

Package ‘RcmdrPlugin.UCA’

October 22, 2021

Type Package

Title UCA Rcmdr Plug-in

Version 4.5-1

Date 2021-10-22

Author Manuel Munoz-Marquez <manuel.munoz@uca.es>

Maintainer Manuel Munoz-Marquez <manuel.munoz@uca.es>

Depends IQCC, qcc, qicharts2, randtests, rmarkdown, TeachingDemos,
tseries

Suggests car, carData, knitr

VignetteBuilder knitr

Imports Rcmdr (>= 1.6)

Description Some extensions to Rcmdr (R Commander), randomness test,
variance test for one normal sample and predictions using active model,
made by R-UCA project and used in teaching statistics at University of Cadiz (UCA).

License GPL (>= 3)

URL <http://knuth.uca.es/RcmdrPlugin.UCA/>

Repository CRAN

Repository/R-Forge/Project rcmdrpluginuca

Repository/R-Forge/Revision 36

Repository/R-Forge/DateTimeStamp 2021-10-22 07:57:10

Date/Publication 2021-10-22 08:50:12 UTC

NeedsCompilation no

R topics documented:

RcmdrPlugin.UCA-package	2
paretochartMenu	3
paretochartMenu.es	4
predictMenu	4

predictMenu.es	6
qccMenu	8
qccMenu.es	8
randomnessMenu	9
randomnessMenu.es	10
RcmdrPlugin.UCA.es	11
sigmaMenu	12
sigmaMenu.es	12

Index	14
--------------	-----------

RcmdrPlugin.UCA-package

RcmdrPlugin.UCA: Some customization to Rcmdr made by R-UCA project

Description

Some extension to Rcmdr (R Commander), randomness test, variance test for one normal sample and predictions using active model, to teach statistics in a first university course made by R-UCA project and used at University of Cadiz (UCA).

All the customizations provides by this package will be propose to be part of the Rcmdr package. All the customization included in the Rcmdr package will be dropped from here.

Details

Package: RcmdrPlugin.UCA
 Type: Package
 Version: 4.4-1
 Date: 2021-11-22
 License: GPL version 3 or newer

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

References

[1] <http://knuth.uca.es/RcmdrPlugin.UCA/>

See Also

For more information see [Rcmdr-package](#).

Para ayuda en español, véase [RcmdrPlugin.UCA.es](#). (For Spanish help see [RcmdrPlugin.UCA.es](#).)

Examples

Examples of use are provided in the help of each option of menu.

paretochartMenu	<i>Menu option to make a Pareto chart</i>
-----------------	---

Description

Within the menu "Quality Control" -> "Pareto Chart", a new entry is provided to make a Pareto chart in which a factor type variable is represented in decreasing order of frequencies. This graph is often used to identify the most common causes of a failure. This option uses the `paretochart` function from the `qicharts2` package. For more information see [paretochart](#).

Details

Here is an example of "Pareto chart" menu entry.

Load data "hospital_infections" selecting from Rcmdr menu: "Data" -> "Data in packages" -> "Read data set from an attached package..." then double-click on "quicharts2", click on "hospital_infections" and on "OK". Rcmdr reply with the following command in source pane (R Script)

```
data(hospital_infections,package="qicharts2")
```

To build a Pareto chart on variable "infection", select from Rcmdr menu: "Quality Control" -> "Pareto chart" select "infection" and "OK". Rcmdr reply with the following command in source pane (R Script)

```
with(hospital_infections,paretochart(infection))
```

and the graph is plotted in another window.

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

For more information see [Rcmdr-package](#).

Para ayuda en español, véase [Gráfico de Pareto \(es\)](#). (For Spanish help see [Gráfico de Pareto \(es\)](#).)

paretochartMenu.es *Opción de menú para hacer gráficos de Pareto*

Description

Dentro del menú "Control de Calidad" -> "Gráfica de Pareto", se proporcionan una nueva entrada para realizar un gráfico de Pareto en el que se representa una variable de tipo factor por orden decreciente de frecuencias. Esta gráfica se suele utilizar para identificar las causas más frecuentes de un fallo. Esta opción usa la función `paretochart` del paquete `qicharts2`. Para más información véase [paretochart](#).

