Package 'STrollR'

March 25, 2018

Title STrollR

Version 0.1.3

Description

A Library for Correcting Standard Errors for Computing Spatial and Temporal Correlation post-estimation. A computationally efficient way to calculate covariance matrices that are corrected for spatial and temporal correlation using a method I call *rolling*. Huge spatiotemporal covariance matrices can be calculated using sparse matrix approaches with spam and spam64. To calculate large sparse spatial weights matrices, use spam::rdist. See my website <https://sites.google.com/a/g.clemson.edu/ja-resources>. or github <https://github.com/Jadamso>.

Depends R (>= 3.4.1)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Published 2018-03-25

Date 2018-03-25

URL https://sites.google.com/a/g.clemson.edu/ja-resources

RoxygenNote 6.0.1

Imports Matrix,
spam,
spam64,
parallel,
data.table,
lfe,
spdep,
sphet

R topics documented:

`bartlettSparse` *Weighting Kernel*

Description

Weighting Kernel

Usage

```
bartlettSparse(d, dmax)
```

Arguments

`d`	returned from spdep::listw
`dmax`	name of file

Value

bartlett weight

df2stack — *Convert list of dataframes with to rasterstack*

Description

Convert list of dataframes with to rasterstack

Usage

```
df2stack(sim_i, DF)
```

Arguments

sim_i	which simulation
DF	list of dataframe

Value

rasterstack

DynPlot — *Create Gif Plots*

Description

Create Gif Plots

Usage

```
DynPlot(DFlist, ti, pname = "STvarX", ind = 1)

DynGif(pname, vw = FALSE)

mkGif(DFlist, ti, pname = "STvarX", ind = 1, vw = FALSE)
```

Arguments

DFlist	
ti	number of time periods
pname	name of file
ind	which simulation
vw	view output

Value

list of rasterstacks

Fac2Num *Converts Factor to Number*

Description

Converts Factor to Number

Usage

```
Fac2Num(x)
```

Arguments

x	numeric factor

fake_data_traditional *Create Space Time Lattice Data*

Description

Create Space Time Lattice Data

Usage

```
fake_data_traditional(n = 10, tf = 5, theta = c(5, 1, 1, 1))
```

Arguments

n	spatial dimension
tf	temporal dimension
theta	parameter vector for RHS

Value

dataframe with (n1,n2,t) coordinates and variables Y,X,X, Country Time

is.lattice *checks if data table has lattice structure ?*

Description

checks if data table has lattice structure ?

Usage

```
is.lattice(DAT, TIME, ID)
```

Arguments

DAT	a data.table
TIME	name of temporal column
ID	name of cellular ID variable

iterateObsJSpatial *Wrapper for Matrix Calculation for Space*

Description

Wrapper for Matrix Calculation for Space

Usage

```
iterateObsJSpatial(sub_dat, Xvars, wmat, verbose = TRUE, checkSym = FALSE,
  XOmegaX = XOmegaX0, ...)
```

Arguments

sub_dat	a dataframe object for 1 time period
Xvars	RHS design matrix
wmat	Weights Matrix
verbose	print output
checkSym	check if wmat is symmetric
XOmegaX	meat function

Value

object to be used in vcov* functions

iterateObsJTemporal *Wrapper for Matrix Calculation for Time*

Description

Wrapper for Matrix Calculation for Time

Usage

```
iterateObsJTemporal(sub_dat, Xvars, wmat, verbose = TRUE, checkSym = FALSE,
  XOmegaX = XOmegaX0, ...)
```

Arguments

sub_dat	a data frame object for one cell unit
Xvars	RHS design matrix
wmat	Weights Matrix
verbose	print output
checkSym	check if wmat is symmetric
XOmegaX	meat function

Value

object to be used in vcov* functions

KNN *K nearest neighbours Calculate the number of neighbours within a neighbourhood.*

Description

K nearest neighbours Calculate the number of neighbours within a neighbourhood.

Usage

```
KNN(w, h = w, type = "Moore")
```

Arguments

w	number of neighbours wide (east-west).
h	number of neighbours long (north-south).
type	type of neighbourhood; "Moore" or "VonNeumann"

Value

the number of nearest neighbours

Examples

```
KNN( 4 )
```

listj *Data Matrix Preparations*

Description

Data Matrix Preparations

Usage

```
listj(ddff, wmat = NA, tmat = NA, t_cutoff = 4, rho_t = NA,
  d_cutoff = 1, rho_sp = NA, latlon = NA, convert_to_angles = TRUE)
```

Arguments

ddff	formatted data.table from vcovST.format
wmat	spatial weights matrix
tmat	temporal weights matrix
t_cutoff	temporal cutoff
rho_t	unsupported vonneumann structure
d_cutoff	distance cutoff
rho_sp	unsupported vonneumann structure
latlon	coordinates in lon,lat or x,y
convert_to_angles	lon,lat to x,y?

