

# Package ‘SpatialEpi’

February 14, 2012

**Type** Package

**Title** Performs various spatial epidemiological analyses

**Version** 0.1

**Date** 2010-01-17

**Depends** MASS, sp, maps, maptools, shapefiles

**Author** Albert Y. Kim <albert@stat.washington.edu>

**Maintainer** Albert Y. Kim <albert@stat.washington.edu>

**Description** Cluster Detection, Disease Mapping

**License** GPL-2

**LazyLoad** yes

**Repository** CRAN

**Date/Publication** 2012-01-22 20:17:17

## R topics documented:

SpatialEpi-package . . . . .	2
besag.newell . . . . .	2
eBayes . . . . .	5
expected . . . . .	6
GammaPriorCh . . . . .	7
kulldorff . . . . .	8
latlong2grid . . . . .	10
LogNormalPriorCh . . . . .	11
mapvariable . . . . .	12
pennLC . . . . .	13
polygon2spatial.polygon . . . . .	14
scotland . . . . .	16
zones . . . . .	17

<b>Index</b>	<b>19</b>
--------------	-----------

---

SpatialEpi-package      *Methods for spatial epidemiology*

---

### Description

This package contains methods for cluster detection and disease mapping, as well as plotting methods using the [sp](#) package.

### Details

Package:      SpatialEpi  
Type:          Package  
Version:       1.0  
Date:          2009-01-20  
License:       GPL  
LazyLoad:     yes

### Author(s)

Albert Y. Kim      <albert@stat.washington.edu>  
Michelle E. Ross   <micher3@u.washington.edu>  
Jon Wakefield      <jonno@stat.washington.edu>

### See Also

[sp](#)

---

besag.newell      *Besag-Newell Cluster Detection Method*

---

### Description

Besag-Newell cluster detection method. There are differences with the original paper and our implementation:

- we base our analysis on  $k$  cases, rather than  $k$  *other* cases as prescribed in the paper.
- we do not subtract 1 from the *accumulated numbers of other cases* and *accumulated numbers of others at risk*, as was prescribed in the paper to discount selection bias

- $M$  is the total number of areas included, not the number of additional areas included. i.e.  $M$  starts at 1, not 0.
- $p$ -values are not based on the original value of  $k$ , rather the actual number of cases observed until we view  $k$  or more cases. Ex: if  $k = 10$ , but as we consider neighbors we encounter 1, 2, 9 then 12 cases, we base our  $p$ -values on  $k = 12$
- we do not provide a Monte-Carlo simulated  $R$ : the number of tests that attain significance at a fixed level  $\alpha$

The first two and last differences are because we view the testing on an area-by-area level, rather than a case-by-case level.

### Usage

```
besag.newell(geo, population, cases, expected.cases=NULL, k, alpha.level)
```

### Arguments

geo	an $n \times 2$ table of the (x,y)-coordinates of the area centroids
cases	aggregated case counts for all $n$ areas
population	aggregated population counts for all $n$ areas
expected.cases	expected numbers of disease for all $n$ areas
k	number of cases to consider
alpha.level	$\alpha$ -level threshold used to declare significance

### Details

For the population and cases tables, the rows are bunched by areas first, and then for each area, the counts for each strata are listed. It is important that the tables are balanced: the strata information are in the same order for each area, and counts for each area/strata combination appear exactly once (even if zero).

### Value

List containing

clusters	information on all clusters that are $\alpha$ -level significant, in decreasing order of the $p$ -value
p.values	for each of the $n$ areas, $p$ -values of each cluster of size at least $k$
m.values	for each of the $n$ areas, the number of areas need to observe at least $k$ cases
observed.k.values	based on m.values, the actual number of cases used to compute the $p$ -values

### Note

The clusters list elements are themselves lists reporting:

location.IDs.included	ID's of areas in cluster, in order of distance
population	population of cluster
number.of.cases	number of cases in cluster
expected.cases	expected number of cases in cluster
SMR	estimated SMR of cluster
p.value	<i>p</i> -value

**Author(s)**

Albert Y. Kim

**References**

Besag J. and Newell J. (1991) The Detection of Clusters in Rare Diseases *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, **154**, 143–155

