Package 'rgrs'

June 5, 2009

Type Package

Title Functions to make R usage in social sciences easier (in french)

Version 0.2-10

Date 2009-06-05

Author Julien Barnier <julien.barnier@ens-lsh.fr>

Maintainer Julien Barnier <julien.barnier@ens-lsh.fr>

Description This package provides functions for beginners and social sciences students or researchers. Currently it includes functions for cross-tabulation, weighting, results export, and maps plotting. The documentation and help pages are written in french.

License GPL (>= 2)

Encoding UTF-8

Depends

Suggests tcltk, odfWeave, R2HTML, RColorBrewer, sp

SystemRequirements xclip (Linux)

URL http://alea.fr.eu.org/j/rgrs.html

R topics documented:

`Cartographie` *Représentations cartographiques simples*

Description

Fonctions permettant la représentation cartographique simple de données de type proportions (`carte.prop`), effectifs (`carte.eff`) ou qualitatif (`carte.qual`).

Usage

```
carte.prop(sp, data, varname, sp.key="id", data.key="id", diverg=FALSE,
           nbcuts=6, at=NULL, at.lim=FALSE, main="", sub=NULL, posleg="topleft",
           palette.pos="Reds", palette.neg="Blues", palette=NULL, ...)

carte.eff(sp, data, varname, sp.key="id", data.key="id", nbcuts=4,
          at=NULL, main="", sub=NULL, posleg="topleft", col.bg="red",
          col.border="white", cex=5, pch=21, plot.polygons=TRUE, ...)

carte.qual(sp, data, varname, sp.key="id", data.key="id", main="", sub=NULL,
           posleg="topleft",  palette.qual="Set3", palette=NULL, ...)

carte.labels(sp, labels, coords=NULL, cex=1, font=2, col="black",
             outline=FALSE, outline.decal=1, outline.col="white")
```

Arguments

`sp`	objet spatial, de classe `SpatialPolygonsDataFrame`.
`data`	tableau de données contenant la variable à représenter.
`varname`	nom de la variable à représenter (sous forme de chaîne de caractère).
`sp.key`	nom de la variable de jointure de l'objet spatial.
`data.key`	nom de la variable de jointure du tableau de données.
`diverg`	si TRUE, les données comportent à la fois des valeurs positives et négatives, à représenter dans des schémas de couleur différents.

`nbcuts`	nombre de classes de valeurs pour la légende (`carte.eff`) et pour les couleurs (`carte.prop`).
`at`	bornes des classes de valeurs pour la légende (`carte.eff`) et pour les couleurs (`carte.prop`).
`at.lim`	si TRUE, les valeurs minimum et maximum sont ajoutées si besoin aux intervalles donnés via l'option `at` pour `carte.prop`.
`main`	titre de la carte.
`sub`	sous-titre de la carte.
`posleg`	position de la légende, à indiquer de la même manière que pour `legend`.
`col.bg`	couleur des symboles pour `carte.eff`.
`col.border`	couleur de la bordure des symboles pour `carte.eff`.
`cex`	facteur d'agrandissement des symboles (`carte.eff`) ou des labels (`carte.labels`).
`pch`	type de symbole pour `carte.eff`.
`plot.polygons`	si FALSE, le contenu de l'objet spatial (polygones) n'est pas affiché.
`palette.pos`	nom de la palette à utiliser pour les classes de valeurs positives. Chaîne de caractère transmise à RColorBrewer (voir Details).
`palette.neg`	nom de la palette à utiliser pour les classes de valeurs négatives. Chaîne de caractère transmise à RColorBrewer (voir Details).
`palette.qual`	nom de la palette à utiliser pour les catégories. Chaîne de caractère transmise à RColorBrewer (voir Details).
`palette`	palette de couleur spécifiée manuellement.
`labels`	vecteur de chaines de caractère contenant les labels à écrire.
`coords`	coordonnées de positionnement des labels. Si NULL, les coordonnées sont calculées en fonction de la forme de chaque polygone.
`col`	couleur des labels.
`font`	type de police utilisée pour les labels. Voir `par`.
`outline`	si TRUE, une bordure est affichée autour des labels.
`outline.decal`	décalage à utiliser pour les bordures de labels.
`outline.col`	couleur pour les bordures de labels.
`...`	paramètres supplémentaires passés à `spplot`.

