Package ‘rSPARCS’

November 6, 2019

Type Package
Title Statistical Package for Analysis Related Cleaning Support
Version 0.0.7
Author Wangjian Zhang, Bo Ye, Zhicheng Du, Ziqiang Lin, Ji- 
jin Yao, Yanan Jin, Wayne R. Lawrence, Yuantao Hao
Maintainer Wangjian Zhang <wzhang27@albany.edu>
Description To clean and analyze hospital data, and generate sets for statistical modeling.
License GPL-3
Encoding UTF-8
LazyData true
Imports data.table,spatialEco,geosphere,tigris,raster,sp,plyr
NeedsCompilation no
Repository CRAN
Date/Publication 2019-11-06 14:40:06 UTC

R topics documented:

  case.series ................................................................. 2
  CXover.data ................................................................. 3
  DBFgeocode ................................................................. 4
  desc.comp ................................................................. 5
  dupl.readm ................................................................. 6
  FIPS.name ................................................................. 7
  mediationking .............................................................. 8
  pick.cases ................................................................. 9
  raster_extract ........................................................... 10

Index 12
**case.series**

*Generate the Case Series*

**Description**

Estimates the daily number of cases reported by multiple grouping factors.

**Usage**

```r
case.series(data, ICD, diagnosis, date, start, end, by1, by2, by3, by4, by5)
```

**Arguments**

- `data`: a data frame containing with each row representing a case, and each column representing the patient characteristics such as gender, age, admission date, and discharge date, etc.
- `ICD`: a vector of ICD 9, or 10 codes, or a mix of them, which users are willing to calculate the daily numbers for; can be of length 3-6.
- `diagnosis`: the name of the variable in the data containing the diagnostic code upon admission.
- `date`: the name of the variable in the data showing the admission date, either in the format like "20181129" or "2018/11/29".
- `start, end`: the start and end date for the case series to be generated.
- `by1, by2, by3, by4, by5`: the name of the variable in the data used as grouping variables.

**Details**

Not limited to hospital data, but also applicable to other surveillance data.

**Value**

A case series will be generated for time series analysis, trend analysis and displaying, with following variables:

- `date`: from the start date to the end date as user specified, with 1 day bin.
- `case`: the daily number of cases diagnosed with diseases of user specified ICD codes.
- `others`: grouping variables.

**Note**

When applied to other medical data without ICD code, users may arbitrarily set a ICD code, meanwhile, define the diagnosis variable in the data to the same ICD code.
Examples

# simulated data
set.seed(2018)

data=data.frame(
  patient=1:10000,
  primdiag=sample(390:398,10000,replace=TRUE),
  onset=sample(seq.Date(as.Date("2015/2/1"),
    as.Date("2016/2/1"),"1 day"),10000,replace=TRUE),
  sex=sample(c("M","F"),10000,replace=TRUE),
  county=sample(c("Albany","New York"),10000,replace=TRUE)
)

output.series=case.series(
  data,ICD=392:396,diagnosis="primdiag",
  date="onset",start="2015/1/1",end="2016/12/31",by1="sex")

head(output.series)

CXover.data  Generate the Dataset for Case Crossover Analysis

Description

Generate the dataset for case crossover analysis.

Usage

CXover.data(data,date,ID,direction)

Arguments

data a data.frame containing the date of each case.
date the name of the variable in the data indicating the date of each case reported to
the database.
ID the name of the variable in the data indicating case ID, if not specified, it will
automatically generated starting from 1.
direction "pre4"or"month4". With "pre4", each case day will be matched with same week-
days in previous 4 weeks. With "month4", each case day will be matched with
same weekdays in the same month.

Details

Not limited to hospital data, but also applicable to other surveillance data.
Value

dataset A data.frame ready for the case crossover analysis, with following variables:
ID same ID represents the same patient.
Date one case day is matched with 3-4 control days.
status indicating whether it is a case day or a control day.

Author(s)

Wang-Jian Zhang (wzhang27@albany.edu)

References


Examples

# similated data
s.set.4eed(2018)
dataset=data.frame(
    patient=1:1000,
    primdiag=sample(390:398,1000,replace=TRUE),
    onset=sample(seq.Date(as.Date("2015/2/1"),as.Date("2016/2/1"),"1 day"),1000,replace=TRUE),
    sex=sample(c("M","F"),1000,replace=TRUE),
    county=sample(c("Albany","New York"),1000,replace=TRUE))

out.data=CXover.data(data=dataset,date="onset",ID="patient",direction="pre4")
head(out.data)

DBFgeocode

Create a dbf File for Geocoding

Description

Generate address variables and output the data as a dbf file for geocoding in ArcGIS.

Usage

DBFgeocode(data,cityname,roadaddress,mailbox,ZIP)

Arguments

data A data.frame containing address variables that are necessary for geocoding.
cityname The name of the variable in the data indicating city or county names.
roadaddress The name of the variable in the data indicating home addresses.
mailbox Optional address information such as the number of mailbox and the number of floor.
ZIP The name of the variable in the data indicating ZIP codes.
Value

Users may output the function return to the computer as the dbf file using write.dbf().