**See Also**

[pennLC](#), [expected](#)

**Examples**

```
## Load Pennsylvania Lung Cancer Data
data(pennLC)
data <- pennLC$data

## Process geographical information and convert to grid
geo <- pennLC$geo[,2:3]
geo <- latlong2grid(geo)

## Get aggregated counts of population and cases for each county
population <- tapply(data$population,data$county,sum)
cases <- tapply(data$cases,data$county,sum)

## Based on the 16 strata levels, computed expected numbers of disease
n.strata <- 16
expected.cases <- expected(data$population, data$cases, n.strata)

## Set Parameters
k <- 1250
alpha.level <- 0.05

# not controlling for stratas
results <- besag.newell(geo, population, cases, expected.cases=NULL, k, alpha.level)

# controlling for stratas
results <- besag.newell(geo, population, cases, expected.cases, k, alpha.level)
```

---

`eBayes`*Empirical Bayes Estimates of Relative Risk*

---

**Description**

The computes empirical Bayes estimates of relative risk of study region with  $n$  areas, given observed and expected numbers of counts of disease and covariate information.

**Usage**

```
eBayes(Y, E, Xmat = NULL)
```

**Arguments**

<code>Y</code>	a length $n$ vector of observed cases
<code>E</code>	a length $n$ vector of expected number of cases
<code>Xmat</code>	$n \times p$ dimension matrix of covariates

**Value**

A list with 5 elements:

<code>RR</code>	the ecological relative risk posterior mean estimates
<code>RRmed</code>	the ecological relative risk posterior mean estimates
<code>beta</code>	the MLE's of the regression coefficients
<code>alpha</code>	the MLE of negative binomial dispersion parameter
<code>SMR</code>	the standardized mortality/morbidity ratio $Y/E$

**References**

Clayton D. and Kaldor J. (1987) Empirical Bayes estimates of age-standardized relative risks for use in disease mapping. *Biometrics*, **43**, 671–681

**See Also**

[scotland](#), [mapvariable](#)

**Examples**

```
data(scotland)
data <- scotland$data

x <- data$AFF
Xmat <- cbind(x,x^2)
results <- eBayes(data$cases,data$expected,Xmat)

scotland.map <- scotland$spatial.polygon
mapvariable(results$RR, scotland.map)
```

---

 expected
 

---



---

*Compute Expected Numbers of Disease*


---

**Description**

Compute the internally indirect standardized expected numbers of disease.

**Usage**

```
expected(population, cases, n.strata)
```

**Arguments**

population	a vector of population counts for each strata in each area
cases	a vector of the corresponding number of cases
n.strata	number of strata considered

**Details**

The population and cases vectors must be *balanced*: all counts are sorted by area first, and then within each area the counts for all strata are listed (even if 0 count) in the same order.

**Value**

expected.cases a vector of the expected numbers of disease for each area

**Author(s)**

Albert Y. Kim

**References**

Elliot, P. et al. (2000) *Spatial Epidemiology: Methods and Applications*. Oxford Medical Publications.

**Examples**

```
data(pennLC)
population <- pennLC$data$population
cases <- pennLC$data$cases

## In each county in Pennsylvania, there are 2 races, gender and 4 age bands considered =
## 16 strata levels
pennLC$data[1:16,]
expected(population,cases,16)
```

---

`GammaPriorCh`*Compute Parameters to Calibrate a Gamma Distribution*

---

**Description**

Compute parameters to calibrate the prior distribution of a relative risk that has a gamma distribution.

**Usage**

```
GammaPriorCh(theta, prob, d)
```

**Arguments**

<code>theta</code>	upper quantile
<code>prob</code>	upper quantile
<code>d</code>	degrees of freedom

**Value**

A list containing

<code>a</code>	shape parameter
<code>b</code>	rate parameter

**Author(s)**

Jon Wakefield

**See Also**

[LogNormalPriorCh](#)

**Examples**

```
param <- GammaPriorCh(5, 0.975, 1)
curve(dgamma(x, shape=param$a, rate=param$b), from=0, to=6, n=1000, ylab="density")
```

kulldorff

*Kulldorff Cluster Detection Method***Description**

Kulldorff spatial cluster detection method for a study region with  $n$  areas. The method constructs *zones* by consecutively aggregating nearest-neighboring areas until a proportion of the total study population is included. Given the observed number of cases, the likelihood of each zone is computed using either binomial or poisson likelihoods. The procedure reports the zone that is the *most likely cluster* and generates significance measures via Monte Carlo sampling. Further, *secondary clusters*, whose Monte Carlo p-values are below the  $\alpha$ -threshold, are reported as well.

