R Data and Methods for Spatial Epidemiology: the SpatialEpi Package

Cici Chen * Albert Y. Kim† Michelle Ross‡ Jon Wakefield§

August 12, 2010

Contents

*Dept of Statistics, University of Washington, Box 354322, Seattle WA, 98195-4322
†Dept of Statistics, University of Washington, Box 354322, Seattle WA, 98195-4322
‡Dept of Biostatistics, University of Washington, Box 357232, Seattle WA, 98195-7232
§Dept of Biostatistics, University of Washington, Box 357232, Seattle WA, 98195-7232
¶cicichen@u.washington.edu, albert@stat.washington.edu, micher3@u.washington.edu, jonno@u.washington.edu

1 Introduction

Spatial epidemiology is the description and analysis of geographically indexed health data with respect to demographic, environmental, behavioral, socioeconomic, genetic, and infectious risk factors [5]. Broadly speaking, the field of spatial epidemiology can be divided into three principal subfields: disease mapping, spatial regression/geographic correlation studies, and analysis of disease clusters. The **SpatialEpi** package implements methods for these subfields.

All the `R` code is this vignette can be extracted into a single script file via the `Stangle` command:

```
> library(SpatialEpi)
> Rcode <- system.file("doc", "SpatialEpi.Rnw", package = "SpatialEpi")
> options(device.ask.default = FALSE)
> Stangle(Rcode)

Writing to file SpatialEpi.R
```

2 Producing Maps

The production of disease atlases is one of the chief tasks in spatial epidemiology. In order to facilitate producing such maps, the **SpatialEpi** package uses the **sp** package to process objects of class `SpatialPolygons` [11]. Further information on the **sp** package can be found in *Applied Spatial Data Analysis with R* [3].

2.1 Converting Different Map Formats into SpatialPolygons

Several different formats of maps can be converted into objects of class `SpatialPolygons`.

2.1.1 Converting Polygons to SpatialPolygons

A polygon file consists of a 2-column matrix of coordinates, where each complete subpolygon representing some subarea of the study region (counties, census tracts, zip/postal codes) is separated by `NA` values. All subpolygons are assumed to be closed by joining the last point to the first point.

Using the `polygon2spatial.polygon()` function, we can convert the polygon file into an object of class `SpatialPolygons`. In the case when certain subareas consist of more than one contiguous land mass, we specify the `nrepeats` vector where each element represents the number of subpolygons corresponding to that subarea.

The advantages of plotting maps as a `SpatialPolygon` rather than a simple polygon are:

a) the aspect ratio of the x and y axes in plots is preserved to reflect geography

b) specific subareas can be highlighted easily

c) subareas that consist of more than one contiguous land mass can be treated as one unit

As an demonstration of these three advantages, in Figure 1 we plot a map of Scotland with all 56 counties of Scotland in 1975 both as a polygon (using the `R polygon()` function) and as a `SpatialPolygons` object. Several of the counties of Scotland consist of more than one contiguous land mass, e.g. the county Argyll-Bute consists of 8 separate land masses.

```
> data(scotland)
> polygon <- scotland$polygon$polygon
> nrepeats <- scotland$polygon$nrepeats
> names <- scotland$data$county.names
> spatial.polygon <- polygon2spatial.polygon(polygon, coordinate.system = "+proj=utm",
+     names, nrepeats)
```

