from simulation data? Value can be FALSE, TRUE or 1 for one mirror plot, or 3 for three mirror plots.
<code>max.plots.per.page</code>	The maximum number of plots per page that can be created with the mirror plots.
<code>mirror.aspect</code>	The aspect ratio of the plots used for mirror functionality.
<code>pass.plot.list</code>	Should we pass the list of plots created with mirror or should we print them directly. Values can be TRUE/FALSE.
<code>x.cex</code>	The size of the x-axis label.
<code>y.cex</code>	The size of the y-axis label.
<code>main.cex</code>	The size of the title.
<code>mirror.internal</code>	an internal mirror argument used in <code>create.mirror</code> . Checks if the <code>strip</code> argument from <code>bwplot</code> has been used.
<code>...</code>	Other arguments passed to <code>xpose.panel.bw</code> .

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.data-class](#), Cross-references above.

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## Box & whisker plot of WRES vs PRED
xpose.plot.bw("WRES", "PRED", xpdb5, binvar="PRED")

## End(Not run)
```

xpose.plot.default *The Xpose 4 generic functions for continuous y-variables.*

Description

This function is a wrapper for the lattice xyplot function.

Usage

```
xpose.plot.default(x,y,object,
  inclZeroWRES = FALSE,
  onlyfirst    = FALSE,
  samp         = NULL,
  panel        = xpose.panel.default,
  groups       = object@Prefs@Xvardef$id,
  ids          = object@Prefs@Graph.prefs$ids,
  logy         = FALSE,
  logx         = FALSE,
  yscale.components= "default",#function(...) yscale.components.default(...),
  xscale.components= "default",#function(...) xscale.components.default(...),

  aspect       = object@Prefs@Graph.prefs$aspect,
  funx         = NULL,
  funy         = NULL,
  iplot        = NULL,

  ## Prediction interval settings
  PI           = NULL,

  ## Conditioning settings
  by=object@Prefs@Graph.prefs$condvar,
```

```

force.by.factor = FALSE,
ordby           = object@Prefs@Graph.prefs$ordby,
byordfun       = object@Prefs@Graph.prefs$byordfun,
shingnum       = object@Prefs@Graph.prefs$shingnum,
shingol        = object@Prefs@Graph.prefs$shingol,
by.interval    = NULL,
strip = function(...){
  strip.default(...,strip.names=c(TRUE,TRUE))
},
use.xpose.factor.strip.names=TRUE,

## Subset stuff
subset         = xsubset(object),

autocorr=FALSE,

## Axes and titles
main           = xpose.create.title(x,y,object,subset,funx,funy,...),
xlb           = xpose.create.label(x,object,funx,logx,autocorr.x=autocorr,...),
ylb           = xpose.create.label(y,object,funy,logy,autocorr.y=autocorr,...),
scales        = list(),

## Superpose smooth
suline        = object@Prefs@Graph.prefs$suline,

## Categorical stuff
bwhoriz       = object@Prefs@Graph.prefs$bwhoriz,

## Dilution stuff
dilution     = FALSE,
dilfrac       = object@Prefs@Graph.prefs$dilfrac,
diltype       = object@Prefs@Graph.prefs$diltype,
dilci         = object@Prefs@Graph.prefs$dilci,
seed          = NULL,

mirror        = FALSE,
max.plots.per.page=4,
mirror.aspect="fill",
pass.plot.list=FALSE,
x.cex=NULL,
y.cex=NULL,
main.cex=NULL,
mirror.internal=list(strip.missing=missing(strip)),
...)

```

Arguments

x A string or a vector of strings with the name(s) of the x-variable(s).

y	A string or a vector of strings with the name(s) of the y-variable(s).
object	An "xpose.data" object.
inclZeroWRES	A logical value indicating whether rows with WRES=0 should be plotted.
onlyfirst	A logical value indicating whether only the first row per individual should be included in the plot.
samp	An integer between 1 and object@Nsim (see xpose.data-class) specifying which of the simulated data sets to extract from SData.
panel	The name of the panel function to use.
groups	A string with the name of any grouping variable (used as the groups argument to panel.xyplot).
ids	A logical value indicating whether text labels should be used as plotting symbols (the variable used for these symbols indicated by the idlab xpose data variable).
logy	Logical value indicating whether the y-axis should be logarithmic.
logx	Logical value indicating whether the x-axis should be logarithmic.
yscale.components	Used to change the way the axis look if logy is used. Can be a user defined function or link{xpose.yscale.components.log10}. If the axes are not log transformed then yscale.components.default is used.
xscale.components	Used to change the way the axis look if logx is used. Can be a user defined function or link{xpose.xscale.components.log10}. If the axes are not log transformed then xscale.components.default is used.
aspect	The aspect ratio of the display (see xyplot).
funx	String with the name of a function to apply to the x-variable before plotting, e.g. "abs".
funy	String with the name of a function to apply to the y-variable before plotting, e.g. "abs".
iplot	Is this an individual plots matrix? Internal use only.
PI	Either "lines", "area" or "both" specifying whether prediction intervals (as lines, as a shaded area or both) should be computed from the data in SData and added to the display. NULL means no prediction interval.
by	A string or a vector of strings with the name(s) of the conditioning variables.
force.by.factor	Logical value. If TRUE, and by is not NULL, the variable specified by by is taken as categorical.
ordby	A string with the name of a variable to be used to reorder any factor conditioning variables (by). The variable is used in a call to the reorder.factor function.
byordfun	The name of the function to be used when reordering a factor conditioning variable (see argument ordby)
shingnum	The number of shingles ("parts") a continuous conditioning variable should be divided into.
shingol	The amount of overlap between adjacent shingles (see argument shingnum)

<code>by.interval</code>	The intervals to use for conditioning on a continuous variable with <code>by</code> .
<code>strip</code>	The name of the function to be used as the <code>strip</code> argument to the <code>xyplot</code> . An easy way to change the strip appearance is to use <code>strip.custom</code> . For example, if you want to change the text in the strips you can use <code>strip=strip.custom(factor.levels=c("Hi", "There"))</code> if the <code>by</code> variable is a factor and <code>strip=strip.custom(var.name=c("New Name"))</code> if the <code>by</code> variable is continuous.
<code>use.xpose.factor.strip.names</code>	Use factor names in strips of conditioning plots..
<code>main</code>	A string giving the plot title or NULL if none.
<code>xlb</code>	A string giving the label for the x-axis. NULL if none.
<code>ylb</code>	A string giving the label for the y-axis. NULL if none.
<code>subset</code>	A string giving the subset expression to be applied to the data before plotting. See <code>xsubset</code> .
<code>autocorr</code>	Is this an autocorrelation plot? Values can be TRUE/FALSE.
<code>scales</code>	A list to be used for the <code>scales</code> argument in <code>xyplot</code> .
<code>suline</code>	A string giving the variable to be used to construct a smooth to superpose on the display. NULL if none. This argument is used if you want to add a superpose line of a variable not present in the <code>y</code> list of variables.
<code>bwhoriz</code>	A logical value indicating if box and whiskers bars should be plotted horizontally or not. Used when the x-variable(s) is categorical.
<code>dilution</code>	Logical value indicating whether data dilution should be used.
<code>diltype</code>	Indicating what type of dilution to apply. NULL means random dilution without stratification. A nonNULL value means stratified dilution.
<code>dilfrac</code>	Dilution fraction indicating the expected fraction of individuals to display in the plots. The exact meaning depends on the type of dilution (see below).
<code>dilci</code>	A number between 0 and 1 giving the range eligible for dilution in a stratified dilution (see below).
<code>seed</code>	Seed number used for random dilution. NULL means no seed.
<code>mirror</code>	Should we create mirror plots from simulation data? Value can be FALSE, TRUE or 1 for one mirror plot, or 3 for three mirror plots.
<code>max.plots.per.page</code>	The maximum number of plots per page that can be created with the mirror plots.
<code>mirror.aspect</code>	The aspect ratio of the plots used for mirror functionality.
<code>pass.plot.list</code>	Should we pass the list of plots created with mirror or should we print them directly. Values can be TRUE/FALSE.
<code>x.cex</code>	The size of the x-axis label.
<code>y.cex</code>	The size of the y-axis label.
<code>main.cex</code>	The size of the title.
<code>mirror.internal</code>	an internal mirror argument used in <code>create.mirror</code> . Checks if the <code>strip</code> argument from <code>xyplot</code> has been used.
<code>...</code>	Other arguments passed to <code>xpose.panel.default</code> .

Details

y must be numeric (continuous) while x can be either numeric or factor. If x is numeric then a regular xy-plot is drawn. If x is a factor, on the other hand, a box and whiskers plot is constructed.

x and y can be either single valued strings or vector of strings. x and y can not both be vectors in the same call to the function.

If `ids` is TRUE, text labels are added to the plotting symbols. The labels are taken from the `idlab` xpose data variable. The way the text labels are plotted is governed by the `idsmode` argument (passed down to the panel function). `idsmode=NULL` (the default) means that only extreme data points are labelled while a non-NULL value adds labels to all data points (the default in Xpose 3). `xpose.panel.default` identifies extreme data points by fitting a loess smooth ($y \sim x$) and looking at the residuals from that fit. Points that are associated with the highest/lowest residuals are labelled. "High" and "low" are judged by the panel function parameter `idsext`, which gives the fraction of the total number of data points that are to be judged extreme in the "up" and "down" direction. The default value for `idsext` is 0.05 (see [xpose.prefs-class](#)). There is also a possibility to label only the high or low extreme points. This is done through the `idsdir` argument to `xpose.panel.default`. A value of "both" (the default) means that both high and low extreme points are labelled while "up" and "down" labels the high and low extreme points respectively.

Data dilution is useful in situations when there is an excessive amount of data. `xpose.plot.default` can dilute data in two different ways. The first is a completely random dilution in which all individuals are eligible for exclusion from the plot. In this case the argument `dilfrac` determines the fraction of individuals that are excluded from the plot. The second type of dilution uses stratification to make sure that none of the extreme individuals are omitted from the plot. Extreme individuals are identified in a similar manner as extreme data points are identified for text labelling. A smooth is fitted to the data and the extreme residuals from that fit is used to inform about extremeness. What is judged as extreme is determined by the argument `dilci`, which defaults to 0.95 (Note that the meaning of this is the opposite to `idsext`). `dilci` give the confidence level of the interval around the fitted curve outside of which points are deemed to be extreme. Extreme individuals are those that have at least one point in the "extremeness" interval. Individuals that do not have any extreme points are eligible for dilution and `dilfrac` give the number of these that should be omitted from the graph. This means that `dilfrac` should usually be greater for stratified dilution than in completely random dilution. Any smooths added to a diluted plot is based on undiluted data.

More graphical parameters may be passed to [xpose.panel.default](#).

Value

Returns a xyplot graph object.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.panel.default](#), [xyplot](#), [panel.xyplot](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## A spaghetti plot of DV vs TIME
xpose.plot.default("TIME", "DV", xpdb5)

## A conditioning plot
xpose.plot.default("TIME", "DV", xpdb5, by = "SEX")

## Multiple x-variables
xpose.plot.default(c("WT", "SEX"), "CL", xpdb5)

## Multiple y-variables
xpose.plot.default("WT", c("CL", "V"), xpdb5)
xpose.plot.default("WT", c("CL", "V"), xpdb5, by=c("SEX", "HCTZ"))

## determining the interval for the conditioning variable
wt.ints <- matrix(c(50,60,60,70,70,80,80,90,90,100,100,150),nrow=6,ncol=2,byrow=T)
xpose.plot.default("TIME", "DV", xpdb5, by="WT", by.interval=wt.ints)

## End(Not run)
```