Details

Ejemplo de uso del menú "Gráfica de Pareto".

Cargar los datos "hospital_infections" eligiendo del menu de Rcmdr: "Datos" -> "Conjuntos de datos en paquetes" -> "Leer conjunto de datos desde paquete adjunto..." pulsar dos veces sobre "qicharts2", pulsar sobre "hospital_infections" y sobre "Aceptar". Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
data(hospital_infections,package="qicharts2")
```

Para construir el diagrama de Pareto de la variable "infection", seleccione del menú de Rcmdr: "Control de Calidad" -> "Gráfica de Pareto" seleccione "infection" y "Aceptar". Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
with(hospital_infections,paretochart(infection))
```

y se obtiene el gráfico de Pareto en una nueva ventana.

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

Para más información véase [Rcmdr-package](#).

predictMenu *Menu option to make predictions using active model*

Description

The menu entry "Predict using active model", in models menu, has two options to predict data using active model depending on how the data for predictor variables will be provided.

The first entry is "Input data and predict". If you select this option a new data set, as a data.frame, will be created and the editor will be invoked. Then you can enter the values of the predictor variable that you want to use for prediction, the values for non predictor variables are not required. When you close the data editor the predicted values for predicted variable are shown.

The second menu entry "Add predictions to existing dataset..." allows to us to add predictions to existing dataset that provides the data values for all predictor variables. After selecting this option, the user can select an existing data set using dialog box.

If the data set does not provides the values for all predicting variables an error will occur and no predicted values will be provided.

Unlike the menu option "Add observation statistics to data...", this option can be used with a different data set than the one used to construct the model, if that dataset provides the values for all the predictor variables.

Details

Here is an example of use of "Input data and predict" menu entry.

Load data "Chile" selecting from Rcmdr menu: "Data" -> "Data in packages" -> "Read data set from an attached package..." then double-click on "car", click on "Chile" and on "OK".

Rcmdr reply with the following command in source pane (R Script)

```
data(Chile, package="car")
```

To build a model select from Rcmdr menu: "Statistics" -> "Model fit" -> "Linear Regression...". As "Response variable" select income and age as "Explanatory variables" and click on "OK". Rcmdr reply with the following command in source pane (R Script)

```
RegModel.1 <-lm(income~age,data=Chile)
```

```
summary(RegModel.1)
```

Note that the active model is set to RegModel.1. So if you want to predict a new value for a 35 and 40 age person. Select from Rcmdr menu: "Models" -> "Predict using active model" -> "Input data and predict". In age column input 35 and 40 and then close the editor. Rcmdr reply with the following command in source pane (R Script)

```
.data <-edit(Chile[0,])
```

```
.data
```

```
predict(RegModel.1, .data)
```

```
remove(.data)
```

And output the predicted value of income for that age using active model (RegModel.1)

Here is an example of use of "Add predictions to existing dataset..." menu entry.

Load data "Chile" selecting from Rcmdr menu: "Data" -> "Data in packages" -> "Read data set from an attached package..." then double-click on "car", click on "Chile" and "OK".

Rcmdr reply with the following command in source pane (R Script)

```
data(Chile, package="car")
```

To build a model select from Rcmdr menu: "Statistics" -> "Model fit" -> "Linear Regression..." as "Response variable" select income and age as "Explanatory variables" and click on "OK". Rcmdr reply with the following command in source pane (R Script)

```
RegModel.1 <-lm(income~age,data=Chile)
```

```
summary(RegModel.1)
```

Note that the active model is set to RegModel.1 So if you want to predict the values for income for age data in Chile dataset. Select from Rcmdr menu: "Models" -> "Predict using active model" ->

"Add predictions to existing dataset...". In the dialog select a compatible dataset with the model. In this case select Chile. Rcmdr reply with the following command in source pane (R Script)

```
Chile$fitted <-predict(RegModel.1,Chile)
```

The predicted value of income has been saved as fitted in the selected dataset (Chile)

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

For more information see [Rcmdr-package](#).