Details

Pour la manière de spécifier des palettes à RColorBrewer, on pourra utiliser l'outil interactif à l'adresse http://colorbrewer.org, les noms de palette sont les mêmes. Si vous utilisez une des palettes du site, les auteurs du projet apprécient que celui-ci soit cité.

Si le nombre de classes de valeurs est trop élevé, il se peut que la palette spécifiée ne dispose pas de suffisamment de couleurs. Il faut alors soit réduire le nombre de classes, soit choisir une autre palette, soit spécifier une palette manuellement.

Value

Affiche la carte et la légende correspondantes. Ne retourne pas de valeur particulière.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

spplot, legend, brewer.pal, palette, par

Examples

```
data(lyon)
data(rp99)

require(sp)
plot(lyon)

carte.prop(lyon, rp99, "tx.chom", sp.key="DepCom", data.key="code")
carte.prop(lyon, rp99, "tx.chom", sp.key="DepCom", data.key="code",main="Taux de chomage 199

carte.eff(lyon, rp99, "pop.act", sp.key="DepCom", data.key="code")
carte.eff(lyon, rp99, "pop.act", sp.key="DepCom", data.key="code", main="Population active e

rp99$qual <- sample(c("A","B","C","D","E"), nrow(rp99),replace=TRUE)
carte.qual(lyon, rp99, "qual", sp.key="DepCom", data.key="code", main="Types d'arrondissemen

carte.prop(lyon, rp99, "tx.chom", sp.key="DepCom", data.key="code", main="Taux de chomage 19
carte.labels(lyon, lyon@data$Nom_Com, outline=TRUE)
```

---

copie *Export d'un objet au format HTML*

Description

Cette fonction transforme l'objet passé en argument en HTML via R2HTML, puis le place dans le presse-papier ou dans un fichier.

Usage

```
copie(obj, ...)
## Default S3 method:
copie(obj, append=FALSE, file=FALSE, filename="temp.html",...)
```

Arguments

`obj`	nom de l'objet à exporter
`append`	si FALSE (par défaut), remplace le contenu du presse-papier ou du fichier par le résultat. Si TRUE, ajoute le résultat à la suite du contenu du presse-papier ou du fichier
`file`	si FALSE (par défaut), exporte dans le presse-papier. Si TRUE, exporte dans le fichier filename
`filename`	nom du fichier dans lequel exporter l'objet, si file=TRUE
`...`	arguments passés à la fonction HTML()

Details

ATTENTION, pour l'instant cette fonction ne fonctionne que sous Windows en ce qui concerne la copie dans le presse-papier. Sous Linux elle nécessite la présence du programme xclip. Elle n'a pas pu être testée sous Mac OS X.

Value

Après exécution, si file=FALSE le presse-papier contient une copie de l'objet formaté en HTML. On peut alors facilement coller le résultat directement sous Microsoft Excel, puis dans Word avec un second copier/coller.

Si on positionne l'argument file à TRUE, l'objet est exporté dans un fichier (par défaut nommé temp.html et situé dans le répertoire de travail. On peut ensuite l'intégrer directement dans Microsoft Word ou OpenOffice Writer via le menu Insertion > Fichier.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`HTML`, `copie.proptab`

Examples

```
data(iris)
tab <- table(cut(iris$Sepal.Length,8),cut(iris$Sepal.Width,4))
## Not run: copie(tab)
```

`copie.proptab` *Export d'un objet proptab au format HTML*

Description

Applique la fonction générique `copie` à un tableau de classe `proptab`.

