

Package ‘Rcoisas’

December 14, 2023

Type Package

Title Minhas Facilidades

Version 0.0.0.908

Date 2023-12-14

Author Fúlvio B. Nedel

Maintainer Fúlvio B. Nedel <fulvionedel@gmail.com>

Description Funções para criação de pirâmides populacionais, descrição e gráficos de variáveis, etc., com 'outputs' em português.

License GPL-3

Imports e1071,
ggplot2,
graphics,
stats,
dplyr

Suggests foreign,
ggthemes,
ggpp,
Hmisc,
lemon,
stringr

Encoding UTF-8

Language pt-br

LazyData true

RoxygenNote 7.2.3

Depends R (>= 2.10)

R topics documented:

adissoma	2
bolero	3
descreve	4
ed	6
formatL	7
fxetar.det_pra_fxetar5	8
fxetarNM	9

ggplot_pir	10
histobox	12
obitosRS2019	13
plot.histobox	13
plotZ	14
plot_pir	15
POPBR12	16
POPRS2019	17
print.descreve	17
RDRS2019	18
sacamostras	18
tabuleiro	19
tabuleiro2	21
zeroesq	21

Index	23
--------------	-----------

adissoma	<i>Totais de colunas em uma tabela</i>
----------	--

Description

Em uma tabela, acrescenta uma linha com o total das colunas, ou soma dos valores das colunas em linhas selecionadas.

Usage

```
adissoma(tabela, rotulo = "Total")
```

Arguments

tabela	Objeto de estrutura matricial de no mínimo duas linhas e duas colunas
rotulo	Rótulo para a linha da soma, o padrão é "Total".

Value

A tabela (de classe `data.frame`) com a linha da soma ao final.

Examples

```
tabela <- "'Faixa Etária det' Masc Fem Ign Total
      '0 a 6 dias' 361 257 5 623
      '7 a 27 dias' 156 111 - 267
      '28 a 364 dias' 155 148 1 304
      '1 a 4 anos' 144 88 - 232
      '5 a 9 anos' 64 46 - 110
      '10 a 14 anos' 83 68 - 151
      '15 a 19 anos' 462 141 - 603
      '20 a 24 anos' 817 271 - 1088
      '25 a 29 anos' 880 393 - 1273
      '30 a 34 anos' 1155 561 - 1716
      '35 a 39 anos' 1508 870 1 2379
      '40 a 44 anos' 1954 1190 - 3144
```

```

      '45 a 49 anos' 2514 1681 - 4195
      '50 a 54 anos' 3488 2175 2 5665
      '55 a 59 anos' 5170 3265 2 8437
      '60 a 64 anos' 6557 4240 2 10799
      '65 a 69 anos' 7566 5370 1 12937
      '70 a 74 anos' 7989 6181 3 14173
      '75 a 79 anos' 7438 6508 4 13950
      '80 anos e mais' 14444 21159 7 35610"
tabela <- read.table(header = TRUE, na.strings = "-", text = tabela)
adissoma(tabela)

rbind(adissoma(tabela[1:4,], rotulo = "0 a 4 anos")[5,],
      tabela[-c(1:4),]
    ) |>
adissoma()

```

bolero

Bolero: tabelas 2x2

Description

Analisa uma tabela 2x2 e apresenta um output com rótulos em português

Usage

```
bolero(independente, dependente = NULL, dec = 2, dnn = NULL)
```

Arguments

independente	Variável independente
dependente	Variável dependente
dec	No. de decimais
dnn	Nome das variáveis a ser impresso no 'output'

Value

Um objeto da classe 'list' com as tabelas de frequências absolutas e relativas, razão de probabilidades e de odds, com seus intervalos de confiança e valores-p.

Examples

```
with(RDRS2019, bolero(SEXO, MORTE))
```

descreve*Descreve uma variável numérica*

Description

Executa a descrição "completa" de uma variável numérica, contínua ou discreta, e desenha um histograma, possivelmente com linhas para a média, mediana e distância da média até 1 ou 2 desvios-padrão, além da curva de probabilidade Normal para os parâmetros apresentados.

Usage

```
descreve(  
  x,  
  by = NULL,  
  dec = 2,  
  na.rm = TRUE,  
  data = NULL,  
  histograma = TRUE,  
  breaks = "Sturges",  
  freq = TRUE,  
  main = NULL,  
  xlab = NULL,  
  ylab = NULL,  
  linhas = 2,  
  curva = TRUE,  
  densidade = FALSE,  
  col.dens = 1,  
  col = "yellow2",  
  col.curva = "DarkGreen",  
  col.media = 2,  
  col.dp = col.media,  
  col.mediana = 4,  
  legenda = TRUE,  
  lugar = "topright",  
  lty.curva = 2,  
  lwd.curva = 1,  
  lty.dens = 3,  
  lwd.dens = 2,  
  lty = NULL,  
  lwd = NULL,  
  cex = NULL,  
  print = "output",  
  soma = FALSE,  
  ...  
)
```

