RStudio

File Edit Code View Plots Session Project Build Tools Help

Go to file/function

Project: (None)

practical1.R

Source on Save | Run | Source

```
rm(list=ls())
search() #list of the packages loaded by default

library(sp)               #tools for spatial data
library(maptools)         #tools for reading and handling spatial objects
library(SpatialEpi)       #for the analysis of epidemiological data
library(RColorBrewer)     #color palettes
library(ggplot2)
library(R2OpenBUGS)
library(reshape)
#gpclibPermit()


#check the current working directoty
getwd()

#set a new working directory, i.e. the path where the files are
setwd("H:/SpatialEpidemiology_2013/practical1/")   #note / instead of \
#source("newfunction.R")

#import the health data
lung<-read.csv("Lungcancer_strata_GL.csv")

#print only the first rows of the data
head(lung)

#summarise the data
```

2:1 (Top Level) R Script

Console ~/

```
  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> rm(list=ls())
> search() #list of the packages loaded by default
 [1] ".GlobalEnv"        "tools:rstudio"     "package:stats"
 [4] "package:graphics"  "package:grDevices" "package:utils"
 [7] "package:datasets"  "package:methods"   "Autoloads"
[10] "package:base"
>
> library(sp)               #tools for spatial data
> library(maptools)
Loading required package: foreign
Loading required package: grid
Loading required package: lattice
Checking rgeos availability: TRUE
>
```

Workspace History

To Console | To Source

```
rm(list=ls())
search() #list of the packages loaded by default
library(sp) #tools for spatial data
library(maptools)
```

Files Plots Packages Help

R: Reshape Grouped Data | Find in Topic

reshape {stats} R Documentation

Reshape Grouped Data

Description

This function reshapes a data frame between 'wide' format with repeated measurements in separate columns of the same record and 'long' format with the repeated measurements in separate records.

Usage

```
reshape(data, varying = NULL, v.names = NULL, timevar = "time",
        idvar = "id", ids = 1:NROW(data),
        times = seq_along(varying[[1]]),
        drop = NULL, direction, new.row.names = NULL,
        sep = ".",
        split = if (sep == "") {
            list(regexp = "[A-Za-z][0-9]", include = TRUE)
        } else {
            list(regexp = sep, include = FALSE, fixed = TRUE)}
        )
```

Arguments

data	a data frame