Value

list object to be passed to vcov* functions

mfxall *Run multiple regressions on the same dataset*

Description

Run multiple regressions on the same dataset

Usage

```
mfxall(FORMS, ..., mfx_fun = mfxi, mc.cores = as.numeric(system("nproc",
  intern = TRUE)), parallel = TRUE)
```

Arguments

FORMS	list of regression formula
...	args passed to mfx_fun
mfx_fun	what type of regression
mc.cores	number of cores if parallel=TRUE
parallel	use parallel processing?

Value

List of Regressions Summaries

`mfxi` *Run a Regression*

Description

Run a Regression

Usage

```
mfxi(formi, datai, scl = TRUE, wmat0 = get("WMAT0"), unit_id = "ID",
  time_id = "Year", coord_id = c("x", "y"))
```

Arguments

`formi`	regression formula
`datai`	data for regression
`scl`	vcov correct for spatial+temporal covariance?
`wmat0`	spatial weights matrix passed to vcovSCL
`unit_id, time_id, coord_id`	passed to vcovSCL

Value

summary table

`NEIGH` *Calculate the weights objects used in spdep sphet*

Description

Calculate the weights objects used in spdep sphet

Usage

```
NEIGH(coord_sp, neigh = 1, knn = TRUE, adj = FALSE, dnn = FALSE,
  rast = FALSE, vario = FALSE, sphet = FALSE, tracer = TRUE,
  tr_type = "mult", tr_m = 20, tr_p = 16, symm = TRUE,
  symm_check = TRUE, SAVE = NA)
```

Arguments

coord_sp	matrix of coordinates or a SpatialPoints object
neigh	number of neighbours to use in calculation
knn	calculate weights using knn approach
adj	calculate vonneumann weights (see VonNeumann)
dnn	dnn approach unsupported
rast	raster approach unsupported
vario	is coord_sp a weights matrix?
sphet	create objects used in sphet?
tracer	create trace matrix objects?
tr_type	type of trace matrix
tr_m	trace matrix m
tr_p	trace matrix p
symm	make weights symmetric
symm_check	check for symmetric weights matrix
SAVE	filename to save to, NA <default> returns as object

Value

filename of saved objects, or returns objects if SAVE=NA

---

sim2stack *Convert simulation to rasterstack*

Description

Convert simulation to rasterstack

Usage

```
sim2stack(e_spt, nsim, xyt)
```

Arguments

e_spt	matrix of draws from spam.mvtnorm
xyt	lattice structure
number	of simulations

Value

list of rasterstacks

spt_cleanup *cleanup spam::mvtnorm*

Description

cleanup spam::mvtnorm

Usage

```
spt_cleanup(m, nsim, xyt)
```

Arguments

m	matrix of simulations (each row a realization)
nsim	number of simulations
xyt	lattice coordinates

Value

dataframe

var2stack *Convert Dataframe with 1 variable to raster for one realization*

Description

Convert Dataframe with 1 variable to raster for one realization

Usage

```
var2stack(df_i, sim_i)
```

Arguments

df_i	dataframe
sim_i	which simulation

Value

raster

varioJ *Variogram Calculation*

Description

Variogram Calculation

Usage

```
varioJ(coords, cutt, residu, latlon = FALSE, indices = FALSE,
  clean = FALSE)
```

Arguments

coords	coordinate matrix
cutt	cutoff from which to calculate variogram
residu	vector of values (i.e. OLS residuals) associated coords
latlon	coordinates are lon,lat or x,y
indices	return indices?
clean	unused currently

Value

data.frame of dij and (ei-ej)^2

vcovSpace.loop *vcovSpace with Parallel Approach*

Description

vcovSpace with Parallel Approach

Usage

```
vcovSpace.loop(DAT, LISTJ = NA, wmat = LISTJ$wmat, verbose = FALSE,
  cores = 4)
```

Arguments

DAT	list of regression objects from vcov.format
LISTJ	list of weighting objects from listj
wmat	weights matrix
verbose	show messages
cores	number of cores in spatial loop

Value

covariance matrix w/ spatial correction

vcovSpace.single *vcovSpace, Single Year Only*

Description

vcovSpace, Single Year Only

Usage

```
vcovSpace.single(DAT, LISTJ = NA, wmat = LISTJ$wmat, verbose = FALSE)
```

Arguments

DAT	list of regression objects from vcov.format
LISTJ	list of weighting objects from listj
wmat	weights matrix
verbose	show messages