Arguments

x	Um vetor numérico
by	tentar incluir esse argumento
dec	Número de dígitos

na.rm	TRUE (default) remove os missings
data	Argumento opcional. Banco de dados contendo x. O padrão é NULL.
histograma	TRUE (default) desenha um histograma
breaks	Número de divisões, de acordo com hist ; o método padrão é Sturges'
freq	TRUE (default) define o eixo y como frequência; FAISE o define como densidade
main	Título do gráfico
xlab	Rótulo do eixo x
ylab	Rótulo do eixo y
linhas	TRUE (default) desenha linhas verticais com a média, mediana e média + 1 ou 2 DP
curva	TRUE (default) desenha a curva normal esperada
densidade	O padrão é FALSE; passe para TRUE para desenhar a curva de densidade da distribuição
col.dens	Cor da curva de densidade; o padrão é 'black'
col	Cor das barras do histograma; o padrão é 'yellow2'
col.curva	Cor da curva normal; o padrão é 'DarkGreen'
col.media	Cor da linha da média; o padrão é 'red'
col.dp	Cor das linhas do SP; por padrão é a mesma que col.media ('red')
col.mediana	Cor da linha da mediana; o padrão é 'blue'
legenda	TRUE (default) desenha a legenda
lugar	Posição da legenda; padrão é 'topright'
lty.curva	Tipo da linha da curva normal; padrão é 2
lwd.curva	Largura da linha da curva normal; padrão é 1
lty.dens	Tipo da linha da curva de densidade; padrão é 3
lwd.dens	Largura da linha da curva de densidade; padrão é 2
lty	Tipo da linha para o histograma; padrão é NULL
lwd	Largura da linha para o histograma; padrão é NULL
cex	Tamanho da fonte
print	Modo de apresentação; print = "tabela" retorna uma tabela com as estatísticas
soma	TRUE apresenta a soma dos valores; padrão é FALSE
...	Toma parâmetros de outras funções utilizadas

See Also

[hist](#) and [par](#) para os parâmetros gráficos

Examples

```
descreve(rnorm(100))
descreve(rnorm(100), breaks = 'Scott')
descreve(rnorm(100), breaks = 'Scott', linhas = FALSE, densidade = TRUE, col = 'skyblue')
descreve(rnorm(100), print = "tabela")
obitosRS2019$sexo <- factor(obitosRS2019$sexo)
descreve(obitosRS2019$idade)
## Not run:
# NÃO FUNCIONA!
descreve(idade, obitosRS2019$sexo, data = obitosRS2019)
# Esse funciona
with(obitosRS2019, descreve(idade, sexo))

## End(Not run)
```

ed

Seleciona as últimas colunas de um banco de dados

Description

Enquanto `tail` mostra as últimas linhas de uma matriz ou "data frame", `ed()` mostra as últimas colunas de uma matriz ou "data frame".

Usage

```
ed(x, n = 5)
```

Arguments

x	Banco de dados (objeto da classe <code>data.frame</code> , <code>matrix</code> ou <code>table</code>)
n	Número de colunas a selecionar (por padrão são 5)

Value

Um objeto de classe `data.frame` com as últimas colunas selecionadas de uma matriz ou banco de dados.

Examples

```
ed(RDRS2019) |> # As últimas 5 colunas do banco
head()
ed(RDRS2019, 8) |> # As últimas 8 colunas do banco
head()
ed(RDRS2019, 1) |> # A última coluna do banco
head()
```

formatLNúmeros em formato latino

Description

Apresenta números em formato latino, com ponto (.) como separador de milhar e vírgula (,) como separador decimal, mudando a classe do objeto de `numeric` para `character`. A função simplesmente altera os padrões de [formatC](#) para `big.mark = "."` e `decimal.mark = ","`.

Usage

```
formatL(x, digits = 1, format = "f", ...)
```

Arguments

<code>x</code>	Um número ou vetor com números
<code>digits</code>	Número de decimais
<code>format</code>	Formato numérico (ver formatC)
<code>...</code>	Permite outros argumentos da função formatC

Value

Um vetor de classe `character` com os valores formatados para impressão.

See Also

[formatC](#), [format](#)

Examples

```
formatL(1234.5678)
formatL(rnorm(5), digits = 2)
set.seed(1)
x <- c(rnorm(5, 30, 10), rep(20, 2), 25)
x
formatL(x)
formatL(x, format = "fg", digits = 3)
## Not run:
> formatL(x, format = "fg", digits = 3) |>
+   stringr::str_trim()
[1] "23,7" "31,8" "21,6" "46"   "33,3" "20"   "20"   "25"

## End(Not run)
```

```
fxetar.det_pra_fxetar5
```

Transforma a "faixa etária detalhada" (DATASUS) em 17 faixas quinquenais.