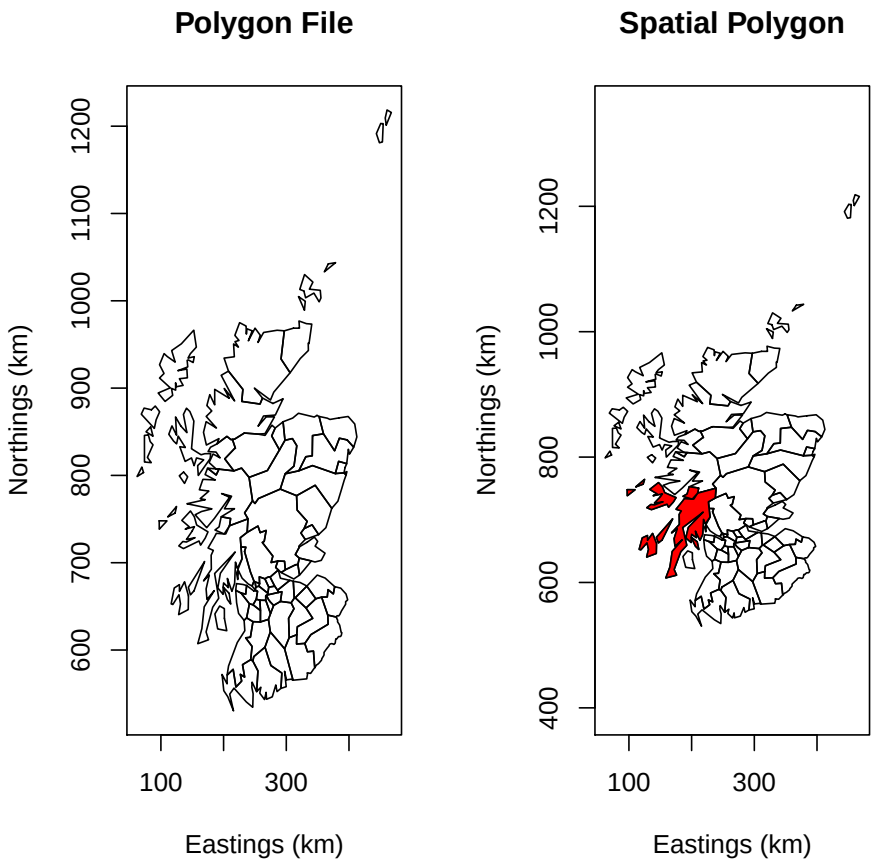

Figure 1: Plots of Scotland as a polygon and a `SpatialPolygon`

```
> par(mfrow = c(1, 2))
> plot(polygon, type = "n", xlab = "Eastings (km)", ylab = "Northings (km)",
+     main = "Polygon File")
> polygon(polygon)
> plot(spatial.polygon, axes = TRUE)
> title(xlab = "Eastings (km)", ylab = "Northings (km)", main = "Spatial Polygon")
> plot(spatial.polygon[23], add = TRUE, col = "red")
```

2.1.2 Converting maps objects to SpatialPolygons

The **maps** `R` package includes several commonly used maps, which can be converted into `SpatialPolygons` objects using the `map2SpatialPolygons` command. For instance, a county-level map of just the states Pennsylvania and Vermont with red borders, along with the boundaries of neighboring states with slightly thicker black borders can be produced with the resulting plot shown in Figure 2).

```
> library(maps)
> county.map <- map("county", c("pennsylvania", "vermont"), fill = TRUE,
+     plot = FALSE)
```

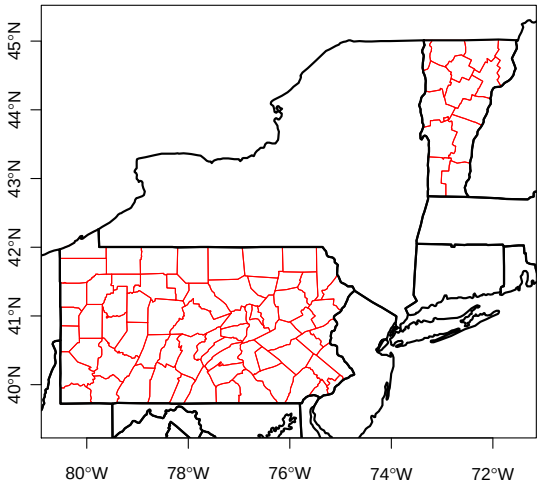

Figure 2: County-Level Map of Pennsylvania and Vermont

```
> county.names <- as.character(county.map$names)
> county <- map2SpatialPolygons(county.map, IDs = county.names,
+     proj4string = CRS("+proj=longlat"))
> state.map <- map("state", c(), fill = TRUE, plot = FALSE)
> state.names <- as.character(state.map$names)
> state <- map2SpatialPolygons(state.map, IDs = state.names, proj4string = CRS("+proj=longlat"))
> plot(county, axes = TRUE, border = "red")
> plot(state, add = TRUE, lwd = 2)
```