Value

covariance matrix w/ spatial correction

vcovST *calculate vcovSTsep for felm object*

Description

calculate vcovSTsep for felm object

vcovST for seperable covariances using split approach

Usage

```
vcovST(vcovfun = vcovSTsep, reg, DF, unit, time, sp_coords, t_cutoff = NA,
  d_cutoff = NA, wmat = NA, tmat = NA, latlon = TRUE,
  convert_to_angles = TRUE, verbose = FALSE, SPLIT_sp = FALSE,
  SPLIT_t = FALSE, SPLIT = FALSE)

vcovST.loop(vcovfun = vcovSpace.loop, reg, DF, unit, time, sp_coords,
  t_cutoff = NA, d_cutoff = NA, wmat = NA, tmat = NA, latlon = TRUE,
  convert_to_angles = TRUE, verbose = FALSE, cores = 4, vcvtime = TRUE)
```

Arguments

vcovfun	function for which type of SHAC correction
reg	an lm or felm object
DF	data.frame with unit, time, sp_coords
unit	cellular id
time	time id
sp_coords	coordinate id
t_cutoff	cutoff for considering time correlation
d_cutoff	cutoff for considering space correlation
wmat	spatial weights matrix
tmat	temporal weights matrix
latlon	are coordinates in lon,lat or x,y form
convert_to_angles	convert lon,lat to x,y
verbose	show messages
SPLIT_sp, SPLIT_t, SPLIT	see vcovST.format
cores	number of cores to use
vcvtime	add temporal clustering

Details

if vcvtime==FALSE, wmat should have diagonal elements

Value

covariance matrix w/ SHAC correction

Functions

- vcovST.loop:

---

vcovST.format *Format Regression Output*

Description

Format Regression Output

Usage

```
vcovST.format(reg, unit, time, sp_coords, DF = NA, SPLIT_sp = FALSE,
  SPLIT_t = FALSE, SPLIT = FALSE)
```

Arguments

reg	felm object to be shaped
unit	string name for cellular_id variable
time	string name for time variable
sp_coords	string name for coordinate variables
DF	data.table to format, <NULL> formatts
SPLIT_sp	return list dataframe of DF for each cell
SPLIT_t	return list dataframe of DF for each time
SPLIT	SPLIT_sp=SPLIT_sp=TRUE

Value

object to be used in vcov* functions

---

vcovSTsep *vcovSTsep*

Description

vcovSTsep

Usage

```
vcovSTsep(DAT, LISTJ = NA, t_double_count = FALSE, verbose = FALSE,
  return_each = FALSE)
```

Arguments

DAT	list of regression objects from vcov.format
LISTJ	list of weighting objects from listj
t_double_count	double count time?
verbose	print output
return_each	for debugging, return only spatial and only temporal covariances

Value

covariance matrix w/ SHAC correction

vcovSTsep.loop *vcovSTsep with Parallel for Space and Time*

Description

vcovSTsep with Parallel for Space and Time

Usage

```
vcovSTsep.loop(DAT, LISTJ = NA, wmat = LISTJ$wmat, t_double_count = FALSE,
  verbose = FALSE, return_each = FALSE, cores = 4, tcores = NA)
```

Arguments

DAT	list of regression objects from vcov.format
LISTJ	list of weighting objects from listj
wmat	weights matrix
t_double_count	double count time?
verbose	show messages
return_each	for debugging, return only spatial and only temporal covariances
cores	number of cores in spatial loop
tcores	number of cores in temporal loop

Value

covariance matrix w/ SHAC correction

VonNeumann *Compute VonNeumann Neighbours*

Description

Compute VonNeumann Neighbours

Usage

```
VonNeumann(coord_sp)
```

Arguments

coord_sp	SpatialPoints object, matrix of coordinates

Value

sparse weights matrix

weight_mat *Compute Sparse Spatial Weights Matrix*

Description

Compute Sparse Spatial Weights Matrix

Usage

```
weight_mat(M, cutoff, latlon = NA, convert_to_angles = TRUE)
```

Arguments

M	matrix of coordinates
cutoff	use distances up cutoff
latlon	are the rows (lat,lon) coordinates?
convert_to_angles	convert cutoff from km to angles?

Value

the number of nearest neighbours

Examples

```
weight_mat(expand.grid( list(x=1:10, y=1:10)), cutoff=.5)
```

write.listw2gwt *Write a listw object as a GWT file*

Description

Write a listw object as a GWT file

Usage

```
write.listw2gwt(listw, dgwt_outfile = paste0(tempdir(), "dgwt.GWT"))
```

Arguments

listw	returned from spdep::listw
dgwt_outfile	name of file

Value

dgwt_outfile

XOmegaX0 *Main 'Meat' Matrix Calculation*

Description

Main 'Meat' Matrix Calculation

Usage

```
XOmegaX0(X, WMAT, e)
```

Arguments

X	design matrix
WMAT	weighting matrix (preferably sparse sparse)
e	vector of residuals

Value

object to be used in vcov* functions

Index