Package 'RGtk2DfEdit'

February 17, 2010

Type Package

Title Improved data frame editor for RGtk2

Version 0.5.3.1

Date 2010-02-17

Author Tom Taverner, with contributions from John Verzani

Maintainer Tom Taverner <thomas.taverner@pnl.gov>

Description RGtk2 data frame editor megawidget

Depends methods, RGtk2(>= 2.12.8), cairoDevice

License GPL (>= 2)

LazyLoad yes

R topics documented:

---

dfedit *Convenience function for editing a data frame in its own window*

Description

Optional name to assign the dataset when updating

Usage

```
dfedit(items, dataset.name = deparse(substitute(items)),
size=c(500, 300))
```

Arguments

items	The data frame to edit
dataset.name	Optional name to assign the dataset when updating
size	Optional window size request

Details

IF set to NULL, is the name of the dataset.

Value

Returns the [gtkDfEdit](#) object

Examples

```
obj <- dfedit(iris)
```

---

gtkDfEdit *gtkDfEdit*

Description

An RGtk2 spreadsheet package for editing data frames. Improves on base edit.data.frame function found in utils

Usage

```
gtkDfEdit(items, dataset.name = deparse(substitute(items)),
size.request=c(500, 300))
```

Arguments

items	The data frame to edit
dataset.name	The name of the data frame object to modify.
size.request	The size request for the window.

Details

gtkDfEdit is an RGtk2 based data frame viewer and editor megawidget intended to be familiar to spreadsheet users and to form part of larger GUI projects. It provides a way to edit a data frame (see Notes for a description).

Changes made in the spreadsheet will appear instantly in the data frame.

The "[" method is used for data-frame like extraction from the object.

The $getSelection method returns a list of selected row and column indices.

The $getModel method returns the backing `RGtkDataFrame`.

The $getDimension method returns the backing data frame dimension.

The $getColumnNames method returns the column names.

The $getRowNames method returns the row names.

The $setColumnName(idx, new.name) method sets the column name at a particular index.

The $setColumnClickHandler method sets a function to handle clicking on a column.

Value

A GtkContainer containing the megawidget.

Note

The editor consists of row names, column names, the main grid of cells, and the left-hand corner cell. You can move around within the grid using the keyboard, the scrollbars, or by clicking and dragging with the mouse.

Navigation Around The Grid

Keyboard navigation uses the familiar arrow or Shift, Shift-Enter, Tab, Shift-Tab, PgUp, PgDown, Ctrl-PgUp, Ctrl-PgDown, Home, End keys. These work when either the grid or the column of row names has the focus.

Pressing a non-navigation key when the row names have focus will cause automatic navigation to the closest match for the row name. The name matching entry dialog will go away after a couple of seconds.

Mouse navigation to a grid location can be done via the scroll bars on the grid.

Editing The Grid

Using non-navigation keys in a selected cell will start editing within the cell. If the column is of factor type, the cell entry will provide the user with an autocompletion containing existing factor levels.

Focusing out of the cell or pressing any navigation key will end the edit. Edited cell entries will be coerced to the column's data type, so alphabetical strings put into numeric columns will turn into a platform-dependent variant of "NA". All character strings are stripped of beginning and end whitespace. Adding a new item to a factor column will automatically update factor levels.

Deleting cells sets their contents to either NA, or "" if the cells are of type character. Deleting cell contents can be done in a number of ways. Deleting a block of selected cells on the grid can be done by selecting them, then pressing the Delete key. The Backspace key deletes the cell the cursor is on ignoring all selections.

Deleting entire rows and columns can be done by selecting the row names or column headers then pressing Delete, or else by right-clicking on the row names or column headers to bring up the context menu then clicking "Clear Contents".

Changes made in the data frame editor are automatically and invisibly updated so the R data frame object is kept synchronous with the grid display.

Ctrl-Z undoes the previous edit to cells, rows or columns, with certain limitations. It will not restore changes to the numbers of rows or columns, or undo data values altered by coercion (see below).

Editing Row And Column Names

Double clicking row names and column names allows the user to edit them. Typing in the replacement name and pressing Enter, Escape or clicking somewhere else will set the changed row or column name.

Duplicate row names will be turned into unique values by replacing each duplicate with the lowest possible ordinal number.

These operations can be undone via Ctrl-Z.

Editing The Data Frame Object Name

The name of the data frame object is displayed in the top-left corner cell.

Double clicking the top-left corner cell allows the data set to be updated and reassigned when Return is pressed. When editing is finished the data frame in the editor will be written to the new dataset name.

Cell Selection

Active cells or cell selections are indicated with a focus rectangle. Active columns are indicated by a colored highlight. By clicking and dragging with the mouse, you can scroll around the grid in two dimensions and select a rectangular block of cells. Alternatively, you can use the keyboard arrow keys with Shift held down to select a block.

Left-clicking and dragging on a region of cells selects the region and draws a focus rectangle around it. Selections are indicated by highlighted rows, column headers and a drawn focus rectangle. Rows can be selected by focusing on the floating row column and then doing either mouse and keyboard selection.