Description

Reclassifica as idades < 20 anos em faixas etárias quinquenais.

Usage

```
fxetar.det_pra_fxetar5(x, tipo = "POPBR")
```

Arguments

<code>x</code>	Um vetor com a idade categorizada segundo: (1) os arquivos de população "POPBR??.DBF" (até o ano 2012) disponibilizados pelo DATASUS, ou (2) um <code>data.frame</code> com o resultado de uma tabulação com a opção "Faixa etária detalhada" no TABNET ou TabWin (v. detalhes)
<code>tipo</code>	Argumento obrigatório indicando a origem dos dados, se um arquivo de população do DATASUS ou uma tabulação do TABNET ou TabWin (<code>tipo = "tabela"</code>). O padrão é <code>tipo = "POPBR"</code> . V. detalhes.

Details

Os arquivos "POPBR??.DBF" têm a idade em anos completos até 19 anos, faixas quinquenais de 20-24 até 75-79 anos e 80 e + anos. A "faixa etária detalhada" é uma opção de tabulação dos dados de mortalidade nos aplicativos TABNET (<https://datasus.saude.gov.br/informacoes-de-saude-tabnet/>) e TabWin, do DATASUS. A idade é detalhada nos componentes da Taxa de Mortalidade Infantil, < 1 ano não especificado, 1-4 anos, faixas quinquenais dos 5 aos 79 anos e 80 e mais. Nas opções de tabulação on-line da "morbidade hospitalar", esses mesmos cortes são usados para a definição da "Faixa Etária 2".

Value

Se `tipo = "POPBR"`, um vetor da classe `character` com a idade categorizada em 17 faixas etárias: quinquenais de 0 a 79 anos e 80 e + anos de idade. Se `tipo = "tabela"`, uma tabela (de classe `data.frame`) com a "faixa etária detalhada" agregada nessas 17 faixas etárias.

Examples

```
data("POPBR12")
str(POPBR12)
xtabs(POPULACAO ~ FXETARIA + SEXO, POPBR12)
POPBR12$FXETAR5 <- fxetar.det_pra_fxetar5(POPBR12$FXETARIA)
str(POPBR12)
xtabs(POPULACAO ~ FXETAR5 + SEXO, POPBR12)
# Um arquivo csv de uma tabulação da Declaração de Óbito no TABNET
# (residentes no RS, 2021):
## Not run:
df <- read.csv2("sim_cnv_obt10rs195350189_4_122_229.csv",
                skip = 3, nrows = 21, encoding = "latin1") |>
```



```
dplyr::select(-Ign) |>
fxetar.det_pra_fxetar5(tipo = 'tabela')

## End(Not run)
# Os valores são os seguintes:
dors21 <- data.frame(
  fxetar.det = c("0 a 6 dias", "7 a 27 dias", "28 a 364 dias", "1 a 4 anos",
    "5 a 9 anos", "10 a 14 anos", "15 a 19 anos", "20 a 24 anos",
    "25 a 29 anos", "30 a 34 anos", "35 a 39 anos", "40 a 44 anos",
    "45 a 49 anos", "50 a 54 anos", "55 a 59 anos", "60 a 64 anos",
    "65 a 69 anos", "70 a 74 anos", "75 a 79 anos", "80 anos e mais"),
  masc = c(361, 156, 155, 144, 64, 83, 462, 817, 880, 1155, 1508, 1954, 2514,
    3488, 5170, 6557, 7566, 7989, 7438, 14444),
  fem = c(257, 111, 148, 88, 46, 68, 141, 271, 393, 561, 870, 1190, 1681,
    2175, 3265, 4240, 5370, 6181, 6508, 21159))
dors21
fxetar.det_pra_fxetar5(dors21, tipo = "tabela")

dors21$total <- dors21$masc + dors21$fem
fxetar.det_pra_fxetar5(dors21, tipo = "tabela")
```

fxetarNM

*Faixa etária nas categorias de Nelson de Moraes***Description**

Recodifica vetores com a idade em números inteiros ou em faixas etárias quinquenais em três grandes faixas etárias: 0-14, 15-59 e 60 e + anos.

Usage

```
fxetarNM(idade = NULL, fxetardet = NULL, grafico = FALSE, ...)
```

Arguments

idade	Vetor numérico representando a idade em valores contínuos ou inteiros.
fxetardet	Vetor da classe factor ou character com diferentes categorizações de faixa etária disponíveis para tabulação no TABNET ("faixa etária detalhada", e outras formas com detalhamento de < 1 ano, 1 a 4 anos e faixas quinquenais ou decenais que permitem o cálculo) representando 17 faixas etárias quinquenais, rotuladas conforme o resultado da função <code>csapAIH: :fxetar_quinq</code> (números separados por hífen, sem espaços: "0-4", ..., "75-79", "80 e +").
grafico	Vetor lógico, FALSE por padrão. Se TRUE, desenha o gráfico da curva.
...	Permite a definição de outros parâmetros gráficos.