Note

In the dbf file, a variable named "singleline" will be used in the second step of geocoding, while variables roadaddress, cityname and ZIP will be separately used in the first step, and the variable ZIP for the last step.

Examples

```r
# simulated data
datatest=data.frame(county=c("Albany","Albany","Albany"),
   address1=c("1 Lincoln ave","2 Lincoln ave","489 Washinton ave"),
   address2=c("1st floor","1st floor","2nd floor"),
   zip=12206)
DBFgeocode(data=datatest,cityname="county",roadaddress="address1",
   mailbox="address2",ZIP="zip")
```

---

desc.comp  

Generate a Descriptive Table

Description

Generate a comprehensive descriptive table with intergroup comparison.

Usage

```
desc.comp(data,variables,by,margin,avg.num,test.num)
```

Arguments

data: a data.frame containing the variables to be described and a group variable
variables: a numeric variable indicating the columns of variables to be described.
by: a number indicating the column of the group variable
margin: calculate the proportion for categorical variables by 1 (row) or 2 (column).
avg.num: "mean", describe continuous variables with mean and standard deviation; "median", describe continuous variables with median and interquantile range; otherwise, normal distribution test will be conducted, for normal distributed variables, "mean" will be used, otherwise, "median" will be used.
test.num: "metric", t test or anova will be used for intergroup comparison; "nonmetric", Wilcoxon rank sum test or Kruskal-Wallis test will be used; otherwise, normal distribution test will be conducted, for normal distributed variables, "metric" will be used, otherwise, "nonmetric" will be used.
**dupl.readm**

**Details**
Not limited to hospital data, but also applicable to other surveillance data.

**Value**
A comprehensive descriptive table with statistics and P value for intergroup comparisons.

**Author(s)**
Wang-Jian Zhang (wzhang27@albany.edu)

**Examples**
```
desc.comp(CO2,variables=2:5,by=1,margin=1)
```

---

**dupl.readm**
*Identify Duplicates and Re-admissions*

**Description**
Identify the duplicates and re-admissions in hospital data with subject identifications.

**Usage**
```
dupl.readm(data,UniqueID,date,period)
```

**Arguments**
- `data` a data.frame containing "UniqueID" and "date"
- `UniqueID` the name of the variable in the data indicating case ID.
- `date` the name of the variable in the data indicating the admission/onset date.
- `period` the time period used to define an re-admission; period=365 by default.

**Details**
Not limited to hospital data, but also applicable to other surveillance data with "UniqueID" and "date".

**Value**
- `id.dupl` indicating whether it is a duplicated record with exactly the same "UniqueID" and "date" as a previous record. In some hospital data, some patients may be reported twice or even more due to insurance issues. For most studies, researchers may remove this kind of duplicates to avoid potential overcounting problems.
- `onlyone` indicating whether this is the only record with this ID.
Period
the time period between the current visit and the previous one for a patient; 0
for the 1st visit; and NA for those with only one record.

Nadmission
indicating the times of admission, e.g. 1st, 2nd admission; a patient may have
more than one 1st admissions if some periods between two visits are greater
than e.g. 365 days.

Author(s)
Wang-Jian Zhang (wzhang27@albany.edu)

Examples
```r
dataset=data.frame(
  ID=c(1,3,4,2,4,6,3,5,7,1),
  onset=c("2015/1/1","2016/1/2","2015/5/9",
         "2015/12/1","2016/8/2","2015/5/9",
         "2015/11/1","2016/3/2","2016/5/9","2015/9/9")
)

out.data=dupl.readm(data=dataset,
                     UniqueID="ID",date="onset",period=365)
head(out.data)
```

FIPS.name

Determine the Area that Each Record Is Located in

Description
Identify the residential county/city/census tract for each case, and add county/city/census tract ID.

Usage
FIPS.name(data,ID.case,long.case,lat.case,map,state.map,level.map,areaID)

Arguments
data  A data.frame containing the ID and coordinates of cases
ID.case  Name of the variable in the data indicating the case ID.
long.case  Name of the variable in the data indicating the longitude of cases.
lat.case  Name of the variable in the data indicating the latitude of cases.
map  The reference map containing the boundary of county/city/census tract. Do not
     have to specify for study areas within the U.S. A map for a region outside
     the U.S. can be imported as a "spatialpolygonsdataframe" object.
state.map  State FIPS code for the study area, e.g. "36" for the New York State. Ignored if
           readers’ own map is being used.
level.map  "county" or "tract", determine whether cases will be matched to counties or
           census tracts. Ignored if readers’ own map is being used.
areaID  Name of the variable in the map indicating the area ID. Use the default if the
        study is within the U.S.
Details

Not limited to hospital data, but also applicable to other surveillance data.

Value

areaID  The area unique ID such as FIPS code and ZIP code will be added to the original data.

Author(s)

Wang-Jian Zhang (wzhang27@albany.edu)

Examples

```r
set.seed(2018)
dataset=data.frame(Patient=1:10,lat=rnorm(10,42,0.5),long=rnorm(10,-76,1))
#data.out=FIPS.name(data=dataset,ID.case="Patient",long.case="long",lat.case="lat",
#state.map="36",level.map="tract",areaID="GEOID")
#head(data.out)
```

### mediationking

**Mediating Analysis**

**Description**

This function provides convenient algorithm to calculate total effect, mediation effect, direct effect and the proportion of mediation effect.