xpose.plot.histogram *The Xpose 4 generic functions for continuous y-variables.*

Description

This function is a wrapper for the lattice xyplot function.

Usage

```
xpose.plot.histogram(x,object,
  inclZeroWRES = FALSE,
  onlyfirst    = FALSE,
  samp         = NULL,
  type         = "density",
  aspect       = object@Prefs@Graph.prefs$aspect,
  scales       = list(),
  by=object@Prefs@Graph.prefs$condvar,
  force.by.factor = FALSE,
  ordby        = object@Prefs@Graph.prefs$ordby,
  byordfun     = object@Prefs@Graph.prefs$byordfun,
  shingnum     = object@Prefs@Graph.prefs$shingnum,
```

```

shingol = object@Prefs@Graph.prefs$shingol,
strip = function(...)
strip.default(...,strip.names=c(TRUE,TRUE)),
subset = xsubset(object),
main = xpose.create.title.hist(x,object,subset,...),
xlb = NULL,
ylb = "Density",
hicol = object@Prefs@Graph.prefs$hicol,
hilty = object@Prefs@Graph.prefs$hilty,
hilwd = object@Prefs@Graph.prefs$hilwd,
hidcol = object@Prefs@Graph.prefs$hidcol,
hidlty = object@Prefs@Graph.prefs$hidlty,
hidlwd = object@Prefs@Graph.prefs$hidlwd,
hiborder = object@Prefs@Graph.prefs$hiborder,
mirror = FALSE,
max.plots.per.page=4,
mirror.aspect="fill",
pass.plot.list=FALSE,
x.cex=NULL,
y.cex=NULL,
main.cex=NULL,
mirror.internal=list(strip.missing=missing(strip)),
...)

```

Arguments

x	A string or a vector of strings with the name(s) of the x-variable(s).
object	An "xpose.data" object.
inclZeroWRES	A logical value indicating whether rows with WRES=0 should be plotted.
onlyfirst	A logical value indicating whether only the first row per individual should be included in the plot.
samp	An integer between 1 and object@Nsim (see xpose.data-class) specifying which of the simulated data sets to extract from SData.
type	The type of histogram to make. See histogram .
aspect	The aspect ratio of the display (see histogram).
scales	A list to be used for the scales argument in histogram .
by	A string or a vector of strings with the name(s) of the conditioning variables.
force.by.factor	Logical value. If TRUE, and by is not NULL, the variable specified by by is taken as categorical.
ordby	A string with the name of a variable to be used to reorder any factor conditioning variables (by). The variable is used in a call to the <code>reorder.factor</code> function.
byordfun	The name of the function to be used when reordering a factor conditioning variable (see argument <code>ordby</code>)
shingnum	The number of shingles ("parts") a continuous conditioning variable should be divided into.

shingol	The amount of overlap between adjacent shingles (see argument shingnum)
strip	The name of the function to be used as the strip argument to the xyplot .
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
main	A string giving the plot title or NULL if none.
xlb	A string giving the label for the x-axis. NULL if none.
ylb	A string giving the label for the y-axis. NULL if none.
hiborder	the border colour of the histogram - an integer or string. The default is black (see histogram).
hicol	the fill colour of the histogram - an integer or string. The default is blue (see histogram).
hilty	the border line type of the histogram - an integer. The default is 1 (see histogram).
hilwd	the border line width of the histogram - an integer. The default is 1 (see histogram).
hidcol	the fill colour of the density line - an integer or string. The default is black (see histogram).
hidlty	the border line type of the density line - an integer. The default is 1 (see histogram).
hidlwd	the border line width of the density line - an integer. The default is 1 (see histogram).
mirror	Should we create mirror plots from simulation data? Value can be FALSE, TRUE or 1 for one mirror plot, or 3 for three mirror plots.
max.plots.per.page	The maximum number of plots per page that can be created with the mirror plots.
mirror.aspect	The aspect ratio of the plots used for mirror functionality.
pass.plot.list	Should we pass the list of plots created with mirror or should we print them directly. Values can be TRUE/FALSE.
x.cex	The size of the x-axis label.
y.cex	The size of the y-axis label.
main.cex	The size of the title.
mirror.internal	an internal mirror argument used in create.mirror . Checks if the strip argument from xyplot has been used.
...	Other arguments passed to xpose.plot.histogram .

Details

x can be either numeric or factor, and can be either single valued strings or vectors of strings.

Value

Returns a histogram.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.panel.histogram](#), [histogram](#), [panel.histogram](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

xpose.plot.histogram("AGE", xpdb5, onlyfirst = TRUE)
xpose.plot.histogram(c("SEX", "AGE"), xpdb5, onlyfirst = TRUE)

## End(Not run)
```

xpose.plot.qq

The generic Xpose functions for QQ plots

Description

This is a wrapper function for the lattice [qqmath](#) function.

Usage

```
xpose.plot.qq(x, object,
  inclZeroWRES = FALSE,
  onlyfirst    = FALSE,
  samp         = NULL,
  aspect       = object@Prefs@Graph.prefs$aspect,
  scales       = list(),
  by=object@Prefs@Graph.prefs$condvar,
  force.by.factor = FALSE,
  ordby        = object@Prefs@Graph.prefs$ordby,
  byordfun     = object@Prefs@Graph.prefs$byordfun,
  shingnum     = object@Prefs@Graph.prefs$shingnum,
  shingol      = object@Prefs@Graph.prefs$shingol,
  strip = function(...)
  strip.default(...,strip.names=c(TRUE,TRUE)),
  subset       = xsubset(object),
  main = xpose.create.title.hist(x,object,subset,...),
  xlb          = "Quantiles of Normal",
  ylb         = paste("Quantiles of ",xlabel(x,object),sep=""),
```

```

pch=object@Prefs@Graph.prefs$pch,
col=object@Prefs@Graph.prefs$col,
cex=object@Prefs@Graph.prefs$cex,
abllty = object@Prefs@Graph.prefs$abllty,
abllwd = object@Prefs@Graph.prefs$abllwd,
ablcol = object@Prefs@Graph.prefs$ablcol,
mirror      = FALSE,
max.plots.per.page=4,
mirror.aspect="fill",
pass.plot.list=FALSE,
x.cex=NULL,
y.cex=NULL,
main.cex=NULL,
mirror.internal=list(strip.missing=missing(strip)),
...)
```

Arguments

x	A string or a vector of strings with the name(s) of the x-variable(s).
object	An "xpose.data" object.
inclZeroWRES	A logical value indicating whether rows with WRES=0 should be plotted.
onlyfirst	A logical value indicating whether only the first row per individual should be included in the plot.
samp	An integer between 1 and object@Nsim (see xpose.data-class) specifying which of the simulated data sets to extract from SData.
aspect	The aspect ratio of the display (see qqmath).
scales	A list to be used for the scales argument in qqmath .
by	A string or a vector of strings with the name(s) of the conditioning variables.
force.by.factor	Logical value. If TRUE, and by is not NULL, the variable specified by by is taken as categorical.
ordby	A string with the name of a variable to be used to reorder any factor conditioning variables (by). The variable is used in a call to the reorder function.
byordfun	The name of the function to be used when reordering a factor conditioning variable (see argument ordby).
shingnum	The number of shingles ("parts") a continuous conditioning variable should be divided into.
shingol	The amount of overlap between adjacent shingles (see argument shingnum).
strip	The name of the function to be used as the strip argument to the xyplot .
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
main	A string giving the plot title or NULL if none.
xlb	A string giving the label for the x-axis. NULL if none.
ylb	A string giving the label for the y-axis. NULL if none.

pch	Plotting symbol.
col	Color of plotting symbol.
cex	Amount to scale the plotting character by.
abllty	Line type for qqline.
abllwd	Line width for qqline.
ablcol	Color for qqline.
mirror	Should we create mirror plots from simulation data? Value can be FALSE, TRUE or 1 for one mirror plot, or 3 for three mirror plots.
max.plots.per.page	The maximum number of plots per page that can be created with the mirror plots.
mirror.aspect	The aspect ratio of the plots used for mirror functionality.
pass.plot.list	Should we pass the list of plots created with mirror or should we print them directly. Values can be TRUE/FALSE.
x.cex	The size of the x-axis label.
y.cex	The size of the y-axis label.
main.cex	The size of the title.
mirror.internal	an internal mirror argument used in create.mirror . Checks if the strip argument from qqmath has been used.
...	Other arguments passed to xpose.plot.qq .

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.panel.qq](#), [qqmath](#), [panel.qqmathline](#), [xpose.data-class](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## A QQ plot of WRES
xpose.plot.qq("WRES", xpdb5)

## End(Not run)
```

xpose.plot.splom *The Xpose 4 generic functions for scatterplot matrices.*

Description

This function is a wrapper for the lattice splom function.