2.2 Converting Between Coordinate Systems

In the Pennsylvania and Vermont example in Section 2.1.2, all coordinates are in longitude/latitude. However, this coordinate system is not appropriate for many distance-based methods as degrees of longitude are not equidistant, and must be converted to a grid based system.

The function `latlong2grid()` can convert either

a) an `n`×2 matrix of coordinates

b) a `SpatialPolygons` object

based on longitude/latitude (expressed in decimal values) into kilometer-based grid coordinates. Figure 3 shows the resulting transformed map.

```
> county.grid <- latlong2grid(county)
> state.grid <- latlong2grid(state)
> plot(county.grid, axes = TRUE, border = "red")
> plot(state.grid, add = TRUE, lwd = 2)
```

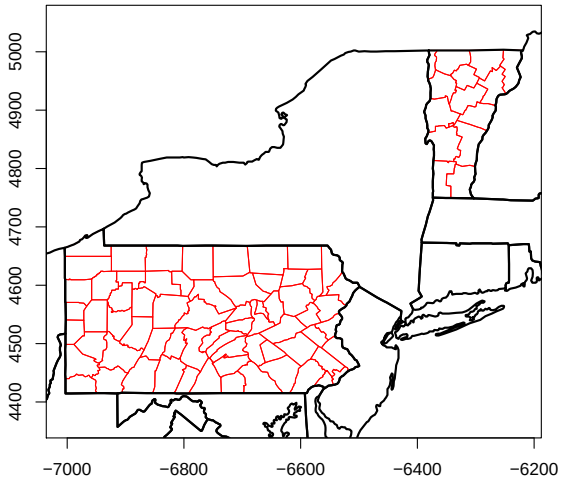

Figure 3: County-Level Map of Pennsylvania Using UTM Coordinate System

Or a simple 2-column matrix of coordinates can be converted as well. As an example, consider the latitude and longitudes of Montreal QC (latitude: 45deg 28' 0" N (deg min sec), longitude: 73deg 45' 0" W) and Vancouver BC (latitude: 45deg 39' 38" N (deg min sec), longitude: 122deg 36' 15" W) in decimal format. These also can be converted to a grid-based coordinate system.

```
> coord <- rbind(c(-73.75, 45.4667), c(-122.6042, 45.6605))
```

```
          [,1]    [,2]
[1,]  -73.7500 45.4667
[2,] -122.6042 45.6605
```

```
> latlong2grid(coord)
```

```
          x        y
1  -6414.30 5052.849
2 -10663.32 5074.387
```

2.3 Plotting a Variable

In spatial epidemiology, the production of disease atlases (a geographic map along with some scheme to illustrate the different levels of a disease) are often used as visual summaries of geographic disease patterns. The `mapvariable()` command simplifies the production of such atlases by taking in a `SpatialPolygons` object and a vector as inputs. For example, consider plotting random Uniform(0,1) random variables on a map of Scotland in Figure 4.

```
> data(scotland)
> scotland.map <- scotland$spatial.polygon
> y <- runif(nrow(scotland$data))
> mapvariable(y, scotland.map)
```