**Usage**

```r
mediationking(dataset,outcome,mediator,exposure,n.sim)
```

**Arguments**

- `dataset`  The dataset that is used for analysis.
- `outcome`  The name of the outcome variable in the dataset.
- `mediator`  The name of the mediator in the dataset.
- `exposure`  The name of the exposure factor in the dataset.
- `n.sim`  Times of simulation to estimate 95% confidence intervals.

**Details**

Please use `set.seed()` if you want to get a consistent result; this function will be expended to allow more covariates shortly.
Value

**Total effect**  The total effect of the exposure on the outcome variable.

**Indirect effect**  The effect of the exposure on the outcome variable that is caused by mediator.

**Direct effect**  The effect of the exposure on the outcome variable that is caused by factors other than the mediator.

**Meditation.proportion**  The proportion of the mediation effect.

Author(s)

Bo Ye (bye2@albany.edu)

Examples

```r
exposure<-rnorm(1000,0,1)
mediator<-rnorm(1000,10,3)
outcome<-rnorm(1000,10,4)
dataset<-data.frame(outcome,mediator,exposure)
#mediationking(dataset,"outcome","mediator","exposure")
```

---

**pick.cases**  Select cases within certain distance around a site

Description

Identify the closest site (e.g. monitoring sites) for each case, and select cases within certain distance around a site, e.g. 15 miles buffer.

Usage

```r
pick.cases(data,long.case,lat.case,long.sites,lat.sites,radius)
```

Arguments

- **data**  a data.frame containing the coordinates of cases.
- **long.case**  the name of variable in the data indicating the longitude of cases.
- **lat.case**  the name of variable in the data indicating the latitude of cases.
- **long.sites**  a numeric vector containing the longitude of sites.
- **lat.sites**  a numeric vector containing the latitude of sites.
- **radius**  radius of the buffer, e.g. "15 miles", "20 kms".

Details

Not limited to hospital data, but also applicable to other surveillance data.
Value

which.site  the closest site to the case.
minDIST     the distance of the case to the closest site; in the same unit as "radius".
Select      an indicator of whether a case was within the buffer.

Author(s)

Wang-Jian Zhang (wzhang27@albany.edu)

References


Examples

```r
set.seed(2018)
data=data.frame(Patient=1:100,lat=rnorm(100,41,0.5),long=rnorm(100,-76,1))

long.monitor=c(-73.75464,-78.80953,-73.902,-73.82153,-77.54817)
lat.monitor=c(42.64225,42.87691,40.81618,40.73614,43.14618)
data.out=pick.cases(data,long.case="long",lat.case="lat",
long.sites=long.monitor,lat.sites=lat.monitor,radius="30 miles")
data.out
```

---

### raster_extract

**Extract Values from a Raster Map**

**Description**

Crop the raster with the boundary of areas of your interest, and extract the values from the raster to each of these areas.

**Usage**

```
raster_extract(rastermap,refmap,ID.var,ID.code,cutpoint)
```

**Arguments**

- **rastermap**: a raster map containing the information you need, such as the National Land Cover Database 2011.
- **refmap**: "SpatialPolygonsDataFrame" object. A reference map containing the boundary information of your study areas.
- **ID.var**: the name of variable in the refmap indicating the unique ID for each of your study areas.
**raster_extract**

ID.code a character vector containing the unique ID for areas that you want to extract the values to. ID.code=ALL” by default where all areas in the reference map are of interest.

cutpoint a number to dichotomize the values in the raster; specified ONLY when those values are continuous.

**Details**

Usually for extracting data which are available as rasters such as the land coverage or land usage data.

**Value**

ID.code the column indicating the unique ID for each area, followed by the number of cells for each category/colour within that area.

Total cells the total number of cells within each area.

**Author(s)**

Wang-Jian Zhang (wzhang27@albany.edu)

**Examples**

```r
#library(raster)
#set.seed(4715)
#rast=raster(matrix(rnorm(500),100,100))
#extent(rast)=c(50,100,10,60)
#crs(rast)=CRS("+proj=longlat +datum=WGS84")

#ref=cbind(x=c(60,80,80,70), y=c(20,25,40,30))
#p=Polygon(ref)
#ps=Polygons(list(p),ID="ID")
#ref= SpatialPolygons(list(ps))
#data=data.frame(value=1, ID="10086",row.names="ID")
#ref= SpatialPolygonsDataFrame(ref,data)
#proj4string(ref)=CRS("+proj=longlat +datum=WGS84")

#raster_extract(rastermap=rast,refmap=SPDF,ID.var="ID",ID.code="ALL",cutpoint=0.5)
```
Index

case.series, 2
CXover.data, 3

DBFgeocode, 4
desc.comp, 5
dupl.readm, 6

FIPS.name, 7
mediationking, 8
pick.cases, 9
raster_extract, 10