Usage

```
## S3 method for class 'proptab':
copie(obj, percent=NULL, digits=NULL, justify="right", ...)
```

Arguments

`obj`	nom de l'objet à exporter
`percent`	affichage du symbole pourcentage dans les cellules du tableau
`digits`	nombre de décimales à afficher
`justify`	justification du contenu des cellules (`"left"`, `"right"` ou `"centre"`)
`...`	arguments passés à la fonction copie()

Details

Pour plus d'informations sur les arguments et les résultats de la fonction, se référer à l'aide des fonctions `format.proptab` et `copie`.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`copie`, `format.proptab`

Examples

```
data(iris)
tab <- table(cut(iris$Sepal.Length,8),cut(iris$Sepal.Width,4))
ptab <- lprop(tab, percent=TRUE)
## Not run: copie(ptab)
```

cramer.v — *Calcule le V de Cramer d'un tableau croisé*

Description

Cette fonction calcule le V de Cramer pour un tableau de contingence.

Usage

```
cramer.v(tab)
```

Arguments

tab	Tableau croisé.

Details

Le tableau croisé passé en argument est un objet de type table.

Value

Valeur du V pour le tableau.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

Examples

```
v1 <- factor(round(runif(500,1,4)))
v2 <- factor(round(runif(500,1,3)))

tab <- table(v1,v2)
print(tab)
cramer.v(tab)
```

format.proptab — *Formate le contenu d'un tableau contenant des proportions*

Description

Cette fonction formate un tableau contenant des pourcentages en contrôlant leur présentation. Cette fonction est prévue pour une utilisation interne, et ne devrait pas être utilisée directement.

Usage

```
## S3 method for class 'proptab':
format(x, digits=NULL, percent=NULL, justify="right", ...)
```

Arguments

x	tableau à formater
digits	indique le nombre de décimales à conserver pour l'affichage. Si NULL, on utilise l'attribut digits de x
percent	indique si on doit afficher (TRUE) ou non (FALSE) le symbole % dans chaque case du tableau. Si NULL, on utilise l'attribut percent de x
justify	justification du contenu des cellules ("left", "right" ou "centre")
...	arguments passés à la fonction format()

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

copie.proptab, print.proptab

---

freq *Retourne le tri à plat d'une variable*

Description

Cette fonction affiche le tri à plat d'une variable (vecteur).

Usage

```
freq(x, digits=1, cum=FALSE, total=FALSE, exclude=NULL, sort="")
```

Arguments

x	vecteur pour lequel on souhaite obtenir le tri à plat ou tableau de dimension 1
digits	nombre de chiffres à conserver après la virgule
cum	si TRUE, affiche les pourcentages cumulés
total	si TRUE, affiche le total des effectifs
exclude	valeurs à exclure du tri à plat (aucune par défaut)
sort	si "inc", le tableau résultat est trié par effectifs croissants. Si "dec", par effectifs décroissants. Sinon, l'ordre des modalités par défaut est conservé.

Details

L'objet x est soit un vecteur, dans ce cas le tri à plat est calculé à l'aide de la fonction table, soit déjà un tri à plat, c'est-à-dire une table à une dimension, dans ce cas c'est cette table qui est utilisée telle quelle.

Value

Un data frame dont les noms de lignes sont les modalités de la variables, et dont les colonnes sont les effectifs, le pourcentage et (si demandé) le pourcentage cumulé de ces modalités.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`table`, `prop`

Examples

```
v <- c(round(runif(230,1,5)), NA)
freq(v)
freq(v, cum=TRUE)
freq(v, exclude=NA)
freq(v, exclude=c(1,2,NA))
freq(v, digits=3)
freq(v, total=TRUE)
freq(v, sort="inc")
tab <- table(v)
freq(tab)
```

---

genere.tableau *Génère une représentation ODF d'un objet*

Description

Cette fonction fait appel à la fonction odfTable() correspondant au type d'objet passé en paramètres.

Usage

```
genere.tableau(x, ...)
```

Arguments

`x`	objet à exporter
`...`	arguments passés à la fonction odfTable()

Details

Actuellement la fonction permet de générer une version ODF des objets de type table à une ou deux dimensions, des data frames, des matrices et des vecteurs.

