Package 'ERSA'

September 23, 2020

Type Package

Title Exploratory Regression 'Shiny' App

Version 0.1.3

Date 2020-9-22

Author Catherine B. Hurley

Maintainer Catherine B. Hurley <catherine.hurley@mu.ie>

Description Constructs a 'shiny' app function with interactive displays for summary and analysis of variance regression tables, and parallel coordinate plots of data and residuals.

License GPL (>= 2.0)

Encoding UTF-8

LazyData true

Imports shiny, miniUI,RColorBrewer, ggplot2, car, leaps, broom, dplyr, tidyr, purrr, combinat,stats, methods

RoxygenNote 7.1.1

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2020-09-22 23:00:02 UTC

R topics documented:

---

add1_models   *Constructs a list of fits by adding predictors sequentially*

Description

Constructs a list of fits by adding predictors sequentially

Usage

```
add1_models(model, preds, data = NULL)
```

Arguments

model   A linear model

preds   Predictors to be added sequentially

data   The dataset (optional)

Value

A list of linear fits

---

createERServer   *A function which returns a shiny server for Exploratory Regression*

Description

A function which returns a shiny server for Exploratory Regression

Usage

```
createERServer(
  ERfit,
  ERdata = NULL,
  ERbarcols = RColorBrewer::brewer.pal(4, "Set2"),
  ERnpcpCols = 4,
  pvalOrder = F
)
```

Arguments

`ERfit`	the lm fit to be explored
`ERdata`	the data used to fit the model. If NULL, attempts to extract from ERfit.
`ERbarcols`	a vector of colours, one per term in lm. Will be expanded via colorRampPalette if not the correct length.
`ERnpcpCols`	number of colours for the PCP
`pvalOrder`	if TRUE, re-arranges predictors in order of p-value

Value

a function

`createERUI`	*Constructs UI for Exploratory Regression app*

Description

Constructs UI for Exploratory Regression app

Usage

```
createERUI(tablesOnly = F, gadget = TRUE)
```

Arguments

`tablesOnly`	if TRUE, shows Plots 1-3 only.
`gadget`	If TRUE, constructs a gadget, otherwise a shinyApp

Value

the UI

`drop1_models`	*Constructs a list of fits by dropping predictors from the supplied model*

Description

Constructs a list of fits by dropping predictors from the supplied model

Usage

```
drop1_models(model, preds, data = NULL)
```

Arguments

`model`	A linear model
`preds`	Predictors to be dropped
`data`	The dataset (optional)

Value

A list of linear fits

ERSA   *ERSA: A package exploring regressions with a Shiny app*

Description

The Exploratory Regression Shiny App (ERSA) package consists of a collection of functions for displaying the results of a regression calculation, which are then packaged together as a shiny app function.

---

exploreReg   *A function to launch the Exploratory Regression Shiny App*

Description

A function to launch the Exploratory Regression Shiny App

Usage

```
exploreReg(
  ERmfull,
  ERdata = NULL,
  ERbarcols = RColorBrewer::brewer.pal(4, "Set2"),
  npcpCols = 4,
  pvalOrder = F,
  tablesOnly = F,
  displayHeight = NULL,
  gadget = TRUE,
  viewer = "dialogViewer"
)
```

Arguments

`ERmfull`	the lm fit to be explored
`ERdata`	the data used to fit the model. If NULL, attempts to extract from ERmfull.
`ERbarcols`	a vector of colours, one per term in lm. Will be expanded via colorRampPalette if not the correct length.
`npcpCols`	number of colours for the PCP
`pvalOrder`	if TRUE, re-arranges predictors in order of p-value
`tablesOnly`	if TRUE, shows Plots 1-3 only.
`displayHeight`	supply a value for the display height
`gadget`	If TRUE, constructs a gadget, otherwise a shinyApp.
`viewer`	For gadget, defaults to "dialogViewer". May be "paneViewer" or "browserViewer"

Value

the result

Examples

```
f <- lm(mpg ~ hp+wt+disp, data=mtcars)
## Not run: exploreReg(f)
```

---

pcpPlot	*A PCP plot of the data, residuals or hat values from regression fits*

Description

A PCP plot of the data, residuals or hat values from regression fits

Usage

```
pcpPlot(
  data,
  fit,
  type = "Variables",
  npcpCols = 4,
  resDiff = F,
  absResid = F,
  sequential = F,
  selnum = NULL
)
```