The keyboard can also be used for grid selection. Left clicking on column headers or row names selects the columns or rows. Multiple, or ranges of, columns or rows can be selected using the usual Ctrl-Click and Shift-Click combinations.

Ctrl-A, or clicking the top-left corner cell, selects all cells on the grid.

Copying And Pasting

Copied and pasted data is in tab-delimited form and can be pasted directly into other spreadsheets or text editors. We use the usual platform specific line separator.

In Linux, the functions `xclip` and `xsel` must be available at the command line for copy and paste to work. In Mac, `pbcopy` and `pbpaste` are used. In Windows, we use the R functions `writeClipboard` and `readLines`.

Ctrl-V pastes cell selections to the clipboard at the selected point into a block defined by the size of the pasted matrix and starting at the corner most selected cell. At this point, this operation will add rows, but not columns, to the grid. Pasting automatically coerces data to the type in the column.

Ctrl-C entered while focus is on the grid copies the selected block of cells.

Alternatively these functions can be accessed from the grid right click context menu "Copy" and "Paste".

Copying a cell block into the clipboard will not include row or column names. To include row and column names in the copy operation, select "Copy With Names" from the grid right click context menu.

Copying and pasting rows and columns can be done through the right click context menus over row headers or column headers in the "Copy" and "Paste" commands. Copying from a column will

include the column header and copying from a row will include the row header. Pasting on columns will update the column headers.

These operations can be undone via Ctrl-Z.

Data Coercion And Special Functions

From the right click context menu on column headers the selected data frame columns' assigned type can be changed. Available data types are Numeric, Integer, Logical, Character, Factor. Factor is a special enumerated data type (also known as a category) which can have its attributes set using the in-built Factor Editor (see below). To coerce a data column, just open this menu and click the desired type.

The column context menu function "Set As Row Names" sets the contents of the column as the data frame's row names. The menu function "Shorten Names..." replaces long string names with their unique abbreviations.

Right clicking the top-left corner cell selects all cells and brings up a menu allowing global cut, copy, and paste actions. "Row Names To Column" inserts the row names into the first column of the data set and replaces the row names by their ordinality. "Edit Dataset Name" allows the data set name in the R environment to be reassigned. "Default Row Names" sets the row names to their ordinal numbers from 1 to the number of rows.

Coercion can be undone but the side effects are not necessarily undoable. For example, coercing a column of integers to a logical will irreversibly set the integer values to 1 and 0 after undoing the operation.

Inserting And Deleting Columns And Rows

Right clicking on row name headers brings up a menu which allows Insert and Delete actions on data columns. "Insert" inserts a blank row before the row clicked. "Insert After" inserts a blank row after the point. "Delete" deletes the selected row range and is not available when rows are not selected.

Right clicking on column name headers similarly brings up a menu which allows Insert and Delete actions on data columns. "Insert" inserts a blank column before the column clicked. To insert a blank column at the end, click the blank header at the right hand side. "Delete" deletes the selected column(s).

These operations cannot be undone.

Editing Factors

Right clicking on a column header of a factor column, then selecting "Factor Editor", or right clicking a selected factor column, opens the Factor Editor which allows factor levels, order and contrasts to be set.

The Factor Editor window displays the choice of data frame factor columns, the factor levels of the selected columns, and the contrasts in the "Factor Contrasts" expander. When a column is selected, if it is a factor, its levels are displayed in the "Factor Level Order" frame. The factor levels can be re-ordered, edited, deleted or additional levels added by using the buttons to the right of the level display.

Factors are associated with contrast matrices for use in analysis of variance and regression models. The Factor Editor allows contrasts to be set by opening the "Factor Contrasts" expander frame and selecting the desired contrast type. The default contrast type sets the first ordered level as the control.

It is often desirable to fill in factor levels according to a pattern, for example, in specifying a balanced experimental design. This can be done in two ways. First, highlighting a region of cells then right clicking on a Factor column, pulls up the context menu including three options, "Fill Selected Down" "Randomize Selected", "Fill In Blocks".

"Fill Selected Down" fills all selected cells in the column with the FIRST selected cell.

"Randomize Selected" replaces all selected cells within the column that was clicked with the same contents, in randomized order.

"Fill In Blocks" opens a new window containing a spin button specifying the block size of factor level repeats to fill the selected region. For example, factor levels A, B, C, block size 2, the region is filled down A, A, B, B, C, C, A, A, B, B, C, C, etc. The region will be filled when the spin button is modified or Enter is pressed, and the fill can be cancelled by pressing Cancel. The OK button will cause the changes to be fixed.

The same factor filling options as described above can be accessed directly from the Factor Editor window, which can be called up as described above using "Selected", "Random Fill" and "Fill with Replicates...". In this case, it fills the entire column, not just the highlighted region.