Para ayuda en español, véase [Menú predicciones \(es\)](#). (For Spanish help see [Menú predicciones \(es\)](#).)

predictMenu.es

Opción de menú para hacer predicciones usando el modelo activo

Description

La entrada de menú "Predecir usando el modelo activo", en el menú modelos, tiene dos opciones para predecir datos usando el modelo activo dependiendo de cómo se proporcionen los valores para las variables predictoras.

La primera entrada es "Introducir datos y predecir". Si selecciona esta opción, se creará un nuevo conjunto de datos, como un data.frame y se abrirá el editor automáticamente. A continuación, puede introducir los valores de la variable predictora que desea utilizar para la predicción, no es necesario introducir los valores de las variables no predictoras. Al cerrar el editor de datos se muestran los valores predichos.

La segunda entrada de menú "Agregar predicciones a un conjunto de datos existente ..." permite agregar predicciones a un conjunto de datos existente que proporcione los datos valores para todas las variables predictoras. Después de seleccionar esta opción, el usuario o usuaria puede seleccionar un conjunto de datos existente utilizando el cuadro de diálogo correspondiente.

Si el conjunto de datos no proporciona los valores para todas las variables predictoras, se producirá un error y no se proporcionarán valores predichos.

A diferencia de la opción del menú "Añadir las estadísticas de las observaciones a los datos...", esta opción puede ser usada con un conjunto de datos distinto al que se ha utilizado para construir el modelo, siempre que dicho conjunto de datos proporcione los valores para todas las variables predictoras.

Details

Ejemplo de uso del menú "Introducir datos y predecir".

Cargar los datos "Chile" eligiendo del menú de Rcmdr: "Datos" -> "Conjuntos de datos en paquetes" -> "Leer conjunto de datos desde paquete adjunto..." pulsar dos veces sobre "car", pulsar

sobre "Chile" y sobre "Aceptar". Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
data(Chile, package="car")
```

Para construir un modelo seleccione del menú de Rcmdr: "Estadísticos" -> "Ajuste de modelos" -> "Regresión lineal...". Como "Variable explicada" seleccione income y age como "Variables explicativas" y pulse "Aceptar". Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
RegModel.1 <-lm(income~age, data=Chile)
```

```
summary(RegModel.1)
```

Obsérvese que el modelo se ha establecido a RegModel.1. Si quiere predecir el ingreso (income) para personas de 35 y 40 años de edad (age). Del menú de Rcmdr seleccione: "Modelos" -> "Predecir usando el modelo activo" -> "Introducir datos y predecir". En la columna edad (age) introduzca 35 y 40 y cierre el editor. Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
.data <-edit(Chile[0,])
```

```
.data
```

```
predict(RegModel.1, .data)
```

```
remove(.data)
```

Se muestra el valor predicho de ingresos para esas edades usando el modelo activo (RegModel.1)

Ejemplo de uso del menú "Añadir predicciones a un conjunto de datos existente..."

Cargar los datos "Chile" eligiendo del menú de Rcmdr: "Datos" -> "Conjuntos de datos en paquetes" -> "Leer conjunto de datos desde paquete adjunto..." pulsar dos veces sobre "car", pulsar sobre "Chile" y "Aceptar". Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
data(Chile, package="car")
```

Para construir un modelo seleccione del menú de Rcmdr: "Estadísticos" -> "Ajuste de modelos" -> "Regresión lineal..." como "Variable explicada" seleccione income y age como "Variables explicativas" y pulse "Aceptar".

Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
RegModel.1 <-lm(income~age, data=Chile) summary(RegModel.1)
```

Obsérvese que el modelo se ha establecido a RegModel.1. Si se quiere predecir los valores para los ingresos (income) para los valores de edad (age) en el conjunto de datos Chile del menú "Modelos" de Rcmdr seleccione: "Predecir usando el modelo activo" -> "Añadir predicciones a un conjunto de datos existente...". En el diálogo seleccione un conjunto de datos compatible con el modelo. En este caso seleccione Chile. Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
Chile$fitted <-predict(RegModel.1, Chile)
```

Las predicciones para ingreso (income) se han guardado como fitted en el conjunto de datos (Chile)

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

Para más información véase [Rcmdr-package](#).

qccMenu

Menu options to make a quality control graphs

Description

Within the menu "Quality Control" new entries are provided to make mutiple quality control graphs.

Details

These options use the qcc function from the qcc package. For more information see [qcc](#).

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

For more information see [qcc](#) and [Rcmdr-package](#).

Para ayuda en español, véase [Gráficos de control de calidad \(es\)](#). (For Spanish help see [Gráficos de control de calidad \(es\)](#).)

qccMenu.es

Opcioned de menú para hacer gráficos de control de calidad

Description

Dentro del menú "Control de Calidad" se proporcionan nuevas entradas para realizar múltiples gráficos de control de calidad.

Details

Estas opciones usan la función qcc del paquete qcc. Para más información véase [qcc](#).

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

Para más información véase [qcc](#) y [Rcmdr-package](#).

Description

In the "Nonparametric Tests" menu, two new entries are provided to perform the randomness test.

The first "Randomness test for two level factor..." can be used to contrast the randomness of a factor with two levels. This option use the function `runs.test` from `tseries` package. For more information see [runs.test](#).

The second entry in the menu "Randomness test for numeric variable..." is used to test the randomness of a numerical variable. This option use the function `runs.test` from `randtest` package. For more information see [runs.test](#).

Details

Here is an example of "Randomness test for a two level factor..." menu entry.

Load data "AMSSurvey" selecting from Rcmdr menu: "Data" -> "Data in packages" -> "Read data set from an attached package..." then double-click on "car", click on "AMSSurvey" and on "OK". Rcmdr reply with the following command in source pane (R Script)

```
data(AMSSurvey, package="car")
```

To make randomness test on variable "sex", select from Rcmdr menu: "Statistics" -> "Nonparametric tests" -> "Randomness test for two level factor..." select "sex" and "OK". Rcmdr reply with the following command in source pane (R Script)

```
with(AMSSurvey, twolevelfactor.runs.test(sex))
```

Here is an example of "Randomness test for a numeric variable..." menu entry.

Load data "sweetpotato" selecting from Rcmdr menu: "Data" -> "Data in packages" -> "Read data set from an attached package..." then double-click on "randtests", click on "sweetpotato" and on "OK". Rcmdr reply with the following command in source pane (R Script)

```
data(sweetpotato, package="randtests")
```

```
sweetpotato <- as.data.frame(sweetpotato)
```

To make randomness test on variable "yield", select from Rcmdr menu: "Statistics" -> "Nonparametric tests" -> "Randomness test for numeric variable..." select "yield" and "OK". Rcmdr reply with the following command in source pane (R Script)

```
with(sweetpotato, numeric.runs.test(yield))
```

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

For more information see [Rcmdr-package](#).

Para ayuda en español, véase [Test de aleatoriedad \(es\)](#). (For help in Spanish see [Test de aleatoriedad \(es\)](#).)

Description

Dentro del menú "Test no paramétricos" se proporcionan dos nuevas entradas para realizar el test de aleatoriedad.

La primera "Test de aleatoriedad para un factor de dos niveles..." sirve para contrastar la aleatoriedad de una variable de tipo factor con dos niveles. Esta opción usa la función `runs.test` del paquete `tseries`. Para más información véase [runs.test](#).

La segunda entrada del menú "Test de aleatoriedad para una variable numérica..." sirve para contrastar la aleatoriedad de una variable numérica. Esta opción usa la función `runs.test` del paquete `randtest`. Para más información véase [runs.test](#).

Details

Ejemplo de uso del menú "Test de aleatoriedad para un factor con dos niveles..."