Actuellement que cette fonction n'est qu'une interface à odfTable() qui évite de devoir convertir les objets de type table en matrice ou en data frame.

Value

Renvoit une représentation au format ODF (XML) de l'objet.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

odfTable

Examples

```
## Not run:
## Not run:
data(iris)
tab <- table(iris$Species)
genere.tableau(tab)
## End(Not run)
```

---

hdv2003 *Histoire de Vie 2003*

Description

Échantillon de 2000 individus et de 20 variables issu de l'enquête *Histoire de Vie* réalisé par l'INSEE en 2003.

Usage

```
data(hdv2003)
```

Format

Data frame comportant 2000 lignes et 20 colonnes

Source

Fichiers détail de l'INSEE : http://www.insee.fr/fr/themes/detail.asp?ref_id=fd-HDV03

lyon *Contour des arrondissements de Lyon*

Description

Contour des 9 arrondissements de Lyon pour représentation cartographique

Usage

```
data(lyon)
```

Format

Objet de classe SpatialPolygonsDataFrame

mls.export *Export de données vers Modalisa*

Description

Exporte un data frame dans un fichier texte importable ensuite sous Modalisa avec la fonction *Import ASCII*

Usage

```
mls.export(df, filename)
```

Arguments

df	data frame à exporter
filename	Nom du fichier d'export

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

mls.import

Examples

```
## Not run:
## Not run: mls.export(mydf, "export_modalisa.txt")
```

`mls.import` *Import de fichiers Modalisa*

Description

Importe un fichier Modalisa enregistré sous forme d'export ASCII

Usage

```
mls.import(filename, enc = "latin1", modif.names = TRUE)
```

Arguments

`filename`	Nom du fichier à importer
`enc`	Encodage du fichier à importer (normalement toujours latin1)
`modif.names`	Correction ou non des noms de variables. Si `modif.names` vaut TRUE, alors les noms de variables importés sont convertis en minuscules et les espaces remplacés par des tirets bas.

Value

Retourne un data frame contenant les données importées.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`mls.export`, `mls.eclate.multi`, `mls.eclate.ordo`, `mls.table.multi`

Examples

```
## Not run:
## Not run: mydf <- mls.import("export_modalisa.TXT")
```

`Pondérations` *Traitement de données pondérées*

Description

Fonctions permettant le calcul de moyennes (wtd.mean), variances (wtd.var), tris à plat et tableaux croisés (wtd.table) pour des variables pondérées.

Usage

```
wtd.mean(x, weights = NULL, normwt = "ignored", na.rm = TRUE)
wtd.var(x, weights = NULL, normwt = FALSE, na.rm = TRUE)
wtd.table(x, y = NULL, weights = NULL, normwt = FALSE, na.rm = TRUE)
```

Arguments

`x, y`	Vecteurs de données. Doit être numérique pour wtd.mean et wtd.var.
`weights`	Vecteur des poids. Doit être de même longueur que x
`normwt`	Normalisation des poids pour que les effectifs totaux pondérés soient les mêmes que les effectifs initiaux
`na.rm`	Suppression des valeurs manquantes

Details

Si `weights` n'est pas fourni, les fonctions utilisent une pondération uniforme.

Value

Pour `wtd.table`, si un seul vecteur est fourni la fonction calcule le tri à plat pondéré de la variables. Si deux vecteurs sont passés en paramètres on obtient le tri croisé pondéré des deux variables.