**Usage**

```
kulldorff(geo, cases, population, expected.cases=NULL, pop.upper.bound, n.simulations, alpha.level, plot)
```

**Arguments**

geo	an $n \times 2$ table of the (x,y)-coordinates of the area centroids
cases	aggregated case counts for all $n$ areas
population	aggregated population counts for all $n$ areas
expected.cases	expected numbers of disease for all $n$ areas
pop.upper.bound	the upper bound on the proportion of the total population each zone can include
n.simulations	number of Monte Carlo samples used for significance measures
alpha.level	$\alpha$ -level threshold used to declare significance
plot	flag for whether to plot histogram of Monte Carlo samples of the log-likelihood of the most likely cluster

**Details**

If `expected.cases` is specified to be `NULL`, then the binomial likelihood is used. Otherwise, a Poisson model is assumed. Typical values of `n.simulations` are 99, 999, 9999...

**Value**

List containing:

most.likely.cluster	information on the most likely cluster
secondary.clusters	information on secondary clusters, if none <code>NULL</code> is returned
type	type of likelihood
log.lkhd	log-likelihood of each zone considered
simulated.log.lkhd	<code>n.simulations</code> Monte Carlo samples of the log-likelihood of the most likely cluster

**Note**

The `most.likely.cluster` and `secondary.clusters` list elements are themselves lists reporting:

<code>location.IDs.included</code>	ID's of areas in cluster, in order of distance
<code>population</code>	population of cluster
<code>number.of.cases</code>	number of cases in cluster
<code>expected.cases</code>	expected number of cases in cluster
<code>SMR</code>	estimated SMR of cluster
<code>log.likelihood.ratio</code>	log-likelihood of cluster
<code>monte.carlo.rank</code>	rank of lkhd of cluster within Monte Carlo simulated values
<code>p.value</code>	Monte Carlo $p$ -value

**Author(s)**

Albert Y. Kim

**References**

- SatScan: Software for the spatial, temporal, and space-time scan statistics <http://www.satscan.org/>
- Kulldorff, M. (1997) A spatial scan statistic. *Communications in Statistics: Theory and Methods*, **26**, 1481–1496.
- Kulldorff M. and Nagarwalla N. (1995) Spatial disease clusters: Detection and Inference. *Statistics in Medicine*, **14**, 799–810.

**See Also**

[pennLC](#), [expected](#)

**Examples**

```
## Load Pennsylvania Lung Cancer Data
data(pennLC)
data <- pennLC$data

## Process geographical information and convert to grid
geo <- pennLC$geo[,2:3]
geo <- latlong2grid(geo)

## Get aggregated counts of population and cases for each county
population <- tapply(data$population,data$county,sum)
cases <- tapply(data$cases,data$county,sum)

## Based on the 16 strata levels, computed expected numbers of disease
n.strata <- 16
expected.cases <- expected(data$population, data$cases, n.strata)
```

```
## Set Parameters
pop.upper.bound <- 0.5
n.simulations <- 999
alpha.level <- 0.05
plot <- TRUE

## Kulldorff using Binomial likelihoods
binomial <- kulldorff(geo, cases, population, NULL, pop.upper.bound, n.simulations, alpha.level, plot)
cluster <- binomial$most.likely.cluster$location.IDs.included

## plot
plot(pennLC$spatial.polygon, axes=TRUE)
plot(pennLC$spatial.polygon[cluster], add=TRUE, col="red")
title("Most Likely Cluster")

## Kulldorff using Poisson likelihoods
poisson <- kulldorff(geo, cases, population, expected.cases, pop.upper.bound, n.simulations, alpha.level, plot)
cluster <- poisson$most.likely.cluster$location.IDs.included

## plot
plot(pennLC$spatial.polygon, axes=TRUE)
plot(pennLC$spatial.polygon[cluster], add=TRUE, col="red")
title("Most Likely Cluster Controlling for Strata")
```

---

latlong2grid

*Convert Coordinates from Latitude/Longitude to Grid*

---

## Description

Convert geographic latitude/longitude coordinates to kilometer-based grid coordinates.