Sorting Data

From the right-click menu on the corner left hand cell or on the columns, the "Sort..." dialog can be opened. This dialog consists of (1) a "Sort Key" Selection frame (2) "Add/Remove Key" frame to add/remove sort keys (3) "OK" and "Cancel" buttons.

Sort operations on the data cannot currently be undone and they will rearrange the underlying R object and cause the undo stack to be cleared.

The "Sort Key" frame contains key choice items consisting of a combo box for key selection, radio buttons for coercion of the key, and radio buttons for choosing the sort direction. Sorting starts with the first key, breaking ties by keys further down the list.

The combo box allows the user to choose the column of the data frame, including the row names, they wish to sort on.

The coercion radio buttons allow the user to sort on the corresponding column by the default `xtfrm` ranking, or by first coercing to character or numerical form. This can be useful for sorting numeric row names or factors.

The "Ascending" and "Descending" radio buttons choose whether the sort on the corresponding key item is in ascending or descending order.

The "Add/Remove Keys" frame contains a button "Add A Key" allowing the user to add another key choice item to the "Sort Key" frame and a button "Remove A Key" to remove the last key choice item in the frame. There is no limit to the number of keys that can be sorted.

Finally the "OK" button initiates the data frame sort and the Cancel button closes the dialog.

Author(s)

Tom Taverner <Thomas.Taverner@pnl.gov>, with contributions from John Verzani

See Also

dfedit

Examples

```
win <- gtkWindowNew()
obj <- gtkDfEdit(iris)
win$add(obj)

obj[1,1,drop=FALSE]
obj$getSelection()
obj$setColumnName(1, "hi there")
obj$setColumnClickHandler(function(obj, col) print(obj[,col]))
```

`gtkDfEditGetColumnNames`
Return the columns of the RGtk2DfEdit object...

Description

Return the columns of the RGtk2DfEdit object

Usage

`gtkDfEditGetColumnNames(object)`

Arguments

`object`   The RGtk2DfEdit object

Value

Returns the column names for the current object

`gtkDfEditGetDataFrame`
Return a data frame from the RGtk2DfEdit object...

Description

Return a data frame from the RGtk2DfEdit object

Usage

`gtkDfEditGetDataFrame(object)`

Arguments

`object`   The RGtk2DfEdit object

Value

Returns the data frame with row names and column names

```
gtkDfEditGetDimension
```
                              Return the dimensions (nrow, ncol) of the RGtk2DfEdit object...

Description

Return the dimensions (nrow, ncol) of the RGtk2DfEdit object

Usage

```
gtkDfEditGetDimension(object)
```

Arguments

object   The RGtk2DfEdit object

Value

Returns the number of rows and columns – not counting row names

---

```
gtkDfEditGetModel
```
  get Model from object...

Description

get Model from object

Usage

```
gtkDfEditGetModel(object)
```

Arguments

object   The RGtk2DfEdit object

Value

the RGtk2DataFrame that is the backend model for the widget

`gtkDfEditGetRowNames`
Return the row names of the RGtk2DfEdit object...

Description

Return the row names of the RGtk2DfEdit object

Usage

`gtkDfEditGetRowNames(object)`

Arguments

`object`   The RGtk2DfEdit object

Value

Returns the row names for the current object

`gtkDfEditGetSelection`
get selected row and column indices...

Description

get selected row and column indices

Usage

`gtkDfEditGetSelection(object)`

Arguments

`object`   The RGtk2DfEdit object

Value

the 1-indexed selected rows

---

gtkDfEditSetColumnClickHandler
Function to call when column is clicked...

Description

Function to call when column is clicked

Usage

```
gtkDfEditSetColumnClickHandler(object, columnClickHandler)
```

Arguments

object   The RGtk2DfEdit object

columnClickHandler

        Function to call when column clicked. Signature is (dataframe, column number). If NULL (default) no handler is called.

Details

IF set to NULL, no handler is called.

---

gtkDfEditSetColumnName
Set the columns of the RGtk2DfEdit object...

Description

Set the columns of the RGtk2DfEdit object

Usage

```
gtkDfEditSetColumnName(object, idx, new.name)
```

Arguments

object   The RGtk2DfEdit object

idx   The col index

new.name   The new column name

Value

Returns the column names for the current object

`gtkDfEditSetRowClickHandler`
Function to call when row is clicked...

Description

Function to call when row is clicked

Usage

`gtkDfEditSetRowClickHandler(object, rowClickHandler)`

Arguments

`object`   The RGtk2DfEdit object

`rowClickHandler`

Function to call when row clicked. Signature is (dataframe, row number). If NULL (default) no handler is called.

Details

IF set to NULL, no handler is called.

`[.GtkDfEdit`   *S3 data extraction method...*

Description

S3 data extraction method

Usage

`\method{[}{GtkDfEdit} (x, i, j, drop=TRUE)`

Arguments

`x`	The RGtk2DfEdit object
`i`	Row index
`j`	Column indext
`drop`	passed to extraction for data frame

Details

Grabs data frame then passes onto [.data.frame method

Value

The extracted entries

Index