Value

`_se` fornecida a `idade_`, devolve um vetor com as frequências observada em cada faixa etária; `_se` fornecida a `faixa etária_`, devolve um vetor da classe caractere com as frequências de cada faixa etária; `_se` não são fornecidas nem a `idade` nem a `faixa etária_` – com `'fxetarNM()'` –, a função devolve um vetor com os nomes das faixas etárias.

Examples

```
# Apenas citar os grupos:
fxetarNM()

# Observar a curva na amostra de óbitos no RS em 2019:
## a idade já foi computada, com a função \code{idadeSUS}, do pacote \code{csapAIH}, com
## csapAIH::idadeSUS(obitosRS2019, sis = "SIM")
## Computar as faixas etárias de Nelson de Moraes
fxetarNM(obitosRS2019$idade) |> table()

## Desenhar a curva (podem-se usar os parâmetros gráficos)
fxetarNM(obitosRS2019$idade, grafico = TRUE,
         col.sub = 4, font.sub = 3, cex.sub = .8, cex.main = .95,
         main = "Curva de Nelson de Moraes. RS, 2019.",
         sub = "\nAmostra aleatória de 10.000 óbitos.")

# ## Tabela do TABNET
# Os óbitos de residentes do RS em 2021 por faixa etária detalhada podem ser tabulados no TABNET,
# em http://tabnet.datasus.gov.br/cgi/deftohtm.exe?sim/cnv/obt10rs.def, resultando (em 21/05/2023)
# na seguinte tabela (http://tabnet.datasus.gov.br/cgi/tabcgi.exe?sim/cnv/obt10rs.def)
obitosRS2021 <- data.frame(
  fxetar = c("0 a 6 dias", "7 a 27 dias", "28 a 364 dias", "1 a 4 anos", "5 a 9 anos",
            "10 a 14 anos", "15 a 19 anos", "20 a 24 anos", "25 a 29 anos", "30 a 34 anos",
            "35 a 39 anos", "40 a 44 anos", "45 a 49 anos", "50 a 54 anos", "55 a 59 anos",
            "60 a 64 anos", "65 a 69 anos", "70 a 74 anos", "75 a 79 anos", "80 anos e mais",
            "Idade ignorada"),
  casos = c(623, 267, 304, 232, 110, 151, 603, 1088, 1273, 1716, 2379, 3144, 4195, 5665, 8437,
            10799, 12937, 14173, 13950, 35610, 66)
)
# Se no TABNET a tabela for salva com a opção "copia como CSV", e depois salva em formato .xlsx,
# poderia ser lida com o comando
## Not run:
obitosRS <- readxl::read_excel("~/Downloads/teste-fxetardet.xlsx", range = "A4:B25")

## End(Not run)
# Com essa tabela, os seguintes comandos desenham a curva de Nelson de Moraes:
xtabs(casos/sum(casos)*100 ~ fxetarNM(fxetardet = obitosRS2021$fxetar), data = obitosRS2021) |>
plot(type = 'l', main = "Curva de Nelson de Moraes.\nRio Grande do Sul, 2021.",
     xlab = "Faixa etária (anos)", ylab = "% dos óbitos")
```

ggplot_pir

Pirâmides populacionais

Description

Pirâmides populacionais

Usage

```
ggplot_pir(
  banco,
  idade,
```

```

    sexo,
    populacao = NULL,
    catsexo = c("masc", "fem"),
    cores = c("steelblue", "hotpink"),
    nsize = 3.5
  )

```

Arguments

banco	Um data frame com população ou casos por sexo e faixa etária.
idade	Nome da variável com a idade ou faixa etária deve ir entre aspas.
sexo	Nome da variável com o sexo; deve ir entre aspas e ser um factor.
populacao	Nome da variável (entre aspas) com a população ou casos, se houver; por padrão é NULL, isto é, a função espera um banco de registros individuais e calcula a frequência por sexo e faixa etária indicada.
catsexo	Vetor com o nome das categorias da variável sexo; padrão é c("masc", "fem"); o sexo masculino deve ser a primeira categoria.
cores	Cores das barras, para as categorias masculino e feminino. O padrão é c("darkblue", "violetred").
nsize	Tamanho do texto com o n° total de habitantes. O padrão é 3.5.

Value

Um objeto de classe gg e ggplot com o gráfico da pirâmide populacional.