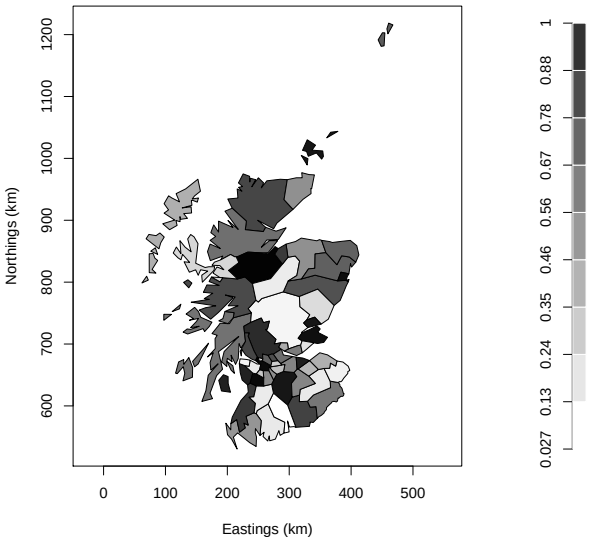

Figure 4: Random Values in Scotland between 1975-1980

3 Data Examples

A spatial epidemiology data set typically consists of:

- Geographically indexed disease case and population data
- Geographic data
- (if available) Covariate data

3.1 Pennsylvania Lung Cancer & Smoking Data

This dataset consists of counts of lung cancer incidence in Pennsylvania for the year 2002 as reported by the Pennsylvania Department of Health. This case data was paired with the corresponding population data from the 2000 Census. Furthermore, smoking data is provided from a behavioral risk study conducted by the Pennsylvania Department of Health and are presented as the percentage of adults who were current smokers sometime between 1996 and 2000. Both population and case counts are stratified by gender, age (0-39, 40-59, 60-69 and 70+) and race (white and other) for a total of 16 strata levels. We plot lung cancer incidence per 1000 individuals in Figure 5:

```
> data(pennLC)
> penn.map <- pennLC$spatial.polygon
> penn.map <- latlong2grid(penn.map)
> population <- tapply(pennLC$data$population, pennLC$data$county,
+     sum)
> cases <- tapply(pennLC$data$cases, pennLC$data$county, sum)
> geo <- latlong2grid(pennLC$geo[, 2:3])
> incidence <- (cases/population) * 1000
> mapvariable(incidence, penn.map)
```

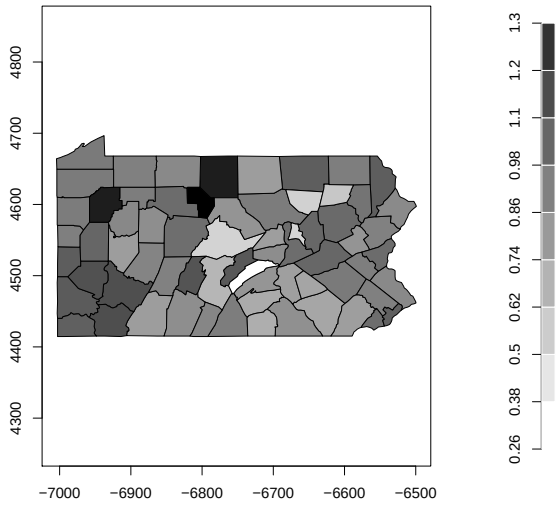

Figure 5: Lung cancer incidence per 1000 in Pennsylvania in 2002

3.2 Scotland Lip Cancer among Males in 1975-1980

AFF agriculture farming and fishing

The expected numbers of disease were calculated by the method of Mantel and Stark [10].

[7]

```
> data(scotland)
> scotland.map <- scotland$spatial.polygon
> y <- scotland$data$cases
> E <- scotland$data$expected
> SMR <- y/E
> mapvariable(SMR, scotland.map)
```

3.3 Infant Mortality in North Carolina

This data consists of counts of infant births and deaths in North Carolina for the years 2000-2004 as reported by the Howard W. Odum Institute for Research in Social Science at the University of North Carolina at Chapel Hill http://www.irss.unc.edu/. The data are stratified by race (white and non-white), gender and low birth weight status (defined to be a birth weight of less than 2,500 g) for a total of 8 strata levels [13].