Cargar los datos "AMSSurvey" eligiendo del menu de Rcmdr: "Datos" -> "Conjuntos de datos en paquetes" -> "Leer conjunto de datos desde paquete adjunto..." pulsar dos veces sobre "car", pulsar sobre "AMSSurvey" y sobre "Aceptar". Rcmdr responde con la siguiente instruccion en el cuadro de instrucciones (R Script)

```
data(AMSSurvey, package="car")
```

Para hacer el test de aleatoriedad a la variable "sex", seleccione del menú de Rcmdr: "Estadísticos" -> "Test no paramétricos" -> "Test de aleatoriedad para un factor con dos niveles..." seleccione "sex" y "OK". Rcmdr responde con la siguiente instruccion en el cuadro de instrucciones (R Script)

```
with(AMSSurvey, twolevelfactor.runs.test(sex))
```

Ejemplo de uso del menú "Test de aleatoriedad para una variable numérica..."

Cargar los datos "sweetpotato" eligiendo del menu de Rcmdr: "Datos" -> "Conjuntos de datos en paquetes" -> "Leer conjunto de datos desde paquete adjunto..." pulsar dos veces sobre "randtests", pulsar sobre "sweetpotato" y sobre "Aceptar". Rcmdr responde con la siguiente instruccion en el cuadro de instrucciones (R Script)

```
data(sweetpotato, package="randtests")
```

```
sweetpotato <- as.data.frame(sweetpotato)
```

Para hacer el test de aleatoriedad a la variable "yield", seleccione del menú de Rcmdr: "Estadísticos" -> "Test no paramétricos" -> "Test de aleatoriedad para un factor con dos niveles..." seleccione "sex" y "OK". Rcmdr responde con la siguiente instruccion en el cuadro de instrucciones (R Script)

```
with(sweetpotato, numeric.runs.test(yield))
```

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

Para más información véase [Rcmdr-package](#).

For help in English see [Randomness test](#). (Para ayuda en inglés, véase [Randomness test](#).)

RcmdrPlugin.UCA.es *RcmdrPlugin.UCA.es: Algunas adaptaciones para Rcmdr hechas por el proyecto R-UCA*

Description

Algunas extensiones a Rcmdr (R Commander), test de aleatoriedad, test de varianza para una muestra normal y predicciones utilizando el modelo activo, para la enseñanza de un primer curso universitario en estadística hechas por el proyecto R-UCA y usadas en la Universidad de Cádiz (UCA).

Todas las adaptaciones serán propuestas para su incorporación a Rcmdr. Las adaptaciones que se incorporen al paquete Rcmdr se suprimirán de este paquete.

Details

Paquete: RcmdrPlugin.UCA
Tipo: Paquete
Version: 4.5-1
Fecha: 2021-11-22
Licencia: GPL (>= 3)

Author(s)

Manuel Muñoz-Márquez <manuel.munoz@uca.es>

References

[1] <http://knuth.uca.es/RcmdrPlugin.UCA/>

See Also

Para más información véase [Rcmdr-package](#).

Examples

```
# Se proporcionan ejemplos de uso en la ayuda de cada opcion de menú.
```

sigmaMenu

Variance test

Description

Within the "Statistics" -> "Variances" menu, a new entry is provided to calculate confidence intervals and make contrasts on variance in a normal population. This option uses the function `sigma.test` of the package `TeachingDemos`. For more information see [sigma.test](#).

Details

Here is an example of "Single-Sample Variance Test..." menu entry.

Load data "sweetpotato" selecting from Rcmdr menu: "Data" -> "Data in packages" -> "Read data set from an attached package..." then double-click on "randtests", click on "sweetpotato" and on "OK". Rcmdr reply with the following command in source pane (R Script)

```
data(sweetpotato, package="randtests")
sweetpotato <- as.data.frame(sweetpotato)
```

To build a confidence interval for sigma on variable "yield", select from Rcmdr menu: "Statistics" -> "Variances" -> "Single-sample Variance Test..." select "yield" and "OK". Rcmdr reply with the following command in source pane (R Script)

```
with(sweetpotato, sigma.test(yield[!is.na(yield)], alternative='two.sided', sigmasq=1.0, conf.level=0.9
```

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

For more information see [Rcmdr-package](#).