Examples

```

data("POPBR12")
str(POPBR12)
# "SEX0" é da classe 'integer';
# além disso, os primeiros 19 anos de idade estão em faixas anuais
# e os demais (até 80) em faixas quinquenais.
## Not run:
ggplot_pir(POPBR12, "FXETARIA", "SEX0", "POPULACAO")

## End(Not run)
# não dá o gráfico desejado.

# Transformando o sexo em fator e reclassificando a faixa etária:
require(dplyr)
POPBR12 <- POPBR12 %>%
  mutate(SEX0 = factor(SEX0, labels = c("masc", "fem")),
         FXETAR5 = fxetar.det_pra_fxetar5(FXETARIA))
ggplot_pir(POPBR12, "FXETAR5", "SEX0", "POPULACAO")

# Cerro Largo, RS
ggplot_pir(POPBR12[POPBR12$MUNIC_RES==430520, ],
           "FXETAR5", "SEX0", "POPULACAO")

## Mortalidade por anos completos de vida

# Se o banco é de registros individuais, sem uma variável com a
# contagem da população, a função conta as frequências em cada

```

```
# sexo e faixa etária específicos.

data("obitosRS2019")
str(obitosRS2019)

obitosRS2019 %>%
  mutate(sexo = factor(sexo, levels = c("masc", "fem"))) %>%
  ggplot_pir("idade", "sexo")

# O comando acima devolve o mesmo gráfico que o comando abaixo:

obitosRS2019 %>%
  mutate(sexo = factor(sexo, levels = c("masc", "fem"))) %>%
  count(idade, sexo) %>% # linha desnecessária
  ggplot_pir("idade", "sexo", "n")
```

histobox

Histograma com boxplot

Description

Desenha um histograma com boxplot integrado ao mesmo gráfico

Usage

```
histobox(
  x,
  limites = NULL,
  col.h = "cyan",
  col.bx = "skyblue1",
  maresq = 4.5,
  mardir = 2.1,
  ...
)
```

Arguments

x	Uma variável numérica
limites	Limites do eixo x
col.h	Cor do histograma
col.bx	Cor do boxplot
maresq	Margem esquerda do gráfico
mardir	Margem direita do gráfico
...	Outros parâmetros de hist e boxplot

Value

O desenho do gráfico

Examples

```
varnum <- rnorm(1000)
histobox(varnum)
histobox(varnum, ylab = "Frequência")
histobox(varnum, maresq = 2.8)
# Se a margem esquerda for zero, o eixo y é removido:
histobox(varnum, col.h = "tomato", col.bx = "yellow", maresq = 0)
histobox(varnum, col.h = "tomato", col.bx = "yellow", mardir = 0)
histobox(varnum, col.h = "tomato", col.bx = "yellow", maresq = 0, mardir = 0)
```

obitosRS2019

*Registros de óbito***Description**

Amostra de Declarações de Óbito (DO) de residentes no RS, 2019.

Usage

```
obitosRS2019
```

Format

Banco de dados com amostra aleatória de 10.000 DO de 2019 de residentes do RS, registrados no Sistema de Informação de Mortalidade (SIM). O banco foi previamente trabalhado para a decodificação da idade, de modo que a idade represente a idade em anos completos (0 para < 1 ano), e para a definição de missings e rótulos no campo sexo. Essas variáveis foram acrescentadas (com nome em letra minúscula, 'idade' e 'sexo'), sendo mantidas as originais ('IDADE', 'SEXO').

Source

Sistema de Informações sobre Mortalidade (SIM)

plot.histobox

*Método para histobox***Description**

Método para histobox

Usage

```
## S3 method for class 'histobox'
plot(x, ...)
```

Arguments

x Uma variável numérica
 ... Permite a inclusão de argumentos para [hist](#)

plotZ*Gráfico da probabilidade de pertencer a uma área da curva Normal*

Description

Gráfico da probabilidade de pertencer a uma área da curva Normal

Usage

```
plotZ(  
  x = NULL,  
  mu = 0,  
  dp = 1,  
  p = NULL,  
  z = NULL,  
  cor = 2,  
  main = NULL,  
  sub = NULL,  
  area = "abaixo",  
  cex.main = 2,  
  cex.sub = 1.5,  
  cex.axis = 1.3,  
  ...  
)
```

Arguments

x	valor a comparar com a média
mu	média
dp	desvio-padrão
p	probabilidade
z	escore-z
cor	cor do preenchimento da área sob a curva
main	título
sub	subtítulo
area	"abaixo" (padrão) calcula e desenha a probabilidade de um valor menor ou igual a x, p ou z; "acima" calcula e desenha a probabilidade de um valor maior que x, p ou z
cex.main	tamanho da fonte do título
cex.sub	tamanho da fonte do subtítulo
cex.axis	tamanho da fonte do eixo
...	Permite o uso de outros parâmetros gráficos (ver o uso de 'density', nos exemplos)

Value

O gráfico (objeto de classe NULL) com a área de probabilidade achurada.