Data on both a county level and aggregated to the 10 regions of North Carolina (as described in Wakefield and Haneuse 2008 [13]), are included. We plot the number infant births between 2000-2004 in Figure 7 and infant mortality per 1,000 infants in Figure 8

```
> data(NC)
> NC.county.map <- NC$county$spatial.polygon
> NC.region.map <- NC$region$spatial.polygon
> county.population <- tapply(NC$county$data$population, NC$county$data$county,
+     sum)
```

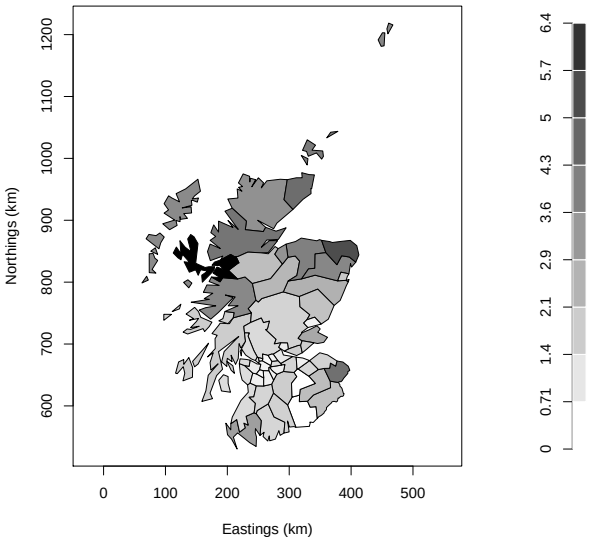

Figure 6: SMR of lung cancer among males in Scotland between 1975-1980

```
> region.population <- tapply(NC$region$data$population, NC$region$data$region,
+     sum)
> region.cases <- tapply(NC$region$data$cases, NC$region$data$region,
+     sum)
> incidence <- (region.cases/region.population) * 1000
> mapvariable(county.population, NC.county.map)
> mapvariable(incidence, NC.region.map)
```

4 Methods

4.1 Expected Numbers of Disease and Standardized Mortality Ratios

In order to control for known risk factors (in this case strata) using internal indirect standardization, we can compute the (indirect) expected number of diseases [6] for each area using the `expected()` command. It is important that the population and cases vectors are 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. i.e. if considering 16 strata, the first 16 elements correspond to the first area, the next 16 correspond to the second area, etc. and the strata are always listed in the same order.

```
> data(pennLC)
> n.strata <- 16
> population <- tapply(pennLC$data$population, pennLC$data$county,
+     sum)
> cases <- tapply(pennLC$data$cases, pennLC$data$county, sum)
> expected.cases <- expected(pennLC$data$population, pennLC$data$cases,
+     n.strata)
```