## Usage

```
latlong2grid(input)
```

## Arguments

**input** either an  $n \times 2$  matrix of longitude and latitude coordinates in decimal format or an object of class `SpatialPolygons` (See [SpatialPolygons-class](#))

## Details

Longitude/latitudes are not a grid-based coordinate system: latitudes are equidistant but the distance between longitudes varies.

**Value**

Either a data frame with the corresponding (x,y) kilometer-based grid coordinates, or a SpatialPolygons object with the coordinates changed.

**Note**

Rough conversion of US lat/long to km (used by GeoBUGS): (see also [forum.swarthmore.edu/dr.math/problems/longandlat.html](http://forum.swarthmore.edu/dr.math/problems/longandlat.html))  
 Radius of earth:  $r = 3963.34$  (equatorial) or  $3949.99$  (polar) mi =  $6378.2$  or  $6356.7$  km, which implies: km per mile =  $1.609299$  or  $1.609295$  a change of 1 degree of latitude corresponds to the same number of km, regardless of longitude.  $arclength=r*\theta$ , so the multiplier for coord\$y should probably be just the radius of earth. On the other hand, a change of 1 degree in longitude corresponds to a different distance, depending on latitude. (at N pole, the change is essentially 0. at the equator, use equatorial radius.

**Author(s)**

Lance A. Waller

**Examples**

```
## Convert coordinates
coord <- data.frame(rbind(
# Montreal, QC: Latitude: 45deg 28' 0" N (deg min sec), Longitude: 73deg 45' 0" W
c(-73.7500, 45.4667),
# Vancouver, BC: Latitude: 45deg 39' 38" N (deg min sec), Longitude: 122deg 36' 15" W
c(-122.6042, 45.6605)
))
latlong2grid(coord)

## Convert SpatialPolygon
data(pennLC)
new <- latlong2grid(pennLC$spatial.polygon)
par(mfrow=c(1,2))
plot(pennLC$spatial.polygon,axes=TRUE)
title("Lat/Long")
plot(new,axes=TRUE)
title("Grid (in km)")
```

**Description**

Compute parameters to calibrate the prior distribution of a relative risk that has a log-normal distribution.

**Usage**

```
LogNormalPriorCh(theta1, theta2, prob1, prob2)
```

**Arguments**

theta1	lower quantile
theta2	upper quantile
prob1	lower probability
prob2	upper probability

**Value**

A list containing:

mu	mean of log-normal distribution
sigma	variance of log-normal distribution

**Author(s)**

Jon Wakefield

**See Also**

[GammaPriorCh](#)

**Examples**

```
# Calibrate the log-normal distribution s.t. the 95% confidence interval is [0.2, 5]
param <- LogNormalPriorCh(0.2, 5, 0.025, 0.975)
curve(dlnorm(x,param$mu,param$sigma), from=0, to=6, ylab="density")
```

---

mapvariable

*Plot Levels of a Variable in a Colour-Coded Map*

---

**Description**

Plot levels of a variable in a colour-coded map along with a legend.

**Usage**

```
mapvariable(y, spatial.polygon, ncut=1000, nlevels=10, lower=NULL, upper=NULL, main=NULL, xlab=NULL, y
```

**Arguments**

y	variable to plot
spatial.polygon	an object of class SpatialPolygons (See <a href="#">SpatialPolygons-class</a> )
ncut	number of cuts in colour levels to plot
nlevels	number of levels to include in legend
lower	lower bound of levels

upper	upper bound of levels
main	an overall title for the plot
xlab	a title for the x axis
ylab	a title for the y axis

**Value**

A map colour-coded to indicate the different levels of y

**Author(s)**

Jon Wakefield, Nicky Best, Sebastien Haneuse, and Albert Y. Kim

**References**

Bivand, R. S., Pebesma E. J., and Gomez-Rubio V. (2008) *Applied Spatial Data Analysis with R*. Springer Series in Statistics.