Examples

```

plotZ(p = .975)
plotZ(p = .025)
plotZ(z = 1.96, density = 20)
plotZ(z = -1.96, density = 40)
plotZ(x = 10, mu = 4.7, dp = 2.7, cor = "yellow")
plotZ(x = 10, mu = 4.7, dp = 2.7, area = "acima")
plotZ(p = .975, area = "acima")
plotZ(p = .025, area = "acima")
plotZ(z = 1.96, area = "acima")
plotZ(z = -1.96, area = "acima")
# plotZ(x = c(8, 10), mu = 4.7, dp = 2.7)

```

plot_pir	<i>Pirâmides populacionais com os arquivos de população disponibilizados pelo DATASUS</i>
----------	---

Description

Pirâmides populacionais com os arquivos de população disponibilizados pelo DATASUS

Usage

```

plot_pir(
  pop,
  tabela = FALSE,
  ano = NULL,
  local = "população",
  title = NULL,
  fontsize = 1.1,
  nfxetar = 17,
  axes = FALSE,
  fonte = NULL,
  x.lim = NULL,
  colmasc = "mediumblue",
  colfem = "red2",
  colfxetar = "white",
  border = par("fg"),
  inside = T,
  drop.unused.levels = FALSE,
  ...
)

```

Arguments

pop	Um data frame com a estrutura dos arquivos "POPBR??DBF" disponibilizados pelo DATASUS, ou uma tabela com o sexo nas colunas (masc, fem) e a idade em 17 faixas etárias (0-4, ... 75-79, 80+) nas linhas
tabela	Argumento lógico. Padrão é FALSE, deve ser mudado para TRUE quando a população é uma tabela como indicado acima

ano	Indicar o período de referência. Apenas para quando a população é uma tabela
local	Para o título: de onde é a população representada?
title	Título do gráfico
fontsize	Tamanho de fonte do título do gráfico
nfxetar	No. de faixas etárias a serem representadas. Por enquanto única opção é 17 (quinquenais até 80 e +), tá na espera pra incluir outras opções
axes	Não sei se vale a pena ir como opção ou já colocar direto o padrão FALSE. **Ver melhor**
fonte	Texto para citar a fonte ao pé do gráfico
x.lim	Limite do eixo x
colmasc	Cor para o sexo masculino
colfem	Cor para o sexo feminino
colfxetar	Cor para os rótulos da faixa etária
border	Cor da borda das barras
inside	Ver a função barplot
drop.unused.levels	Apagar níveis não usados nos fatores?
...	argumentos de outras funções

Examples

```
## Not run:
data("POPBR12")
plot_pir(POPBR12, local = 'Brasil')
plot_pir(POPBR12[substr(POPBR12$MUNIC_RES, 1,2)==42, ], local='Santa Catarina')
plot_pir(POPBR12[substr(POPBR12$MUNIC_RES, 1,2)==43, ], local='Rio Grande do Sul')
plot_pir(POPBR12[POPBR12$MUNIC_RES==431490, ], local='Porto Alegre, RS')
plot_pir(POPBR12[POPBR12$MUNIC_RES==430520, ], local='Cerro Largo, RS')

## End(Not run)
```

POPBR12

População brasileira, 2012

Description

População por sexo e faixa etária. Municípios brasileiros, 2012.

Usage

POPBR12

Format

Banco de dados com estimativas populacionais por sexo e faixa etária para os municípios brasileiros em 2012. É o último arquivo com estimativas populacionais a esse nível de detalhamento publicado no repositório FTP do DATASUS. O banco é apresentado na forma em que foi baixado de <ftp.datasus.gov.br/dissemin/publicos/IBGE/POP/POPBR12.zip>.

POPRS2019

*População por sexo e faixa etária. Municípios gaúchos, 2019.***Description**

População por sexo e faixa etária. Municípios gaúchos, 2019.

Usage

POPRS2019

Format

Banco de dados com estimativas populacionais por sexo e faixa etária para os municípios do Rio Grande do Sul em 2019. Criado com a função `popbr2000_2021`, que por sua vez usa os dados tabulados do DATASUS e importados pelo pacote `brpop`, de Raphael Saldanha.

```
print.descreve
```

Imprime o resultado da função `descreve`

Description

Imprime o resultado da função `descreve`

Usage

```
## S3 method for class 'descreve'
print(x, print = "output", ...)
```

Arguments

<code>x</code>	Um objeto da classe <code>descreve</code> , output de <code>descreve</code>
<code>print</code>	Modo de apresentação; <code>print = "tabela"</code> retorna uma tabela com as estatísticas
<code>...</code>	Não sei se serve de algo mas parece que precisa

RDRS2019

*Arquivo da AIH - RS, 2019***Description**

Internações hospitalares pagas pelo SUS no RS em 2019.