Usage

```
xpose.plot.splom(plist, object,
  varnames=NULL,
  main = "Scatterplot Matrix",
  xlb = NULL,
  ylb = NULL,
  scales = list(),
  onlyfirst=TRUE,
  inclZeroWRES=FALSE,
  subset = xsubset(object),
  by=object@Prefs@Graph.prefs$condvar,
  force.by.factor=FALSE,
  include.cat.vars = FALSE,
  ordby      = NULL,
  byordfun   = object@Prefs@Graph.prefs$byordfun,
  shingnum   = object@Prefs@Graph.prefs$shingnum,
  shingol    = object@Prefs@Graph.prefs$shingol,
  strip = function(...)
  strip.default(...,strip.names=c(TRUE,TRUE)),
  groups = NULL,
  ids = object@Prefs@Graph.prefs$ids,
  smooth      = TRUE,
  lmline = NULL,
  panel      = xpose.panel.splom,
  aspect = object@Prefs@Graph.prefs$aspect,
  samp=NULL,
  max.plots.per.page=4,
  mirror      = FALSE,
  mirror.aspect="fill",
  pass.plot.list=FALSE,
  x.cex=NULL,
  y.cex=NULL,
  main.cex=NULL,
  mirror.internal=list(strip.missing=missing(strip)),
  ...)
```

Arguments

plist A vector of strings containing variable names for the scatterplot matrix.

object	An "xpose.data" object.
varnames	A vector of strings containing labels for the variables in the scatterplot matrix.
inclZeroWRES	A logical value indicating whether rows with WRES=0 should be plotted.
onlyfirst	A logical value indicating whether only the first row per individual should be included in the plot.
panel	The name of the panel function to use.
lmline	logical variable specifying whether a linear regression line should be superimposed over an xyplot . NULL ~ FALSE. (y~x)
smooth	A NULL value indicates that no superposed line should be added to the graph. If TRUE then a smooth of the data will be superimposed.
groups	A string with the name of any grouping variable (used as the groups argument to <code>panel.xyplot</code>).
ids	A logical value indicating whether text labels should be used as plotting symbols (the variable used for these symbols indicated by the <code>idlab</code> xpose data variable).
aspect	The aspect ratio of the display (see xyplot).
by	A string or a vector of strings with the name(s) of the conditioning variables.
force.by.factor	Logical value. If TRUE, and by is not NULL, the variable specified by by is taken as categorical.
include.cat.vars	Logical value.
ordby	A string with the name of a variable to be used to reorder any factor conditioning variables (by). The variable is used in a call to the <code>reorder.factor</code> function.
byordfun	The name of the function to be used when reordering a factor conditioning variable (see argument <code>ordby</code>)
shingnum	The number of shingles ("parts") a continuous conditioning variable should be divided into.
shingol	The amount of overlap between adjacent shingles (see argument <code>shingnum</code>)
strip	The name of the function to be used as the strip argument to the xyplot .
main	A string giving the plot title or NULL if none.
xlb	A string giving the label for the x-axis. NULL if none.
ylb	A string giving the label for the y-axis. NULL if none.
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
scales	A list to be used for the scales argument in <code>xyplot</code> .
mirror	Should we create mirror plots from simulation data? Value can be FALSE, TRUE or 1 for one mirror plot, or 3 for three mirror plots.
max.plots.per.page	The maximum number of plots per page that can be created with the mirror plots.
mirror.aspect	The aspect ratio of the plots used for mirror functionality.

<code>samp</code>	An integer between 1 and <code>object@Nsim</code> (see xpose.data-class) specifying which of the simulated data sets to extract from <code>SData</code> .
<code>pass.plot.list</code>	Should we pass the list of plots created with <code>mirror</code> or should we print them directly. Values can be TRUE/FALSE.
<code>x.cex</code>	The size of the x-axis label.
<code>y.cex</code>	The size of the y-axis label.
<code>main.cex</code>	The size of the title.
<code>mirror.internal</code>	an internal mirror argument used in create.mirror . Checks if the <code>strip</code> argument from qqmath has been used.
<code>...</code>	Other arguments passed to xpose.panel.default .

Details

If `ids` is TRUE, text labels are added to the plotting symbols. The labels are taken from the `idlab` xpose data variable. The way the text labels are plotted is governed by the `idsmode` argument (passed down to the `panel` function). `idsmode=NULL` (the default) means that only extreme data points are labelled while a non-NULL value adds labels to all data points (the default in Xpose 3). `xpose.panel.default` identifies extreme data points by fitting a loess smooth ($y \sim x$) and looking at the residuals from that fit. Points that are associated with the highest/lowest residuals are labelled. "High" and "low" are judged by the `panel` function parameter `idsext`, which gives the fraction of the total number of data points that are to be judged extreme in the "up" and "down" direction. The default value for `idsext` is 0.05 (see [link{xpose.prefs-class}](#)). There is also a possibility to label only the high or low extreme points. This is done through the `idsdir` argument to `xpose.panel.default`. A value of "both" (the default) means that both high and low extreme points are labelled while "up" and "down" labels the high and low extreme points respectively.

More graphical parameters may be passed to [xpose.panel.splom](#). for example, if you want to adjust the size of the `varnames` and `axis tick` labels you can use the parameters `varname.cex=0.5` and `axis.text.cex=0.5`.

Value

Returns a scatterplot matrix graph object.

Author(s)

E. Niclas Jonsson, Mats Karlsson, Andrew Hooker & Justin Wilkins

See Also

[xpose.panel.splom](#), [splom](#), [panel.splom](#), [xpose.prefs-class](#), [xpose.data-class](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
```

```
xpdb5 <- xpose.data(5)

## CL, WT, HT, SEX with a regression line
xpose.plot.splom(c("CL", "WT", "HT", "SEX"), xpdb5, lmline = TRUE)

## End(Not run)
```

```
xpose.prefs-class      Class "xpose.prefs"
```

Description

An object of the "xpose.prefs" class holds information about all the variable and graphical preferences for a particular "xpose.data" object.

Objects from the Class

Objects can be created by calls of the form `new("xpose.prefs", ...)` but this is usually not necessary since the "xpose.prefs" object is created at the same time as the "xpose.data" object.

Slots

Xvardef: Object of class "list" A list with the Xpose variable definitions. The values of the list are strings with the name of one or more columns in the data frames of Data or SData ([xpose.data-class](#)).

To get a listing of the current variable definitions in a particular "xpose.data" object type: `object@Prefs@Xvardef` on the command line (object is the name of the xpose.data object). These definitions have default values that are adjusted to reflect the contents of the Data slot in the xpose.data object when the object is initialized. Look in the `createXposeClasses` function for the default values.

To access a specific variable definition, use `object@Prefs@Xvardef[["vardef"]]`, where vardef is the name of the Xpose variable. You can also use the utility function `xvardef`.

Labels: Object of class "list" A list that translates column headers in Data and SData to labels to be used in graphs, e.g. "TIME" to "Time (h)". This is used in the labelling of axes, for example.

To get a listing of the current labels in a particular "xpose.data" object type: `object@Prefs@Labels` on the command line (object is the name of the xpose.data object). These definitions have default values that are adjusted to reflect the contents of the Data slot in the xpose.data object when the object is initialized. Default Labels that are not present in the NONMEM table files are deleted and columns that are present in the table files but that do not have default Label get a Label that is the same as the column name. Look in the `createXposeClasses` function for the default values.

To access a specific variable definition, use `object@Prefs@Label[["name"]]`, where name is the name of the column in the NONMEM table files. You can also use the utility function `xlabel` to access the labels.

The label for a particular column can also be changed using the `xlabel` function.

Graph.prefs: Object of class "list" This is a list with graphical parameters for the various elements in the Xpose graphs.

To get a listing of the current graphical preferences in a particular "xpose.data" object type: `object@Prefs@Graph.prefs` on the command line (object is the name of the xpose.data object).

Look in the `createXposeClasses` function for the default values and some comments.

Miss: Object of class "numeric" This is a scalar value that indicate missing values in the NONMEM table files. The default is -99. Can be accessed by `object@Prefs@Miss`.

Cat.levels: Object of class "numeric" This value is used when an "xpose.data" object is created. Any variable from a NONMEM table file that has fewer than or equal number of unque value tha `Cat.levels` are converted to factors. The default value is 4.

DV.Cat.levels: Object of class "numeric" This value is used when an "xpose.data" object is created. If a variable from a NONMEM table file with the name DV has fewer than or equal number of unque value than `DV.Cat.levels` are converted to a factor. The default value is 7.

Subset: Object of class "character" or "NULL" A string that indicates the subset of the data to use in a plot. An example is `SEX==1`. This string is used by the `Data` and `SData` functions when extracting the data to plot from an "xpose.data" object. Note that it is not used in the `subset` argument to `xyplo`t. The value of `Subset` can be accessed and changed using the `xsubset` function.

Gam.prefs: Object of class "list" This is a list containing preferences for the general additive model (GAM).

To get a listing of the current graphical preferences in a particular "xpose.data" object type: `object@Prefs@Gam.prefs` on the command line (object is the name of the xpose.data object).

Look in the `createXposeClasses` function for the default values, and see also `xpose.gam`.

Methods

No methods defined with class "xpose.prefs" in the signature.

Author(s)

Niclas Jonsson & Andrew Hooker

See Also

`xvardef`, `xlabel`, `xsubset`, `Data`, `SData`, `xpose.data`, `read.nm.tables`, `xpose.data-class`, `xpose.gam`

xpose.string.print *Print a pretty string.*

Description

Print a string with a certain number of characters per row.

Usage

```
xpose.string.print(value, fill = 60, file = "")
```

Arguments

value	The text to print.
fill	How wide should the text be per row.
file	Where to print. "" means to the screen.

Author(s)

Niclas Jonsson and Andrew C. Hooker

xpose.VPC *Visual Predictive Check (VPC) using XPOSE*

Description

This Function is used to create a VPC in xpose using the output from the vpc command in Pearl Speaks NONMEM (PsN). The function reads in the output files created by PsN and creates a plot from the data. The dependent variable, independent variable and conditioning variable are automatically determined from the PsN files.

Usage

```
xpose.VPC(vpc.info = "vpc_results.csv",
          vpctab = dir(pattern="^vpctab")[1],
          object = NULL,
          ids=FALSE,
          type="p",
          by=NULL,
          PI=NULL,
          PI.ci="area",
          PI.real=T,
          PI.ci.med.arcol="red",
          subset=NULL,
          main="Default",
```