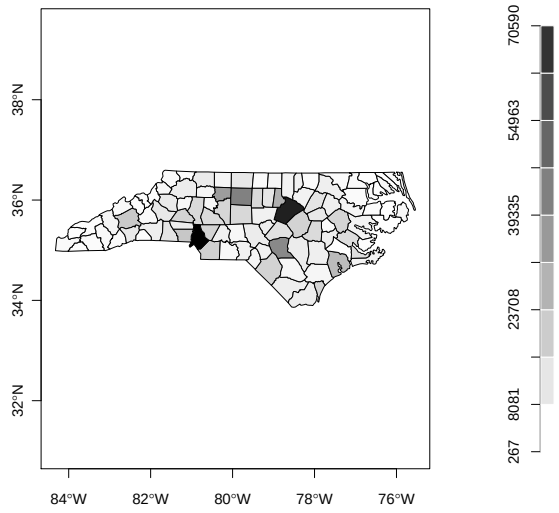

Figure 7: Infants Births in Each County in North Carolina 2000-2004

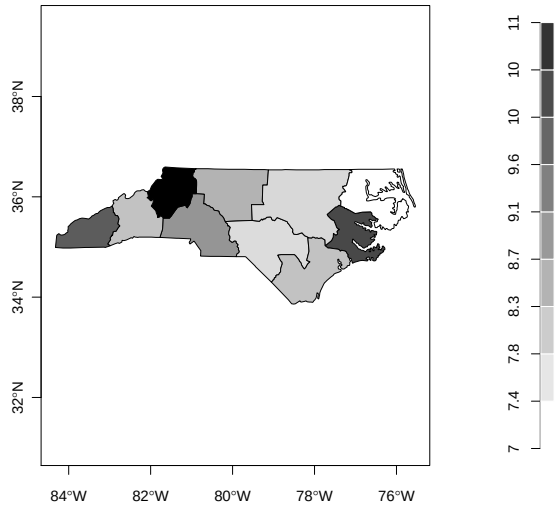

Figure 8: Deaths per 1000 Infants in Each Region in North Carolina 2000-2004

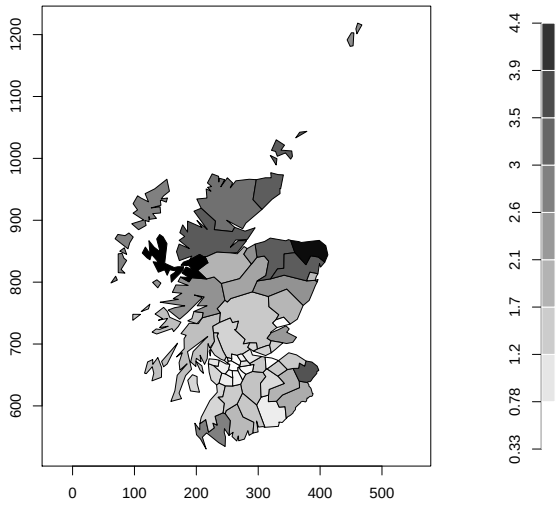

Figure 9: Empirical Bayes estimates of the relative risk of lung cancer

4.2 Disease Mapping

4.2.1 Empirical Bayes

Given that SMR estimates for areas with small values of expected numbers of disease can be highly variable, Clayton and Kaldor proposed an empirical Bayes approach to estimate disease rates [4]. The estimates represent a weighted compromise between an area's SMR and the overall mean relative risk. These estimates are much more stable than the raw SMR's. In this example, we use a linear model $\alpha + \beta_1 x + \beta_2 x^2$ in the `eBayes()` function to estimate relative risk with the resulting plot in Figure 9

```
> 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)
```

4.3 Cluster Detection

Cluster detection is the routine surveillance of a large expanse of small administrative zones for evidence of individual "hot-spots" of disease without any preconceptions about their locations. [2]. For aggregated count data, a typical procedure is to consider a set of *zones*, each zone being some amalgamation of areas in the study regions. Each zone is then viewed as a potential cluster.

4.3.1 Kulldorff

The `kulldorff()` function implements the Kulldorff method for finding the most likely cluster as described in Kulldorff and Nagarwalla (1995) [9] and Kulldorff (1997) [8]. The `kulldorff()`