Arguments

`data`	a data frame
`fit`	a lm for the data frame
`type`	one of "Variables", "Residuals", "Hatvalues"
`npcpCols`	number of colours
`resDiff`	difference residuals, TRUE or FALSE
`absResid`	absolute residuals, TRUE or FALSE
`sequential`	use sequential fits (TRUE) or drop1 fits (FALSE)
`selnum`	row numbers of cases to be highlighted

Value

ggplot

Examples

```
f <- lm(mpg ~ wt+hp+disp, data=mtcars)
pcpPlot(mtcars, f, type="Residuals")
```

---

plotSeqSS	*Plots barcharts of sequential sums of squares of lm*

Description

Plots barcharts of sequential sums of squares of lm

Usage

```
plotSeqSS(fits, barcols = NULL, legend = F)
```

Arguments

`fits`	list of lm objects
`barcols`	a vector of colours, one per term in lms
`legend`	TRUE or FALSE

Value

a ggplot

Examples

```
plotSeqSS(list(fit1= lm(mpg ~ wt+hp+disp, data=mtcars),
fit2=lm(mpg ~ wt*hp*disp, data=mtcars)))
```

plotSum	*Plots of model summaries*

Description

Plots of model summaries

Usage

```
plotAnovaStats(
  fit0,
  barcols = NULL,
  preds = NULL,
  alpha = 0.05,
  type = "SS",
  width = 0.3
)

plottStats(fit0, barcols = NULL, preds = NULL, alpha = 0.05, width = 0.3)

plotCIStats(
  fit0,
  barcols = NULL,
  preds = NULL,
  alpha = 0.05,
  stdunits = FALSE,
  width = 0.3
)
```

Arguments

fit0	is an lm object
barcols	a vector of colours, one per term in lm
preds	terms to include in plot
alpha	significance level
type	"SS" or "F", from type 3 Anova
width	bar width
stdunits	TRUE or FALSE. If TRUE, coefficients refer to standardised predictor units.

Value

a ggplot

Functions

- `plotAnovaStats`: Plots barchart of F or SS from lm

- `plottStats`: Plots barchart of t stats from lm

- `plotCIStats`: Plots confidence intervals from lm

Examples

```
plotAnovaStats(lm(mpg ~ wt+hp+disp, data=mtcars))
plottStats(lm(mpg ~ wt+hp+disp, data=mtcars))
plotCIStats(lm(mpg ~ wt+hp+disp, data=mtcars))
```

---

reorderTerms   *Re-order model terms*

Description

Re-order model terms

Usage

```
pvalOrder(m, d = NULL, refit = TRUE)

bselOrder(m, d = NULL, refit = TRUE, maxNPred = NULL)

fselOrder(m, d = NULL, refit = TRUE, maxNPred = NULL)

revPredOrder(m, d = NULL, refit = TRUE)

randomPredOrder(m, d = NULL, refit = TRUE)

regsubsetsOrder(m, d = NULL, refit = TRUE, collapse = TRUE)
```

Arguments

m	an lm objecct
d	the data frame. If NULL, attempts to extract from m.
refit	TRUE or FALSE
maxNPred	maximum number of predictors to use, defaults to all.
collapse	TRUE or FALSE

Value

a vector of terms in order last to first, or an lm if refit=TRUE. regsubsetsOrder returns a list of predictor vectors, or a list of fits

Functions

- `pvalOrder`: Arranges model terms in order of increasing p-value

- `bselOrder`: Arranges model terms using backwards selection

- `fselOrder`: Forwards selection

- `revPredOrder`: Reverses order of terms in a fit

- `randomPredOrder`: Reorders terms in a fit randomly

- `regsubsetsOrder`: Best subsets regression.

Examples

```
bselOrder(lm(mpg~wt+hp+disp, data=mtcars))
fselOrder(lm(mpg~wt+hp+disp, data=mtcars))
revPredOrder(lm(mpg~wt+hp+disp, data=mtcars))
randomPredOrder(lm(mpg~wt+hp+disp, data=mtcars))
regsubsetsOrder(lm(mpg~wt+hp+disp, data=mtcars))
```

---

termColours	*Constructs colour vector for model terms*

Description

Constructs colour vector for model terms

Usage

```
termColours(f, pal = RColorBrewer::brewer.pal(4, "Set2"))
```

Arguments

f	a model fit with term labels
pal	use this palette

Value

a vector of colours. Residuals are given a grey color

Examples

```
termColours(lm(mpg ~ wt+hp, data=mtcars))
```

Index