Author(s)

Les fonctions `wtd.mean` et `wtd.var` sont des copies conformes des fonctions du même nom de l'extension `Hmisc`. Elles ont été développées par : Frank Harrell Department of Biostatistics Vanderbilt University School of Medicine f.harrell@vanderbilt.edu

Pour `wtd.table` : Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`table`, `mean`, `var`, `wtd.table`, `wtd.quantile` et l'extension survey

Examples

```
data(hdv2003)

mean(hdv2003$age)
wtd.mean(hdv2003$age, weights=hdv2003$poids)

table(hdv2003$sexe)
wtd.table(hdv2003$sexe, weights=hdv2003$poids)
wtd.table(hdv2003$sexe, weights=hdv2003$poids, normwt=TRUE)

table(hdv2003$sexe, hdv2003$hard.rock)
wtd.table(hdv2003$sexe, hdv2003$hard.rock, weights=hdv2003$poids)
```

---

print.proptab *Affiche un tableau contenant des proportions*

Description

Cette fonction affiche un tableau contenant des pourcentages en contrôlant leur présentation.

Usage

```
## S3 method for class 'proptab':
print(x, digits=NULL, percent=NULL, justify="right", ...)
```

Arguments

x	tableau à afficher
digits	indique le nombre de décimales à conserver pour l'affichage. Si NULL, on utilise l'attribut digits de tab
percent	indique si on doit afficher (TRUE) ou non (FALSE) le symbole % dans chaque case du tableau. Si NULL, on utilise l'attribut percent de tab
justify	justification du contenu des cellules ("left", "right" ou "centre")
...	arguments passés à la fonction print.table()

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

format.proptab, Profils, print

Examples

```
tab <- table(x=round(runif(100,1,3)),y=round(runif(100,1,5)))
ptab <- lprop(tab, digits=1, percent=TRUE)
print(ptab)
print(ptab, digits=2, percent=FALSE)
```

`Profils` *Fonctions de calcul de pourcentages à partir d'un tableau croisé*

Description

Fonctions calculant différents pourcentages d'un tableau croisé

Usage

```
cprop(tab, digits = 1, total = TRUE, percent = FALSE)
lprop(tab, digits = 1, total = TRUE, percent = FALSE)
prop(tab, digits = 1, total = TRUE, percent = FALSE)
thprop(tab, digits = 1, percent = FALSE)
theff(tab, digits = 2)
residus(tab, digits = 2)
```

Arguments

`tab`	Tableau croisé (objet de type table)
`digits`	Nombre de chiffres après la virgule à conserver à l'affichage
`total`	Ajouter des lignes/colonnes pour les marges du tableau
`percent`	Ajout du symbole % dans chaque case lors de l'affichage du tableau

Details

Dans le cas des tableaux contenant des proportions (`cprop`, `lprop`, `prop`, `thprop`), les options `digits` et `percent` sont des attributs du tableau résultant qui contrôlent l'affichage du tableau avec `print` ou `copie`. On peut modifier ponctuellement ces options en les passant directement à `print.proptab` `copie.proptab`. Les données numériques stockées conservent l'intégralité des valeurs décimales.

Value

`cprop` retourne un tableau contenant les pourcentages colonnes, `lprop` renvoit un tableau contenant les poiurcentages lignes, `prop` renvoit un tableau contenant les pourcentages globaux, `thprop` renvoit un tableau de pourcentages théoriques sous l'hypothèse d'indépendance, `theff` renvoit un tableau d'effectifs théoriques sous l'hypothèse d'indépendance, et `residus` renvoit le tableau des résidus de Pearson.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`table`, `prop.table`, `sweep`, `chisq.test`, `print.proptab`, `copie.proptab`

Examples

```
v1 <- factor(round(runif(500,1,4)))
v2 <- factor(round(runif(500,1,3)))

tab <- table(v1,v2)
tab
lprop(tab,digits=5)
cprop(tab,digits=2)
prop(tab)
thprop(tab, percent=TRUE)
theff(tab)
residus(tab)
```

---

quant.cut *Transforme une variable quantitative en variable qualitative*

Description

Cette fonction transforme une variable quantitative en une variable qualitative ayant des modalités comportant les même effectifs.