Para ayuda en español, véase [Test de varianza \(es\)](#). (For Spanish help see [Test de varianza \(es\)](#).)

sigmaMenu.es

Test de varianza

Description

Dentro del menú "Estadísticos" -> "Varianzas", se proporcionan una nueva entrada para calcular intervalos de confianza y hacer contrastes sobre la varianza en una población normal. Esta opción usa la función `sigma.test` del paquete `TeachingDemos`. Para más información véase [sigma.test](#).

Details

Ejemplo de uso del menú "Test de varianza para una muestra...".

Cargar los datos "sweetpotato" eligiendo del menu de Rcmdr: "Datos" -> "Conjuntos de datos en paquetes" -> "Leer conjunto de datos desde paquete adjunto..." pulsar dos veces sobre "randtests", pulsar sobre "sweetpotato" y sobre "Aceptar". Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
data(sweetpotato,package="randtests")
```

```
sweetpotato <- as.data.frame(sweetpotato)
```

Para construir el intervalo de confianza para sigma para la variable "yield", seleccione del menú de Rcmdr: "Estadísticos" -> "Varianzas" -> "Test de varianza para una muestra..." seleccione "yield" y "Aceptar". Rcmdr responde con la siguiente instrucción en el cuadro de instrucciones (R Script)

```
with(sweetpotato, sigma.test(yield[!is.na(yield)], alternative='two.sided', sigmasq=1.0, conf.level=0.9
```

Author(s)

Manuel Munoz-Marquez <manuel.munoz@uca.es>

See Also

Para más información véase [Rcmdr-package](#).

Index

* package

- paretochartMenu, [3](#)
 - paretochartMenu.es, [4](#)
 - predictMenu, [4](#)
 - predictMenu.es, [6](#)
 - qccMenu, [8](#)
 - qccMenu.es, [8](#)
 - randomnessMenu, [9](#)
 - randomnessMenu.es, [10](#)
 - RcmdrPlugin.UCA-package, [2](#)
 - RcmdrPlugin.UCA.es, [11](#)
 - sigmaMenu, [12](#)
 - sigmaMenu.es, [12](#)
- c-chart (qccMenu), [8](#)
- Gráfico de Pareto (es), [3](#)
- Gráficos de control de calidad (es), [8](#)
- Menú predicciones (es), [6](#)
- np-chart (qccMenu), [8](#)
- p-chart (qccMenu), [8](#)
- Pareto chart (paretochartMenu), [3](#)
- paretochart, [3](#), [4](#)
- paretochartMenu, [3](#)
- paretochartMenu.es, [4](#)
- Predict menu (predictMenu), [4](#)
- predictMenu, [4](#)
- predictMenu.es, [6](#)
- qcc, [8](#)
- qccMenu, [8](#)
- qccMenu.es, [8](#)
- Quality control graphs (qccMenu), [8](#)
- R-chart (qccMenu), [8](#)
- Randomness test, [11](#)
- Randomness test (randomnessMenu), [9](#)
- randomnessMenu, [9](#)
- randomnessMenu.es, [10](#)
- RcmdrPlugin.UCA
(RcmdrPlugin.UCA-package), [2](#)
- RcmdrPlugin.UCA-package, [2](#)
- RcmdrPlugin.UCA.es, [2](#), [11](#)
- runs.test, [9](#), [10](#)
- S-chart (qccMenu), [8](#)
- sigma.test, [12](#)
- sigmaMenu, [12](#)
- sigmaMenu.es, [12](#)
- Test de aleatoriedad (es), [9](#)
- Test de varianza (es), [12](#)
- u-chart (qccMenu), [8](#)
- Variance test (sigmaMenu), [12](#)
- xbar-chart (qccMenu), [8](#)