E. J. Pebesma and R. S. Bivand. (2005) Classes and methods for spatial data in R. *R News*, **5**, 9–13.

**Examples**

```
data(scotland)
map <- scotland$spatial.polygon
y <- scotland$data$cases
E <- scotland$data$expected
SMR <- y/E
mapvariable(SMR, map, main="Scotland", xlab="Eastings (km)", ylab="Northings (km)")
```

---

pennLC

*Pennsylvania Lung Cancer*


---

**Description**

County-level (n=67) population/case data for lung cancer in Pennsylvania in 2002, stratified on race (white vs non-white), gender and age (Under 40, 40-59, 60-69 and 70+). Additionally, county-specific smoking rates.

**Usage**

```
data(pennLC)
```

**Format**

List of 3 items:

geo	a table of county IDs, longitude/latitude of the geographic centroid of each county
data	a table of county IDs, number of cases, population and strata information
smoking	a table of county IDs and proportion of smokers
spatial.polygon	an object of class SpatialPolygons (See <a href="#">SpatialPolygons-class</a> )

**Source**

Population data was obtained from the 2000 decennial census, lung cancer and smoking data were obtained from the Pennsylvania Department of Health website: <http://www.dsf.health.state.pa.us/>

**See Also**

[scotland](#)

**Examples**

```
data(pennLC)
pennLC$geo
pennLC$data
pennLC$smoking

# Map smoking rates in Pennsylvania
mapvariable(pennLC$smoking[,2], pennLC$spatial.polygon)
```

---

`polygon2spatial.polygon`

*Convert a Polygon to a Spatial Polygons Object*

---

**Description**

Converts a polygon (a matrix of coordinates with NA values to separate subpolygons) into a Spatial Polygons object.

**Usage**

```
polygon2spatial.polygon(poly, coordinate.system, area.names = NULL, nrepeats = NULL)
```

**Arguments**

<code>poly</code>	a 2-column matrix of coordinates, where each complete subpolygon is separated by NA's
<code>coordinate.system</code>	the coordinate system to use
<code>area.names</code>	names of all areas
<code>nrepeats</code>	number of subpolygons for each area

**Details**

Just as when plotting with the [polygon](#) function, it is assumed that each subpolygon is to be closed by joining the last point to the first point. In the matrix `poly`, NA values separate complete subpolygons.

`coordinate.system` must be either `'+proj=utm'` or `'+proj=longlat'`.

In the case with an area consists of more than one separate closed polygon, `nrepeats` specifies the number of closed polygons associated with each area.

**Value**

An object of class `SpatialPolygons` (See [SpatialPolygons-class](#) from the `sp` package).

**Author(s)**

Albert Y. Kim

**References**

Bivand, R. S., Pebesma E. J., and Gomez-Rubio V. (2008) *Applied Spatial Data Analysis with R*. Springer Series in Statistics.

E. J. Pebesma and R. S. Bivand. (2005) Classes and methods for spatial data in R. *R News*, **5**, 9–13.

**Examples**

```
data(scotland)

polygon <- scotland$polygon$polygon
coord.system <- '+proj=utm'
names <- scotland$data$county.names
nrepeats <- scotland$polygon$nrepeats

spatial.polygon <- polygon2spatial.polygon(polygon,coord.system,names,nrepeats)

par(mfrow=c(1,2))
# plot using polygon function
plot(polygon,type='n',xlab="Eastings (km)",ylab="Northings (km)",main="Polygon File")
polygon(polygon)

# plot as spatial polygon object
plot(spatial.polygon,axes=TRUE)
title(xlab="Eastings (km)",ylab="Northings (km)",main="Spatial Polygon")

# Note that area 23 (argyll-bute) consists of 8 separate polygons
nrepeats[23]
plot(spatial.polygon[23],add=TRUE,col="red")
```