```

main.sub=NULL,
main.sub.cex=0.85,
inclZeroWRES=FALSE,
force.x.continuous=FALSE,
funy=NULL,
logy=FALSE,
ylb = "Default",
verbose = FALSE,
...)
```

Arguments

vpc.info	The results file from the vpc command in PsN. for example 'vpc_results.csv', or if the file is in a separate directory './vpc_dir1/vpc_results.csv'.
vpctab	The 'vpctab' from the vpc command in PsN. For example 'vpctab5', or if the file is in a separate directory './vpc_dir1/vpctab5'. Can be NULL. The default looks in the current working directory and takes the first file that starts with 'vpctab' that it finds. Note that this default can result in the wrong files being read if there are multiple 'vpctab' files in the directory. One of object or vpctab is required. If both are present then the information from the vpctab will over-ride the xpose data object object (i.e. the values from the vpctab will replace any matching values in the object \@Data portion of the xpose data object).
object	An xpose data object. Created from xpose.data . One of object or vpctab is required. If both are present then the information from the vpctab will over-ride the xpose data object object (i.e. the values from the vpctab will replace any matching values in the object \@Data portion of the xpose data object).
ids	A logical value indicating whether text ID labels should be used as plotting symbols (the variable used for these symbols indicated by the idlab xpose data variable). Can be FALSE or TRUE.
type	Character string describing the way the points in the plot will be displayed. For more details, see plot . Use type="n" if you don't want observations in the plot.
by	A string or a vector of strings with the name(s) of the conditioning variables. For example by = c("SEX", "WT"). Because the function automatically determines the conditioning variable from the PsN input file specified in vpc.info, the by command can control if separate plots are created for each condition (by=NULL), or if a conditioning plot should be created (by="WT" for example). If the vpc.info file has a conditioning variable then by must match that variable. If there is no conditioning variable in vpc.info then the PI for each conditioned plot will be the PI for the entire data set (not only for the conditioning subset).
PI	Either "lines", "area" or "both" specifying whether prediction intervals (as lines, a shaded area or both) should be added to the plot. NULL means no prediction interval.
PI.ci	Plot the prediction interval of the simulated data's percentiles for each bin. Values can be "both", "area" or "lines" This can be thought of as a prediction interval about the PI.real or a confidence interval about the PI. However, note

	that with increasing number of simulations the CI will not go towards zero because the interval is also dependent on the size of the data set.
PI.real	Plot the percentiles of the real data in the various bins. values can be NULL or TRUE. Note that for a bin with few actual observations the percentiles will be approximate. For example, the 95th percentile of 4 data points will always be the largest of the 4 data points.
PI.ci.med.arcol	The color of the median PI.ci.
force.x.continuous	Logical value indicating whether x-values should be taken as continuous, even if categorical.
funy	String of function to apply to Y data. For example "abs"
logy	Logical value indicating whether the y-axis should be logarithmic, base 10.
ylb	Label for the y-axis
subset	A string giving the subset expression to be applied to the data before plotting. See xsubset .
main	A string giving the plot title or NULL if none. "Default" creates a default title.
main.sub	Used for names above each plot when using multiple plots. Should be a vector c("Group 1", "Group 2")
main.sub.cex	The size of the main.sub titles.
inclZeroWRES	Logical value indicating whether rows with WRES=0 is included in the plot.
verbose	Text messages passed to screen or not.
...	Other arguments passed to xpose.panel.default , xpose.plot.default and others. Please see these functions for more descriptions of what you can do.

Value

A plot or a list of plots.

Additional arguments

Additional graphical elements available in the VPC plots

PI.real = NULL or TRUE Plot the percentiles of the real data in the various bins. Note that for a bin with few actual observations the percentiles will be approximate. For example, the 95th percentile of 4 data points will always be the largest of the 4 data points.

PI.mirror = NULL, TRUE or AN.INTEGER.VALUE Plot the percentiles of one simulated data set in each bin. TRUE takes the first mirror from 'vpc_results.csv' and AN.INTEGER.VALUE can be 1, 2, ... n where n is the number of mirror's output in the 'vpc_results.csv' file.

PI.ci = "both", "area" or "lines" Plot the confidence interval for the simulated data's percentiles for each bin (for each simulated data set compute the percentiles for each bin, then, from all of the percentiles from all of the simulated datasets compute the 95% CI of these percentiles). These CIs can be used to asses the PI.real values for model misspecification.

Again, as with the `PI.real`, note that with few observations per bin the CIs will be approximate because the percentiles in each bin will be approximate. For example, the 95th percentile of 4 data points will always be the largest of the 4 data points.

`PI.limits = c(0.025, 0.975)` A vector of two values that describe the limits of the prediction interval that should be displayed. These limits should be found in the `'vpc_results.csv'` file. These limits are also used as the percentages for the `PI.real`, `PI.mirror` and `PI.ci`. However, the confidence interval in `PI.ci` is always the one defined in the `'vpc_results.csv'` file.

Additional options to control the look and feel of the PI. See See [grid.polygon](#) and [plot](#) for more details.

`PI.arcol` The color of the PI area

`PI.up.lty` The upper line type. can be "dotted" or "dashed", etc.

`PI.up.type` The upper type used for plotting. Defaults to a line.

`PI.up.col` The upper line color

`PI.up.lwd` The upper line width

`PI.down.lty` The lower line type. can be "dotted" or "dashed", etc.

`PI.down.type` The lower type used for plotting. Defaults to a line.

`PI.down.col` The lower line color

`PI.down.lwd` The lower line width

`PI.med.lty` The median line type. can be "dotted" or "dashed", etc.

`PI.med.type` The median type used for plotting. Defaults to a line.

`PI.med.col` The median line color

`PI.med.lwd` The median line width

Additional options to control the look and feel of the PI.ci. See See [grid.polygon](#) and [plot](#) for more details.

`PI.ci.up.arcol` The color of the upper PI.ci.

`PI.ci.med.arcol` The color of the median PI.ci.

`PI.ci.down.arcol` The color of the lower PI.ci.

`PI.ci.up.lty` The upper line type. can be "dotted" or "dashed", etc.

`PI.ci.up.type` The upper type used for plotting. Defaults to a line.

`PI.ci.up.col` The upper line color

`PI.ci.up.lwd` The upper line width

`PI.ci.down.lty` The lower line type. can be "dotted" or "dashed", etc.

`PI.ci.down.type` The lower type used for plotting. Defaults to a line.

`PI.ci.down.col` The lower line color

`PI.ci.down.lwd` The lower line width

PI.ci.med.lty The median line type. can be "dotted" or "dashed", etc.

PI.ci.med.type The median type used for plotting. Defaults to a line.

PI.ci.med.col The median line color

PI.ci.med.lwd The median line width

PI.ci.area.smooth Should the "area" for PI.ci be smoothed to match the "lines" argument? Allowed values are TRUE/FALSE. The "area" is set by default to show the bins used in the PI.ci computation. By smoothing, information is lost and, in general, the confidence intervals will be smaller than they are in reality.

Additional options to control the look and feel of the PI.real. See [grid.polygon](#) and [plot](#) for more details.

PI.real.up.lty The upper line type. can be "dotted" or "dashed", etc.

PI.real.up.type The upper type used for plotting. Defaults to a line.

PI.real.up.col The upper line color

PI.real.up.lwd The upper line width

PI.real.down.lty The lower line type. can be "dotted" or "dashed", etc.

PI.real.down.type The lower type used for plotting. Defaults to a line.

PI.real.down.col The lower line color

PI.real.down.lwd The lower line width

PI.real.med.lty The median line type. can be "dotted" or "dashed", etc.

PI.real.med.type The median type used for plotting. Defaults to a line.

PI.real.med.col The median line color

PI.real.med.lwd The median line width

Additional options to control the look and feel of the PI.mirror. See [plot](#) for more details.

PI.mirror.up.lty The upper line type. can be "dotted" or "dashed", etc.

PI.mirror.up.type The upper type used for plotting. Defaults to a line.

PI.mirror.up.col The upper line color

PI.mirror.up.lwd The upper line width

PI.mirror.down.lty The lower line type. can be "dotted" or "dashed", etc.

PI.mirror.down.type The lower type used for plotting. Defaults to a line.

PI.mirror.down.col The lower line color

PI.mirror.down.lwd The lower line width

PI.mirror.med.lty The median line type. can be "dotted" or "dashed", etc.

PI.mirror.med.type The median type used for plotting. Defaults to a line.

PI.mirror.med.col The median line color

PI.mirror.med.lwd The median line width

Author(s)

Andrew Hooker

See Also[read.vpctab](#) [read.npc.vpc.results](#) [xpose.panel.default](#) [xpose.plot.default](#)**Examples**

```
## Not run:
library(xpose4)

xpose.VPC()

## to be more clear about which files should be read in
vpc.file <- "vpc_results.csv"
vpctab <- "vpctab5"
xpose.VPC(vpc.info=vpc.file, vpctab=vpctab)

## with lines and a shaded area for the prediction intervals
xpose.VPC(vpc.file, vpctab=vpctab, PI="both")

## with the percentages of the real data
xpose.VPC(vpc.file, vpctab=vpctab, PI.real=T)

## with mirrors (if supplied in 'vpc.file')
xpose.VPC(vpc.file, vpctab=vpctab, PI.real=T, PI.mirror=5)

## with CIs
xpose.VPC(vpc.file, vpctab=vpctab, PI.real=T, PI.ci="area")
xpose.VPC(vpc.file, vpctab=vpctab, PI.real=T, PI.ci="area", PI=NULL)

## stratification (if 'vpc.file' is stratified)
cond.var <- "WT"
xpose.VPC(vpc.file, vpctab=vpctab)
xpose.VPC(vpc.file, vpctab=vpctab, by=cond.var)
xpose.VPC(vpctab=vpctab, vpc.info=vpc.file, PI="both", by=cond.var, type="n")

## with no data points in the plot
xpose.VPC(vpc.file, vpctab=vpctab, by=cond.var, PI.real=T, PI.ci="area", PI=NULL, type="n")