Usage

```
quant.cut(var, nbclass, include.lowest=TRUE, right=FALSE, dig.lab=5, ...)
```

Arguments

`var`	variable (vecteur) à transformer
`nbclass`	nombre de classes souhaité
`include.lowest, right, dig.lab, ...`	paramètres passés à la fonction cut

Details

Il s'agit juste d'un wrapper autour des fonctions cut et quantile

Value

Renvoit un vecteur de type factor généré par cut

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`cut`, `quantile`

Examples

```
data(iris)
sepal.width3cl <- quant.cut(iris$Sepal.Width,3)
freq(sepal.width3cl)
```

Questions multiples Modalisa

Traitement des questions à réponses multiples importées depuis Modalisa.

Description

Ces fonctions permettent de transformer ou de traiter des questions à réponses multiples importées depuis un export Modalisa.

Usage

```
mls.table.multi(var)
mls.eclate.multi(var, vname = "MLS.mult.", mnames = NULL)
mls.eclate.ordo(var, vname = "MLS.ordo.", mnames = NULL, nb = 3)
```

Arguments

`var`	Variable correspondant à une question à réponses multiples
`vname`	Préfixe à ajouter aux noms des variables générées (si `mnames=NULL`)
`mnames`	Noms des variables générées
`nb`	Dans le cas de questions à réponses multiples ordonnées, nombre de modalités à retenir

Details

`mls.table.multi` génère le tri à plat des modalités d'une question à réponses multiples. `mls.eclate.multi` transforme une question à réponses multiples en autant de questions binaires qu'il y a de modalités. `mls.eclate.ordo` fait la même chose pour une question à réponses multiples ordonnées.

Value

`mls.table.multi` renvoit une table contenant le tri à plat des modalités de la question à réponses multiples.

`mls.table.multi` renvoit un data frame avec autant de variables que la variable initiale a de modalités. Chaque variable créée possède les modalités `O` si l'individu correspondant a choisi la modalité, et `N` sinon.

`mls.table.ordo` renvoit un data frame semblable à celui généré par `mls.table.multi`, mais il se limite aux `nb` premières modalités.

Note

De manière générale, il est préférable de toujours éviter l'usage de questions à réponses multiples dans Modalisa, et de privilégier dès la conception du questionnaire l'usage de séries de questions binaires.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`mls.import`

Examples

```
## Not run:
## Not run:
mydf <- mls.import("export_modalisa.TXT")
mls.table.multi(mydf$couleurs)
test <- cbind(test,mls.eclate.multi(mydf$couleurs, vname="couleur"))
## End(Not run)
```

`renomme.variable` *Renomme une colonne d'un tableau de données*

Description

Renomme une colonne (variable) d'un tableau de données

Usage

```
renomme.variable(df, old, new)
```

Arguments

`df`	tableau de données (data.frame)
`old`	nom de la variable à renommer
`new`	nouveau nom

Value

Renvoit un tableau de données avec la colonne indiquée renommée

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

Examples

```
data(iris)
str(iris)
iris <- renomme.variable(iris, "Species", "especes")
str(iris)
```

---

rgrs.update *Mise à jour du paquet Rgrs*

Description

Vérifie si une nouvelle version de Rgrs est disponible et effectue la mise à jour si nécessaire.

Usage

```
rgrs.update()
```

Details

Nécessite d'avoir une connexion à Internet active pour pouvoir contacter le serveur R-forge.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

update.packages

rp99 *Recensement 1999 - Communes du Rhône*

Description

Résultats tirés du recensement de la population de 1999 pour les communes du Rhône.

Usage

```
data(rp99)
```

Format

Data frame comportant 301 lignes et 21 colonnes

Source

Bases de données recensement de l'INSEE : http://www.insee.fr/fr/bases-de-donnees/default.asp?page=recensements.htm

selectwd *Sélecteur de répertoire de travail*

Description

Affiche ue boîte de sélection de répertoire en Tk et modifie le répertoire de travail selon la sélection.

Usage

```
selectwd()
```

Value

La fonction renvoit le répertoire choisi sous forme de chaîne de caractères, et affiche la commande `setwd` correspondant à la sélection effectuée.

Author(s)

Julien Barnier <julien.barnier@ens-lsh.fr>

See Also

`setwd`, `getwd`

Index