Usage

RDRS2019

Format

Banco de dados com amostra aleatória de 10.000 internações dos "arquivos reduzidos" (RD*.DBC) das Autorizações de Internação Hospitalar (AIH) pagas pelo SUS em hospitais do RS em 2019 ("ano de competência" = 2019). O banco é formado pela junção (adição de linhas) de cada um dos arquivos RDRS1901.DBC, ..., RDRS1912.DBC.

Source

Bases de Dados do Sistema de Informações Hospitalares do SUS (BD-SIH/SUS)

sacamostras

*Toma múltiplas amostras aleatórias de um banco de dados.***Description**

Toma múltiplas amostras aleatórias sem reposição de um banco de dados. Cada amostra é salva em um objeto da classe 'data.frame' e exportada para um arquivo .csv, enquanto o conjunto dos objetos é salvo num arquivo de dados do R (.RData). Os arquivos são salvos no diretório de trabalho da sessão ativa. O n° de amostras a extrair é dado pelo comprimento de uma lista indicada no argumento 'suffixes' e cria novos "data frames" em um número definido de amostras aleatórias de tamanho n, criando com cada amostra um objeto da classe 'data.frame', que será salvo em um arquivo .csv e com os demais data frames, em um arquivo .RData. Criada com ajuda do chatGPT em 06/06/2023.

Usage

```
sacamostras(data, size, prefix, suffixes, tipo = 2, seed = NULL)
```

Arguments

data	"Data frame" fonte das amostras.
size	Tamanho das amostras (n° de registros).
prefix	Texto para o início do nome dos "data frames" e bancos de dados.
suffixes	Vetor numérico ou de caracteres, será usado como texto para o final do nome dos bancos de dados. Se for um vetor numérico e o argumento seed não for definido, será então usado para definir a "semente" das amostras (v. argumento seed).

tipo	Formatação do arquivo csv; 'tipo = 1' invoca a função <code>write.csv</code> , com campos separados por vírgula (',') e decimais separados por ponto ('.'); enquanto 'tipo = 2' (padrão) invoca a função <code>write.csv2</code> , criando arquivos com campos separados por ponto-e-vírgula (';') e decimais separados por vírgula (',').
seed	Vetor de números inteiros do tamanho do número de amostras desejado. É usado como "semente" para os números (pseudo)aleatórios que geram as amostras, permitindo assim sua reprodução. Se NULL (padrão), é tomado de suffixes, caso este seja um vetor de números inteiros. Argumento obrigatório quando suffixes não for numérico.

Examples

```
# Perceba que a função não é endereçada a nenhum objeto (como em `x <- sacamostras(...)`),
# uma vez que ela já cria os bancos de dados como objetos no espaço de trabalho.
semente = 1:3
# Amostras de 100 registros:
sacamostras(data = RDRS2019, size = 100, prefix = "amostra_", suffixes = semente)
# Amostras de 10% dos registros:
sacamostras(data = RDRS2019, size = .01, prefix = "amostra_", suffixes = semente)
# Amostras de 10% dos registros, com outra semente:
sacamostras(data = RDRS2019, size = .01, prefix = "s", suffixes = semente, seed = 11:13)

# Amostras de 10% dos registros, com outros nomes, mas mesma "semente":
sacamostras(data = RDRS2019, size = .01,
             prefix = "amostra", suffixes = paste0("0", 1:3),
             seed = semente)
all.equal(amostra01_1, amostra_1)

# Amostra de 10% dos registros, com mesmos nomes mas outra "semente":
sacamostras(data = RDRS2019, size = .01, prefix = "amostra", suffixes = semente, seed = 11:13)
all.equal(amostra1_11, s1_11)

# A função retorna um aviso de erro se o argumento 'suffixes' não tiver
# o mesmo comprimento do argumento 'seed':
## Not run:
sacamostras(data = RDRS2019, size = .01, prefix = "amostra_", suffixes = "bis", seed = semente)

## End(Not run)
# Amostra de 10% dos registros:
sacamostras(data = RDRS2019, size = .01, prefix = "amostra_",
             suffixes = rep("bis", length(semente)), seed = semente)
all.equal(amostra_bis_1, amostra_1)
sacamostras(data = RDRS2019, size = .01, prefix = "amostra", suffixes = semente, seed = 11:13)
all.equal(amostra1_11, s1_11)
rm(list = ls(pattern = "amostra"))
rm(list = ls(pattern = "s"))
unlink(c("amostra*", "s*")) # apaga os arquivos criados
```

Description

Constrói uma tabela com distribuição de frequências brutas, relativas e acumuladas, com rótulos em português

Usage

```
tabuleiro(
  x,
  digits = 1,
  total = TRUE,
  cum = TRUE,
  format = "en",
  data = NULL,
  ...
)
```