## with different DV and IDV, just read in new files and plot
vpc.file <- "vpc_results.csv"
vpctab <- "vpctab5"
cond.var <- "WT"
xpose.VPC(vpctab=vpctab, vpc.info=vpc.file, PI="both", by=cond.var)
xpose.VPC(vpctab=vpctab, vpc.info=vpc.file, PI="both")

## to use an xpose data object instead of vpctab
##
## In this example
## we expect to find the required NONMEM run and table files for run
```

```

## 5 in the current working directory
runnumber <- 5
xpdb <- xpose.data(runnumber)
xpose.VPC(vpc.file,object=xpdb)

## to read files in a directory different than the current working directory
vpc.file <- "../vpc_strat_WT_4_mirror_5/vpc_results.csv"
vpctab <- "../vpc_strat_WT_4_mirror_5/vpctab5"
xpose.VPC(vpc.info=vpc.file, vpctab=vpctab)

## to rearrange order of factors in VPC plot
xpdb@Data$SEX <- factor(xpdb@Data$SEX,levels=c("2","1"))
xpose.VPC(by="SEX",object=xpdb)

## End(Not run)

```

xpose.VPC.both	<i>Xpose Visual Predictive Check (VPC) for both continuous and Limit of Quantification data.</i>
----------------	--

Description

Xpose Visual Predictive Check (VPC) for both continuous and Below or Above Limit of Quantification (BLQ or ALQ) data.

Usage

```

xpose.VPC.both(vpc.info="vpc_results.csv",
               vpctab = dir(pattern="^vpctab")[1],
               object = NULL,
               subset=NULL,
               main="Default",
               main.sub=NULL,
               inclZeroWRES=FALSE,
               cont.logy=F,
               hline="default",
               add.args.cont=list(),
               add.args.cat=list(),
               ...)

```

Arguments

vpc.info	Name of PSN file to use. File will come from VPC command in PsN.
vpctab	Name of vpctab file produced from PsN.
object	Xpose data object.

<code>subset</code>	Subset of data to look at.
<code>main</code>	Title for plot.
<code>main.sub</code>	Used for names above each plot when using multiple plots. Should be a vector, e.g. <code>c("title 1", "title 2")</code> .
<code>inclZeroWRES</code>	Include WRES=0 rows in the computations for these plots?
<code>cont.logy</code>	Should the continuous plot y-axis be on the log scale?
<code>hline</code>	Horizontal line marking the limits of quantification. If they are defined, they must be a vector of values.
<code>add.args.cont</code>	Additional arguments to the continuous plot. xpose.VPC .
<code>add.args.cat</code>	Additional arguments to the categorical plot. xpose.VPC.categorical .
<code>...</code>	Additional arguments to both plots.

Author(s)

Andrew C. Hooker

See Also

[xpose.VPC](#), [xpose.VPC.categorical](#).

Examples

```
## Not run:
library(xpose4)

## move to the directory where results from PsN
## are found
cur.dir <- getwd()
setwd(paste(cur.dir, "/vpc_cont_LL0Q/", sep=""))

xpose.VPC()
xpose.VPC.categorical(censored=T)

xpose.VPC.both()

xpose.VPC.both(subset="DV>1.75")

xpose.VPC.both(add.args.cont=list(ylim=c(0,80)))

xpose.VPC.both(add.args.cont = list(ylim = c(0.01, 80)), xlim = c(0,
  40), add.args.cat = list(ylim = c(0, 0.4)), cont.logy = T)

xpose.VPC.both(cont.logy=T)

## End(Not run)
```

xpose.VPC.categorical *Xpose visual predictive check for categorical data.*

Description

Xpose visual predictive check for categorical data (binary, ordered categorical and count data).

Usage

```
xpose.VPC.categorical(vpc.info="vpc_results.csv",
  vpctab = dir(pattern="^vpctab")[1],
  object = NULL,
  subset=NULL,
  main="Default",
  main.sub="Default",
  main.sub.cex=0.85,
  real.col=4,
  real.lty="b",
  real.cex=1,
  real.lwd=1,
  median.line=FALSE,
  median.col="darkgrey",
  median.lty=1,
  ci.lines=FALSE,
  ci.col="blue",
  ci.lines.col="darkblue",
  ci.lines.lty=3,
  xlb="Default",
  ylb="Proportion of Total",
  force.x.continuous=FALSE,
  level.to.plot=NULL,
  max.plots.per.page=1,
  rug=TRUE,
  rug.col="orange",
  censored=FALSE,
  ...)
```

Arguments

vpc.info	Name of PSN file to use. File will come from VPC command in PsN.
vpctab	Name of vpctab file produced from PsN.
object	Xpose data object.
subset	Subset of data to look at.
main	Title for plot.
main.sub	Used for names above each plot when using multiple plots. Should be a vector, e.g. c("title 1", "title 2").

main.sub.cex	Size of main.sub
real.col	Color of real line.
real.lty	Real line type.
real.cex	Size of real line.
real.lwd	Width of real line.
median.line	Draw a median line?
median.col	Color of median line.
median.lty	median line type.
ci.lines	Lines marking confidence interval?
ci.col	Color of CI area.
ci.lines.col	Color of CI lines.
ci.lines.lty	Type of CI lines.
xlb	X-axis label. If other than "default" passed directly to xyplot .
ylb	Y-axis label. Passed directly to xyplot .
force.x.continuous	For the x variable to be continuous.
level.to.plot	Which levels of the variable to plot. Smallest level is 1, largest is number_of_levels. For example, with 4 levels, the largest level would be 4, the smallest would be 1.
max.plots.per.page	The number of plots per page.
rug	Should there be markings on the plot showing where the intervals for the VPC are?
rug.col	Color of the rug.
censored	Is this censored data? Censored data can be both below and above the limit of quantification.
...	Additional information passed to function.

Author(s)

Andrew C. Hooker

See Also

[xpose.VPC.both](#).

Examples

```
## Not run:
library(xpose4)

## move to the directory where results from PsN
## are found
cur.dir <- getwd()
```



```
setwd(paste(cur.dir, "/binary/vpc_36", sep=""))

xpose.VPC.categorical(level.to.plot=1,max.plots.per.page=4)
xpose.VPC.categorical(level.to.plot=1,max.plots.per.page=4,by="DOSE")

## ordered categorical plots
setwd(paste(cur.dir, "/ordered_cat/vpc_45", sep=""))
xpose.VPC.categorical()

## count
setwd(paste(cur.dir, "/count/vpc65b", sep=""))
xpose.VPC.categorical()

setwd(paste(cur.dir, "/count/vpc65a", sep=""))
xpose.VPC.categorical()

## End(Not run)
```

```
xpose.yscale.components.log10
```

Functions to create nice looking axes when using Log scales.

Description

The functions are used to create standard tic marks and axis labels when the axes are on the log scale.

Usage

```
xpose.yscale.components.log10(lim, ...)
xpose.xscale.components.log10(lim, ...)
xpose.logTicks(lim, loc = c(1, 5))
```

Arguments

lim	Limits
loc	Locations
...	Additional arguments passed to the function.

Details

These functions create log scales that look like they should (not the default R scales). These functions are used as input to the `xscale.components` argument in a lattice plot.

Author(s)

Andrew Hooker

See Also

[xpose.plot.default](#) [xscale.components](#)

Examples

```
## Not run:
xpdb5 <- xpose.data(5)
xpose.plot.default("PRED", "DV", xpdb, logy=T, logx=T)
xpose.plot.default("PRED", "DV", xpdb, logy=T, logx=T,
                   yscale.components = xpose.yscale.components.log10,
                   xscale.components = xpose.xscale.components.log10)

## both give the same result

## End(Not run)
```

xsubset

Extract or set the value of the Subset slot.

Description

Extract or set the value of the Subset slot of an "xpose.data" object.

Usage

```
xsubset(object)
xsubset(object) <- value
```

Arguments

object	An "xpose.data" object.
value	A string with the subset expression.

Details

The subset string has the same syntax as the subset argument to, e.g. `panel.xyplot`. Note, however, that the "xpose.data" subset is not used as an argument to `panel.xyplot`. It is intended as the subset argument to the `Data` and `SData` functions.

Value

A string representing the subset expression.

Author(s)

Niclas Jonsson

See Also[Data, SData](#)**Examples**

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

xsubset(xpdb5) <- "DV > 0"
xsubset(xpdb5)

## End(Not run)
```

`xvardef`*Extracts and set Xpose variable definitions.*

Description

This function extracts and set Xpose variable definitions in "xpose.data" objects.

Usage

```
xvardef(x, object)

xvardef(object) <- value
```

Arguments

<code>x</code>	The name of an xpose variable (see below).
<code>object</code>	An <code>xpose.data</code> object.
<code>value</code>	A two element vector of which the first element is the name of the variable and the second the column name in the Data slot of the object.

Details

The Xpose variable definitions are used to map particular variable types to column names in the data.frame in the Data slot of the "xpose.data" object. The single-valued Xpose variable definitions are: `id`, `idlab`, `idv`, `occ`, `dv`, `pred`, `ipred`, `iwres`, `res`. The (potentially) vector-valued Xpose variable definitions are: `parms`, `covariates`, `ranpar`, `tvparms` (parameters, covariates, random effects parameters=etas, typical value parameters). The default values of these can be found in the `createXposeClasses` function.

Value

Returns a string with the name of the data variable defined as the Xpose data variable.