---

`scotland`*Lip Cancer in Scotland*

---

**Description**

County-level (n=56) data for lip cancer among males in Scotland between 1975-1980

**Usage**

```
data(scotland)
```

**Format**

List containing:

<code>geo</code>	a table of county IDs, x-coordinates (eastings) and y-coordinates (northings) of the geographic centroid of each county
<code>data</code>	a table of county IDs, number of cases, population and strata information
<code>spatial.polygon</code>	a Spatial Polygons class (See <a href="#">SpatialPolygons-class</a> ) map of Scotland
<code>polygon</code>	a polygon map of Scotland (See <a href="#">polygon2spatial.polygon</a> )

**Source**

Kemp I., Boyle P., Smans M. and Muir C. (1985) Atlas of cancer in Scotland, 1975-1980, incidence and epidemiologic perspective *International Agency for Research on Cancer* **72**.

**References**

Clayton D. and Kaldor J. (1987) Empirical Bayes estimates of age-standardized relative risks for use in disease mapping. *Biometrics*, **43**, 671–681

**See Also**

[mapvariable](#), [polygon2spatial.polygon](#), [pennLC](#)

**Examples**

```
data(scotland)
data <- scotland$data
scotland.map <- scotland$spatial.polygon

SMR <- data$cases/data$expected
mapvariable(SMR,scotland.map)
```

---

zones *Create Set of All Zones and Output Geographical Information*

---

### Description

Based on the population counts and centroid coordinates of each area, output the set of zones as defined by Kulldorff and other geographical information.

### Usage

```
zones(geo, area.population, pop.upper.bound)
```

### Arguments

geo                    an  $n \times 2$  table of the (x,y)-coordinates of the area centroids  
 area.population      a vector of population counts of each area  
 pop.upper.bound      maximum proportion of study region each zone can contain

### Value

A list containing

- nearest.neighbors    a list of  $n$  elements, where each element is a vector of the nearest neighbors in order of distance up until pop.upper.bound of the total population is attained
- n.zones              total number of zones
- cluster.coords      an  $n \times 2$  table of the center and the radial areas for each zone
- dist                  an  $n \times n$  inter-point distance matrix

### Author(s)

Albert Y. Kim

### References

Kulldorff, M. (1997) A spatial scan statistic. *Communications in Statistics: Theory and Methods*, **26**, 1481–1496.  
 Kulldorff M. and Nagarwalla N. (1995) Spatial disease clusters: Detection and Inference. *Statistics in Medicine*, **14**, 799–810.

**Examples**

```
data(pennLC)
geo <- pennLC$geo[,2:3]
geo <- latlong2grid(geo)
area.population <- tapply(pennLC$data$population, pennLC$data$county, sum)
pop.upper.bound <- 0.5

geo.info <- zones(geo, area.population, pop.upper.bound)
```

# Index

## \*Topic **datasets**

pennLC, [13](#)  
scotland, [16](#)

## \*Topic **file**

besag.newell, [2](#)  
eBayes, [5](#)  
expected, [6](#)  
GammaPriorCh, [7](#)  
kulldorff, [8](#)  
latlong2grid, [10](#)  
LogNormalPriorCh, [11](#)  
mapvariable, [12](#)  
polygon2spatial.polygon, [14](#)  
zones, [17](#)

## \*Topic **package**

SpatialEpi-package, [2](#)

besag.newell, [2](#)

eBayes, [5](#)  
expected, [4](#), [6](#), [9](#)

GammaPriorCh, [7](#), [12](#)

kulldorff, [8](#)

latlong2grid, [10](#)  
LogNormalPriorCh, [7](#), [11](#)

mapvariable, [5](#), [12](#), [16](#)

pennLC, [4](#), [9](#), [13](#), [16](#)  
polygon, [15](#)  
polygon2spatial.polygon, [14](#), [16](#)

scotland, [5](#), [14](#), [16](#)

sp, [2](#)  
SpatialEpi (SpatialEpi-package), [2](#)  
SpatialEpi-package, [2](#)  
SpatialPolygons-class, [10](#), [12](#), [13](#), [15](#), [16](#)

zones, [17](#)