Arguments

x	O vetor a ser tabulado
digits	nº de decimais na tabela
total	TRUE (default) apresenta o total de categorias na tabela
cum	TRUE (default) apresenta as frequências acumuladas das cateogrias
format	Caráter indicando se o formato da tabela é anglo-saxão ("en", default) ou latino ("pt"); se "pt", os decimais serão apresentados depois de vírgula e os milhares separados por ponto # NÃO IMPLEMENTADO – COLOCAR ISSO NUM MÉTODO PRINT.tabuleiro
data	Optional argument. Data frame containing x. Defaults to NULL
...	permite o uso de argumentos da função table

Examples

```
set.seed(1)
x <- rbinom(100000, 3, .25)

tabuleiro(x)

# Sem o total
tabuleiro(x, total = FALSE)

# Sem as frequências acumuladas
tabuleiro(x, cum = FALSE)

# Oculta a frequência acumulada absoluta e mantém a % acumulada
tabuleiro(x, total = FALSE)[,-3]
```

tabuleiro2*Tabela de frequências com separadores latinos*

Description

Tabela de frequências com separadores latinos

Usage

```
tabuleiro2(varcat, digits = 1)
```

Arguments

varcat	Uma variável categórica
digits	No. de decimais

Examples

```
# criar uma variável politômica
x <- cut(rnorm(1000), 3) # sem missings
tabuleiro2(x)
x[1:100] <- NA # gerar missings
tabuleiro2(x)

## Not run:
knitr::kable(tabuleiro2(x), align = rep("r", 4))

## End(Not run)
```

zeroesq*Zeros à esquerda*

Description

Unifica o número de caracteres em um vetor, colocando zeros ("0") à esquerda do valor original.

Usage

```
zeroesq(x, n = NULL)
```

Arguments

x	Vetor (de caracteres ou numérico) com algarismos arábicos.
n	Limite do número de zeros a incluir; o padrão é 'NULL', sendo então definido pela função como o necessário até alcançar o número máximo de caracteres do vetor; pode ser definido pelo usuário.

Details

Muitas vezes temos, nos bancos de dados, uma variável registrada em algarismos com grande amplitude de valores mas que na verdade são códigos em vez de valores numéricos, como no caso do número de identificação do registro, ou da Equipe de Saúde da Família em um grande município. Se a variável é numérica ('integer', 'numeric' ou 'double/float'), não é um problema grave, nesse caso costuma-se ignorar os resumos apresentados sobre a variável ou transformá-la em texto ('character'). Entretanto, em qualquer caso pode trazer alguns inconvenientes, como - se, num banco de dados com mais de mil registros uma variável "identificador" aparecer como 1, ..., em vez de 0001, ..., ela não pode estar inserida (sem trabalho prévio) em definições condicionais para múltiplas variáveis do tipo "em todos os fatores, transforme '9', '99' e '999' em NA", ou "em todas variáveis numéricas, transforme e '999' em NA"; - ordenação de numerais em formato de texto não segue as mesmas regras que em formato numérico (temos "1, 10, 11, ..., 19, 2, 20, 21, ...), o que implica na definição dos níveis ao definir um fator.

Value

Vetor de caracteres com os dígitos vazios à esquerda do valor original preenchidos por zero (0).

Examples

```
(codid <- c(1:3, 10:12, 101:103, 1000))

# Se o vetor é numérico a ordenação acontece sem problemas:
class(codid)
sort(codid)

# Mas se ele for de caracteres não:
sort(as.character(codid))

# o que gostaríamos é:
sort(as.character(zeroesq(codid)))

# ou, diretamente (porque o vetor pode ser numérico ou caractere):
sort(zeroesq(codid))
```

Index

- * **DATASUS**
 - fxetar.det_pra_fxetar5, [8](#)
- * **datasets**
 - obitosRS2019, [13](#)
 - POPBR12, [16](#)
 - POPRS2019, [17](#)
 - RDRS2019, [18](#)
- * **descreve**
 - print.descreve, [17](#)
- adissoma, [2](#)
- barplot, [16](#)
- bolero, [3](#)
- boxplot, [12](#)
- descreve, [4](#), [17](#)
- ed, [6](#)
- format, [7](#)
- formatC, [7](#)
- formatL, [7](#)
- fxetar.det_pra_fxetar5, [8](#)
- fxetarNM, [9](#)
- ggplot_pir, [10](#)
- hist, [5](#), [12](#), [13](#)
- histobox, [12](#)
- obitosRS2019, [13](#)
- par, [5](#)
- plot.histobox, [13](#)
- plot_pir, [15](#)
- plotZ, [14](#)
- POPBR12, [16](#)
- popbr2000_2021, [17](#)
- POPRS2019, [17](#)
- print.descreve, [17](#)
- RDRS2019, [18](#)
- sacamostras, [18](#)
- table, [20](#)
- tabuleiro, [19](#)
- tabuleiro2, [21](#)
- tail, [6](#)
- write.csv, [19](#)
- write.csv2, [19](#)
- zeroesq, [21](#)