Author(s)

Niclas Jonsson

See Also

[xpose.data-class](#), [xpose.prefs-class](#)

Examples

```
## Not run:
## xpdb5 is an Xpose data object
## We expect to find the required NONMEM run and table files for run
## 5 in the current working directory
xpdb5 <- xpose.data(5)

## get the column name in the Data slot of object xpdb5
## corresponding to the label dv
xvardef("dv", xpdb5)

## reset the which column the label dv points to in the Data slot of
## object xpdb5
xvardef(xpdb5) <- c("dv", "DVA")

## End(Not run)
```

Index

*Topic **\textasciitildebootgam**

- bootgam.print, 52
- xp.distr.mod.size, 187
- xp.dofv.plot, 188
- xp.inc.prob, 189
- xp.inc.prob.comb.2, 189
- xp.inc.stab.cov, 190
- xp.incl.index.cov, 191
- xp.incl.index.cov.comp, 192
- xp.incl.index.cov.ind, 192

*Topic **\textasciitildebootscm**

- bootgam.print, 52
- bootscm.import, 53
- xp.boot.par.est, 185
- xp.boot.par.est.corr, 186
- xp.distr.mod.size, 187
- xp.dofv.plot, 188
- xp.inc.prob, 189
- xp.inc.prob.comb.2, 189
- xp.inc.stab.cov, 190
- xp.incl.index.cov, 191
- xp.incl.index.cov.comp, 192
- xp.incl.index.cov.ind, 192

*Topic **classes**

- xpose.data-class, 198
- xpose.prefs-class, 245

*Topic **datasets**

- simpraz.xpdb, 168

*Topic **methods**

- absval.cwres.vs.cov.bw, 17
- absval.cwres.vs.pred, 18
- absval.cwres.vs.pred.by.cov, 20
- absval.delta.model.comp, 22
- absval.iwres.cwres.vs.ipred.pred, 24
- absval.iwres.vs.ipred, 25
- absval.iwres.vs.ipred.by.cov, 27
- absval.iwres.vs.pred, 29
- absval.wres.vs.cov.bw, 30

- absval.wres.vs.idv, 32
- absval.wres.vs.pred, 34
- absval.wres.vs.pred.by.cov, 35
- add.functions, 37
- add.grid.table, 39
- add.model.comp, 41
- addit.gof, 42
- autocorr.cwres, 44
- autocorr.wres, 46
- basic.gof, 47
- basic.model.comp, 49
- boot.hist, 51
- cat.dv.vs.idv.sb, 54
- cat.pc, 56
- change.graphical.parameters, 58
- change.misc.parameters, 60
- change.parm, 63
- change.var.name, 64
- change.xlabel, 65
- change.xvardef, 66
- classic, 68
- compute.cwres, 68
- create.mirror, 77
- create.xpose.plot.classes, 78
- createXposeClasses, 78
- cwres.dist.hist, 79
- cwres.dist.qq, 80
- cwres.vs.cov, 81
- cwres.vs.idv, 82
- cwres.vs.idv.bw, 84
- cwres.vs.pred, 85
- cwres.vs.pred.bw, 87
- cwres.wres.vs.idv, 88
- Data, 90
- data.checkout, 92
- db.names, 93
- dist.hist, 94
- dist.qq, 96
- dOFV.vs.cov, 97

- dOFV.vs.id, 99
- dOFV1.vs.dOFV2, 101
- dv.preds.vs.idv, 102
- dv.vs.idv, 103
- dv.vs.ipred, 105
- dv.vs.ipred.by.cov, 106
- dv.vs.ipred.by.idv, 108
- dv.vs.pred, 109
- dv.vs.pred.by.cov, 110
- dv.vs.pred.by.idv, 112
- dv.vs.pred.ipred, 113
- export.graph.par, 115
- export.variable.definitions, 116
- functions, 117
- gam.functions, 118
- generic, 121
- gof, 122
- import.graph.par, 124
- import.variable.definitions, 126
- ind.plots, 127
- ind.plots.wres.hist, 130
- ind.plots.wres.qq, 133
- internal.functions, 135
- ipred.vs.idv, 136
- iwres.dist.hist, 138
- iwres.dist.qq, 139
- iwres.vs.idv, 140
- kaplan.plot, 141
- make.sb.data, 145
- npc.coverage, 146
- nsim, 148
- parm.splom, 149
- parm.vs.cov, 151
- parm.vs.parm, 153
- pred.vs.idv, 154
- print.xpose.multiple.plot, 155
- randtest.hist, 156
- ranpar.vs.cov, 157
- Read NONMEM output files, 158
- read.nm.tables, 160
- read.npc.vpc.results, 162
- read.TTE.sim.data, 163
- read.vpctab, 164
- reset, 165
- runsum, 166
- simprazExample, 171
- summary, 171
- tables, 173
- tabulate.parameters, 174
- wres.dist.hist, 175
- wres.dist.qq, 176
- wres.vs.cov, 177
- wres.vs.idv, 178
- wres.vs.idv.bw, 180
- wres.vs.pred, 181
- wres.vs.pred.bw, 182
- xlabel, 184
- xpose.ask.for.filename, 193
- xpose.create.title, 194
- xpose.create.title.text, 195
- xpose.data, 196
- xpose.dev.new, 200
- xpose.gam, 200
- xpose.license.citation, 203
- xpose.multiple.plot, 203
- xpose.multiple.plot.default, 205
- xpose.panel.bw, 206
- xpose.panel.default, 209
- xpose.panel.histogram, 222
- xpose.panel.qq, 225
- xpose.panel.splom, 226
- xpose.plot.bw, 228
- xpose.plot.default, 231
- xpose.plot.histogram, 236
- xpose.plot.qq, 239
- xpose.plot.splom, 242
- xpose.string.print, 247
- xpose.VPC, 247
- xpose.VPC.both, 253
- xpose.VPC.categorical, 255
- xpose.yscale.components.log10, 257
- xpose4-package, 5
- xpose4classic-package, 7
- xpose4data-package, 9
- xpose4generic-package, 11
- xpose4psn-package, 12
- xpose4specific-package, 13
- xsubset, 258
- xvardef, 259
- *Topic **package**
 - xpose4-package, 5
 - xpose4classic-package, 7
 - xpose4data-package, 9
 - xpose4generic-package, 11
 - xpose4psn-package, 12
 - xpose4specific-package, 13

- absval.cwres.vs.cov.bw, [14](#), [17](#)
- absval.cwres.vs.pred, [14](#), [18](#), [21](#)
- absval.cwres.vs.pred.by.cov, [14](#), [20](#)
- absval.dcwres.vs.cov.model.comp, [14](#)
- absval.dcwres.vs.cov.model.comp
 - (absval.delta.model.comp), [22](#)
- absval.delta.model.comp, [22](#)
- absval.dipred.vs.cov.model.comp, [14](#)
- absval.dipred.vs.cov.model.comp
 - (absval.delta.model.comp), [22](#)
- absval.diwres.vs.cov.model.comp, [14](#)
- absval.diwres.vs.cov.model.comp
 - (absval.delta.model.comp), [22](#)
- absval.dpred.vs.cov.model.comp, [14](#)
- absval.dpred.vs.cov.model.comp
 - (absval.delta.model.comp), [22](#)
- absval.dwres.vs.cov.model.comp, [14](#)
- absval.dwres.vs.cov.model.comp
 - (absval.delta.model.comp), [22](#)
- absval.iwres.cwres.vs.ipred.pred, [14](#), [24](#)
- absval.iwres.vs.cov.bw, [14](#)
- absval.iwres.vs.cov.bw
 - (absval.wres.vs.cov.bw), [30](#)
- absval.iwres.vs.idv, [14](#)
- absval.iwres.vs.idv
 - (absval.iwres.vs.pred), [29](#)
- absval.iwres.vs.ipred, [14](#), [25](#), [25](#), [28](#), [48](#)
- absval.iwres.vs.ipred.by.cov, [14](#), [27](#)
- absval.iwres.vs.pred, [14](#), [29](#)
- absval.iwres.wres.vs.ipred.pred, [14](#)
- absval.iwres.wres.vs.ipred.pred
 - (absval.iwres.cwres.vs.ipred.pred), [24](#)
- absval.wres.vs.cov.bw, [14](#), [30](#)
- absval.wres.vs.idv, [14](#), [32](#)
- absval.wres.vs.pred, [14](#), [25](#), [34](#), [36](#)
- absval.wres.vs.pred.by.cov, [14](#), [35](#)
- add.absval, [9](#)
- add.absval (add.functions), [37](#)
- add.dichot, [9](#)
- add.dichot (add.functions), [37](#)
- add.exp, [9](#)
- add.exp (add.functions), [37](#)
- add.functions, [37](#)
- add.grid.table, [11](#), [39](#)
- add.grid.text, [11](#)
- add.grid.text (add.grid.table), [39](#)
- add.log, [9](#)
- add.log (add.functions), [37](#)
- add.model.comp, [14](#), [41](#)
- add.modify.db.items.menu, [8](#)
- add.modify.db.items.menu (classic), [68](#)
- add.tad, [9](#)
- add.tad (add.functions), [37](#)
- addid, [11](#)
- addid (generic), [121](#)
- addit.gof, [14](#), [42](#)
- another.menu, [8](#)
- another.menu (classic), [68](#)
- autocorr.cwres, [14](#), [44](#)
- autocorr.iwres, [14](#)
- autocorr.iwres (autocorr.wres), [46](#)
- autocorr.wres, [14](#), [46](#)
- basic.gof, [14](#), [47](#), [206](#)
- basic.model.comp, [14](#), [49](#)
- boot.hist, [13](#), [51](#)
- bootgam.print, [13](#), [14](#), [52](#)
- bootscm.import, [9](#), [13](#), [53](#)
- bwplot, [18](#), [31](#), [85](#), [88](#), [180](#), [183](#), [228–230](#)
- calc.npar, [9](#)
- calc.npar (Read NONMEM output files), [158](#)
- cat.dv.vs.idv.sb, [14](#), [54](#)
- cat.pc, [13](#), [14](#), [56](#)
- categorical.table, [11](#)
- categorical.table (tables), [173](#)
- change.ab.graph.par, [9](#)
- change.ab.graph.par
 - (change.graphical.parameters), [58](#)
- change.bw.graph.par, [9](#)
- change.bw.graph.par
 - (change.graphical.parameters), [58](#)
- change.cat.cont, [9](#)
- change.cat.cont
 - (change.misc.parameters), [60](#)
- change.cat.cont<-
 - (change.misc.parameters), [60](#)
- change.cat.levels, [9](#)
- change.cat.levels
 - (change.misc.parameters), [60](#)
- change.cat.levels<-
 - (change.misc.parameters), [60](#)

- change.cond.graph.par, 9
- change.cond.graph.par
(change.graphical.parameters),
58
- change.dil.graph.par, 9
- change.dil.graph.par
(change.graphical.parameters),
58
- change.dispersion, 8
- change.dispersion(classic), 68
- change.dv.cat.levels, 9
- change.dv.cat.levels
(change.misc.parameters), 60
- change.dv.cat.levels<-
(change.misc.parameters), 60
- change.graphical.par, 8
- change.graphical.par(classic), 68
- change.graphical.parameters, 58
- change.label.par, 9
- change.label.par
(change.graphical.parameters),
58
- change.lm.graph.par, 9
- change.lm.graph.par
(change.graphical.parameters),
58
- change.medianNorm, 8
- change.medianNorm(classic), 68
- change.misc.graph.par, 9
- change.misc.graph.par
(change.graphical.parameters),
58
- change.misc.parameters, 60
- change.miss, 9
- change.miss(change.misc.parameters), 60
- change.onlyfirst, 8
- change.onlyfirst(classic), 68
- change.parm, 9, 63
- change.pi.graph.par, 9
- change.pi.graph.par
(change.graphical.parameters),
58
- change.smooth.graph.par, 9
- change.smooth.graph.par
(change.graphical.parameters),
58
- change.steppit, 8
- change.steppit(classic), 68
- change.subset, 9
- change.subset(change.misc.parameters),
60
- change.use.ids, 8
- change.use.ids(classic), 68
- change.var.name, 9, 64
- change.xlabel, 9, 65
- change.xp.obj, 8
- change.xp.obj(classic), 68
- change.xvardef, 9, 66, 165
- change.xvardef<-(change.xvardef), 66
- check.gamobj, 14
- check.gamobj(gam.functions), 118
- check.vars, 9
- check.vars(functions), 117
- classic, 68
- compute.cwres, 11, 17–19, 21, 23, 33, 42, 45,
48, 50, 68, 79, 80, 82, 83, 85–89,
161, 176, 182, 198
- computePI, 11
- computePI(generic), 121
- continuous.table, 11
- continuous.table(tables), 173
- cov.hist, 14
- cov.hist(dist.hist), 94
- cov.qq, 14
- cov.qq(dist.qq), 96
- cov.splom, 14
- cov.splom(parm.splom), 149
- cov.summary, 14
- cov.summary(summary), 171
- covariate.model.menu, 8
- covariate.model.menu(classic), 68
- create.mirror, 11, 77, 230, 234, 238, 241,
244
- create.parameter.list, 9
- create.parameter.list(Read NONMEM
output files), 158
- create.rand, 11
- create.rand(generic), 121
- create.strat.rand, 11
- create.strat.rand(generic), 121
- create.xpose.plot.classes, 11, 78
- createXposeClasses, 9, 78
- cwres.dist.hist, 15, 79
- cwres.dist.qq, 15, 80
- cwres.menu, 8
- cwres.menu(classic), 68

- cwres.vs.cov, [15](#), [81](#)
- cwres.vs.idv, [15](#), [48](#), [82](#)
- cwres.vs.idv.bw, [15](#), [84](#)
- cwres.vs.pred, [15](#), [85](#)
- cwres.vs.pred.bw, [15](#), [87](#)
- cwres.wres.vs.idv, [15](#), [88](#)
- cwres.wres.vs.pred, [15](#)
- cwres.wres.vs.pred (cwres.wres.vs.idv), [88](#)

- Data, [10](#), [62](#), [64](#), [65](#), [90](#), [173](#), [198](#), [199](#), [246](#), [259](#)
- data.checkout, [15](#), [92](#)
- data.checkout.menu, [8](#)
- data.checkout.menu (classic), [68](#)
- Data<- (Data), [90](#)
- db.names, [10](#), [93](#)
- dev.new, [200](#)
- dget, [126](#)
- dist.hist, [94](#)
- dist.qq, [96](#)
- documentation.menu, [8](#)
- documentation.menu (classic), [68](#)
- dOFV.vs.cov, [15](#), [97](#)
- dOFV.vs.id, [15](#), [99](#)
- dOFV1.vs.dOFV2, [15](#), [101](#)
- dotplot, [93](#), [121](#)
- dput, [116](#), [117](#)
- dv.preds.vs.idv, [15](#), [102](#)
- dv.vs.idv, [15](#), [103](#), [103](#)
- dv.vs.ipred, [15](#), [48](#), [105](#), [107](#), [108](#), [114](#)
- dv.vs.ipred.by.cov, [15](#), [106](#)
- dv.vs.ipred.by.idv, [15](#), [108](#)
- dv.vs.pred, [15](#), [48](#), [109](#), [111](#), [113](#), [114](#)
- dv.vs.pred.by.cov, [15](#), [110](#)
- dv.vs.pred.by.idv, [15](#), [112](#)
- dv.vs.pred.ipred, [15](#), [113](#)

- eq.xpose, [11](#)
- eq.xpose (generic), [121](#)
- export.graph.par, [10](#), [59](#), [115](#), [125](#)
- export.variable.definitions, [10](#), [115](#), [116](#), [125](#), [126](#)

- find.right.table, [11](#)
- find.right.table (internal.functions), [135](#)
- fix.wrapped.lines, [10](#)
- fix.wrapped.lines (Read NONMEM output files), [158](#)
- functions, [117](#)

- gam, [120](#), [121](#)
- gam.change.weights, [8](#)
- gam.change.weights (classic), [68](#)
- gam.functions, [118](#)
- gam.menu, [8](#)
- gam.menu (classic), [68](#)
- gam.settings.menu, [8](#)
- gam.settings.menu (classic), [68](#)
- generic, [121](#)
- get.doc, [10](#)
- get.doc (change.misc.parameters), [60](#)
- get.polygon.regions, [11](#)
- get.polygon.regions (internal.functions), [135](#)
- get.refrunno, [11](#)
- get.refrunno (generic), [121](#)
- gof, [15](#), [122](#)
- gof.menu, [8](#)
- gof.menu (classic), [68](#)
- gofSetup, [15](#)
- gofSetup (gof), [122](#)
- grid.polygon, [147](#), [250](#), [251](#)
- grid.text, [41](#)

- histogram, [52](#), [79](#), [82](#), [95](#), [131](#), [132](#), [138](#), [152](#), [156–158](#), [175](#), [178](#), [237–239](#)

- import.graph.par, [10](#), [59](#), [116](#), [124](#), [165](#)
- import.variable.definitions, [10](#), [115](#), [117](#), [125](#), [126](#)
- ind.cwres, [11](#)
- ind.cwres (compute.cwres), [68](#)
- ind.plots, [15](#), [127](#)
- ind.plots.cwres.hist, [15](#)
- ind.plots.cwres.hist (ind.plots.wres.hist), [130](#)
- ind.plots.cwres.qq, [15](#)
- ind.plots.cwres.qq (ind.plots.wres.qq), [133](#)
- ind.plots.wres.hist, [15](#), [130](#)
- ind.plots.wres.qq, [15](#), [133](#)
- internal.functions, [135](#)
- ipred.vs.idv, [15](#), [103](#), [136](#)
- is.cwres.readable.file, [11](#)

- is.cwres.readable.file (compute.cwres), 68
- is.readable.file, 10
- is.readable.file (functions), 117
- iwres.dist.hist, 15, 138
- iwres.dist.qq, 15, 139
- iwres.vs.idv, 15, 43, 140

- kaplan.plot, 13, 15, 141

- lattice-package, 52, 156, 157
- list.gam.settings, 8
- list.gam.settings (classic), 68
- lm, 59
- loess, 59
- loess.smooth, 59

- main.menu, 8
- main.menu (classic), 68
- make.sb.data, 10, 145
- manage.db, 8
- manage.db (classic), 68
- model.comparison.covariates.menu, 8
- model.comparison.covariates.menu (classic), 68
- model.comparison.menu, 8
- model.comparison.menu (classic), 68

- npc.coverage, 13, 146, 163
- nsim, 10, 91, 148
- nsim<- (nsim), 148

- panel.abline, 59, 83, 86, 89, 105, 106, 108, 109, 111, 112, 114, 140, 179, 181, 215, 223
- panel.bwplot, 59, 208, 216
- panel.histogram, 239
- panel.lmline, 59
- panel.loess, 59, 216, 227
- panel.pairs, 227
- panel.qqmathline, 225, 241
- panel.splom, 227, 244
- panel.xyplot, 229, 235
- par, 59
- parameters.menu, 8
- parameters.menu (classic), 68
- parm.hist, 15
- parm.hist (dist.hist), 94
- parm.qq, 15
- parm.qq (dist.qq), 96
- parm.splom, 15, 149
- parm.summary, 16
- parm.summary (summary), 171
- parm.vs.cov, 16, 151, 206
- parm.vs.parm, 16, 153, 206
- plot, 44, 46, 147, 248, 250, 251
- plot.default, 59
- points, 134, 215, 227
- pred.vs.idv, 16, 103, 154
- preferences.menu, 8
- preferences.menu (classic), 68
- print,xpose.multiple.plot-method (create.xpose.plot.classes), 78
- print.char.matrix, 172, 173
- print.xpose.multiple.plot, 11, 155, 204

- qqmath, 80, 97, 135, 139, 176, 225, 239–241, 244
- qqplot, 135

- randtest.hist, 13, 156
- ranpar.hist, 16
- ranpar.hist (dist.hist), 94
- ranpar.qq, 16
- ranpar.qq (dist.qq), 96
- ranpar.splom, 16
- ranpar.splom (parm.splom), 149
- ranpar.vs.cov, 16, 157
- Read NONMEM output files, 158
- read.cwres.data, 11
- read.cwres.data (compute.cwres), 68
- read.lst, 10
- read.lst (Read NONMEM output files), 158
- read.lst6, 10
- read.lst6 (Read NONMEM output files), 158
- read.lst7, 10
- read.lst7 (Read NONMEM output files), 158
- read.nm.tables, 10, 160, 198, 199, 246
- read.npc.vpc.results, 10, 148, 162, 217, 252
- read.phi, 10
- read.phi (Read NONMEM output files), 158
- read.table, 92, 93
- read.TTE.sim.data, 10, 163
- read.vpctab, 10, 164, 252
- reset, 165

- reset.graph.par, [10](#)
- reset.variables, [10](#)
- residual.diagnostics.menu, [8](#)
- residual.diagnostics.menu (classic), [68](#)
- runsum, [16](#), [26](#), [41](#), [166](#)
- runsum.print, [8](#)
- runsum.print (classic), [68](#)

- SData, [10](#), [62](#), [64](#), [65](#), [173](#), [198](#), [199](#), [246](#), [259](#)
- SData (Data), [90](#)
- SData<- (Data), [90](#)
- set.doc, [10](#)
- set.doc (change.misc.parameters), [60](#)
- setup.PPI, [11](#), [217](#)
- setup.PPI (internal.functions), [135](#)
- shingle, [59](#)
- simpraz.xpdb, [10](#), [168](#)
- simprazExample, [10](#), [170](#), [171](#)
- splom, [150](#), [227](#), [244](#)
- sqrtn, [11](#)
- sqrtn (compute.cwres), [68](#)
- step.gam, [202](#)
- strip.custom, [234](#)
- strip.default, [129](#)
- struct.gof.menu, [8](#)
- struct.gof.menu (classic), [68](#)
- structural.diagnostics.menu, [8](#)
- structural.diagnostics.menu (classic), [68](#)
- subset, [62](#)
- summary, [171](#)
- Surv, [144](#)
- survfit, [143](#), [144](#)

- table.file.read.settings.menu, [8](#)
- table.file.read.settings.menu (classic), [68](#)
- tables, [173](#)
- tabulate.parameters, [16](#), [174](#)
- test.xpose.data, [10](#)
- test.xpose.data (functions), [117](#)
- trellis.par.get, [208](#), [209](#), [216](#), [217](#)

- vpc.npc.menu, [8](#)
- vpc.npc.menu (classic), [68](#)

- wres.dist.hist, [16](#), [175](#)
- wres.dist.qq, [16](#), [176](#)
- wres.vs.cov, [16](#), [177](#)
- wres.vs.idv, [16](#), [43](#), [48](#), [178](#)
- wres.vs.idv.bw, [16](#), [180](#)
- wres.vs.pred, [16](#), [43](#), [181](#)
- wres.vs.pred.bw, [16](#), [182](#)

- xlabel, [10](#), [184](#), [245](#), [246](#)
- xlabel<- (xlabel), [184](#)
- xp.akaike.plot, [16](#)
- xp.akaike.plot (gam.functions), [118](#)
- xp.boot.par.est, [13](#), [16](#), [185](#)
- xp.boot.par.est.corr, [13](#), [16](#), [186](#)
- xp.check.scope, [16](#)
- xp.check.scope (gam.functions), [118](#)
- xp.cook, [16](#)
- xp.cook (gam.functions), [118](#)
- xp.distr.mod.size, [13](#), [16](#), [187](#)
- xp.dofv.plot, [13](#), [16](#), [188](#)
- xp.gam, [16](#), [202](#)
- xp.gam (gam.functions), [118](#)
- xp.get.disp, [16](#)
- xp.get.disp (xpose.gam), [200](#)
- xp.inc.prob, [13](#), [16](#), [189](#)
- xp.inc.prob.comb.2, [13](#), [16](#), [189](#)
- xp.inc.stab.cov, [13](#), [16](#), [190](#)
- xp.incl.index.cov, [13](#), [16](#), [191](#)
- xp.incl.index.cov.comp, [13](#), [16](#), [192](#)
- xp.incl.index.cov.ind, [13](#), [16](#), [192](#)
- xp.ind.inf.fit, [16](#)
- xp.ind.inf.fit (gam.functions), [118](#)
- xp.ind.inf.terms, [16](#)
- xp.ind.inf.terms (gam.functions), [118](#)
- xp.ind.stud.res, [16](#)
- xp.ind.stud.res (gam.functions), [118](#)
- xp.plot, [16](#)
- xp.plot (gam.functions), [118](#)
- xp.scope3, [16](#)
- xp.scope3 (gam.functions), [118](#)
- xp.summary, [17](#)
- xp.summary (gam.functions), [118](#)
- xpose (xpose4-package), [5](#)
- xpose.ask.for.filename, [10](#), [193](#)
- xpose.ask.for.lst, [10](#)
- xpose.ask.for.lst (xpose.ask.for.filename), [193](#)
- xpose.ask.for.mod, [10](#)
- xpose.ask.for.mod (xpose.ask.for.filename), [193](#)
- xpose.bin, [10](#)
- xpose.bin (functions), [117](#)

- xpose.calculate.cwres, [11](#)
- xpose.calculate.cwres (compute.cwres), [68](#)
- xpose.create.label, [11](#)
- xpose.create.label (xpose.create.title), [194](#)
- xpose.create.title, [11](#), [194](#)
- xpose.create.title.hist, [12](#)
- xpose.create.title.text, [12](#), [195](#)
- xpose.data, [10](#), [38](#), [59](#), [62](#), [64](#), [65](#), [67](#), [93](#), [94](#), [125](#), [126](#), [164](#), [165](#), [170](#), [184](#), [194](#), [196](#), [196](#), [199](#), [246](#), [248](#)
- xpose.data-class, [198](#), [245](#)
- xpose.dev.new, [12](#), [200](#)
- xpose.gam, [17](#), [200](#), [246](#)
- xpose.license.citation, [17](#), [203](#)
- xpose.logTicks, [12](#)
- xpose.logTicks (xpose.yscale.components.log10), [257](#)
- xpose.multiple.plot, [12](#), [144](#), [155](#), [156](#), [203](#)
- xpose.multiple.plot.default, [12](#), [43](#), [146](#), [148](#), [204](#), [205](#)
- xpose.multiple.plot.title, [12](#), [17](#), [21](#), [23](#), [24](#), [28](#), [31](#), [36](#), [41](#), [43](#), [45](#), [46](#), [48](#), [50](#), [81](#), [95](#), [96](#), [107](#), [111](#), [128](#), [131](#), [133](#), [152](#), [153](#), [157](#), [172](#), [177](#)
- xpose.multiple.plot.title (xpose.create.title), [194](#)
- xpose.panel.bw, [12](#), [18](#), [31](#), [59](#), [85](#), [87](#), [88](#), [180](#), [183](#), [206](#), [229](#), [230](#)
- xpose.panel.default, [12](#), [19](#), [21](#), [23](#), [25](#), [26](#), [30](#), [33](#), [35](#), [36](#), [42](#), [43](#), [48](#), [50](#), [59](#), [83](#), [86](#), [89](#), [102–105](#), [107](#), [108](#), [110](#), [111](#), [113](#), [114](#), [129](#), [137](#), [140](#), [141](#), [152–155](#), [158](#), [179](#), [181](#), [209](#), [234](#), [235](#), [244](#), [249](#), [252](#)
- xpose.panel.histogram, [12](#), [52](#), [79](#), [95](#), [132](#), [138](#), [156](#), [157](#), [175](#), [222](#), [239](#)
- xpose.panel.qq, [12](#), [80](#), [97](#), [135](#), [139](#), [176](#), [225](#), [241](#)
- xpose.panel.splom, [12](#), [150](#), [226](#), [244](#)
- xpose.plot.bw, [12](#), [17](#), [18](#), [31](#), [59](#), [85](#), [87](#), [88](#), [180](#), [183](#), [208](#), [228](#)
- xpose.plot.default, [12](#), [19](#), [21](#), [23–26](#), [28–30](#), [32–36](#), [42](#), [43](#), [45](#), [46](#), [48](#), [50](#), [59](#), [77](#), [82](#), [83](#), [86](#), [89](#), [102–105](#), [107](#), [108](#), [110](#), [111](#), [113](#), [114](#), [128](#), [129](#), [137](#), [140](#), [141](#), [152–155](#), [158](#), [177–179](#), [181](#), [182](#), [231](#), [249](#), [252](#), [258](#)
- xpose.plot.histogram, [12](#), [52](#), [77](#), [79](#), [82](#), [95](#), [130–132](#), [138](#), [150](#), [152](#), [156–158](#), [175](#), [177](#), [178](#), [236](#), [238](#)
- xpose.plot.qq, [12](#), [77](#), [80](#), [96](#), [97](#), [133–135](#), [139](#), [176](#), [225](#), [239](#), [241](#)
- xpose.plot.splom, [12](#), [77](#), [150](#), [227](#), [242](#)
- xpose.prefs-class, [245](#)
- xpose.print, [10](#)
- xpose.print (db.names), [93](#)
- xpose.read, [10](#)
- xpose.read (import.graph.par), [124](#)
- xpose.stack, [12](#)
- xpose.stack (generic), [121](#)
- xpose.string.print, [12](#), [247](#)
- xpose.VPC, [13](#), [17](#), [136](#), [163](#), [165](#), [217](#), [247](#), [254](#)
- xpose.VPC.both, [13](#), [17](#), [253](#), [256](#)
- xpose.VPC.categorical, [13](#), [17](#), [254](#), [255](#)
- xpose.write, [10](#)
- xpose.write (export.graph.par), [115](#)
- xpose.xscale.components.log10, [12](#)
- xpose.xscale.components.log10 (xpose.yscale.components.log10), [257](#)
- xpose.yscale.components.log10, [12](#), [257](#)
- xpose4, [8](#)
- xpose4 (xpose4-package), [5](#)
- xpose4-package, [5](#)
- xpose4classic-package, [5](#), [7](#)
- xpose4data-package, [5](#), [9](#)
- xpose4generic-package, [5](#), [11](#)
- xpose4psn-package, [5](#), [12](#)
- xpose4specific-package, [5](#), [7](#), [11](#), [13](#), [158](#)
- xpPage, [17](#)
- xpPage (gof), [122](#)
- xscale.components, [257](#), [258](#)
- xscale.components.default, [233](#)
- xsubset, [10](#), [41](#), [50](#), [90](#), [128](#), [131](#), [134](#), [167](#), [172](#), [230](#), [234](#), [238](#), [240](#), [243](#), [246](#), [249](#), [258](#)
- xsubset<- (xsubset), [258](#)
- xvardef, [10](#), [33](#), [67](#), [118](#), [184](#), [245](#), [246](#), [259](#)
- xvardef<- (xvardef), [259](#)
- xypplot, [19](#), [21](#), [23](#), [25](#), [26](#), [28](#), [30](#), [33](#), [35](#), [36](#),

42, 43, 45, 47, 48, 50, 55, 82, 83, 86,
89, 103–105, 107, 108, 110, 111,
113, 114, 129, 137, 141, 146–148,
150, 152, 155, 158, 178, 179, 182,
207, 214, 215, 226, 227, 233–235,
238, 240, 243, 256

`yscale.components.default`, 233