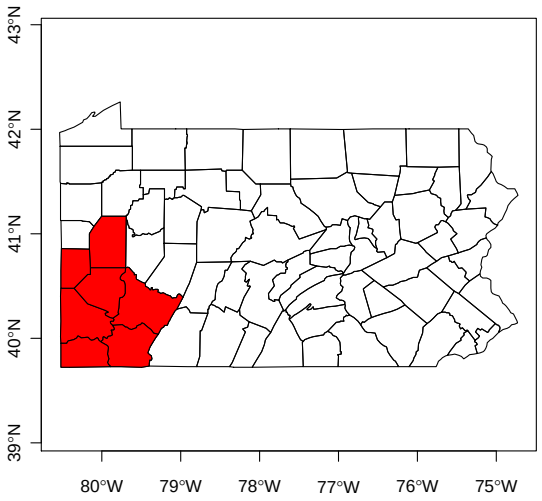

Figure 10: Binomial Analysis Kulldorff

```
> data <- pennLC$data
> geo <- latlong2grid(pennLC$geo[, 2:3])
> population <- tapply(data$population, data$county, sum)
> cases <- tapply(data$cases, data$county, sum)
> expected.cases <- expected(data$population, data$cases, 16)
> pop.upper.bound <- 0.5
> n.simulations <- 999
> alpha.level <- 0.05
> plot <- TRUE
```

We can pursue either a binomial or poisson analysis.

```
> binomial <- kulldorff(geo, cases, population, NULL, pop.upper.bound,
+     n.simulations, alpha.level, plot)
> cluster <- binomial$most.likely.cluster$location.IDs.included
> plot(pennLC$spatial.polygon, axes = TRUE)
> plot(pennLC$spatial.polygon[cluster], add = TRUE, col = "red")
> title("Most Likely Cluster")
```

```
> poisson <- kulldorff(geo, cases, population, expected.cases,
+     pop.upper.bound, n.simulations, alpha.level, plot)
> cluster <- poisson$most.likely.cluster$location.IDs.included
> plot(pennLC$spatial.polygon, axes = TRUE)
> plot(pennLC$spatial.polygon[cluster], add = TRUE, col = "red")
> title("Most Likely Cluster Controlling for Strata")
```

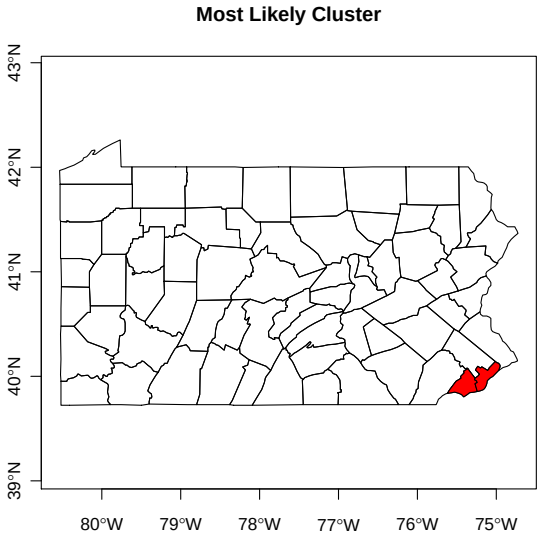

Figure 11: Poisson Analysis Kulldorff

4.3.2 Besag-Newell

The `besag.newell()` function implements the Besag-Newell method as described in Besag and Newell (1995) [2].

Using the same dataset as in Section 4.3.1

```
> k <- 1250
> alpha.level <- 0.05
> results <- besag.newell(geo, population, cases, expected.cases = NULL,
+     k, alpha.level)
> results <- besag.newell(geo, population, cases, expected.cases,
+     k, alpha.level)
```

4.4 Generating Samples from an Improper Gaussian Random Field

Consider an improper GMF, see for example Besag [1] and Rue [12]. The model has the format:

$$p(\boldsymbol{u} \mid \sigma_u^2) = (2\pi)^{-(m-r)/2} |Q^*|^{1/2} \sigma_u^{-(m-r)} \exp(-\frac{1}{2\sigma_u^2} \boldsymbol{u}^T \boldsymbol{Q} \boldsymbol{u})$$

where $\boldsymbol{u} = (u_1, u_2, \cdots, u_m)$ is the vector of random effects. $\boldsymbol{Q}$ is a (scaled) "precision" matrix of rank. $\tilde{\boldsymbol{Q}}$ is the precision matrix with rank $m - r$.

To simulate samples from this distribution, the algorithm is:

1. Simulate $z_j \sim N(0, \lambda_j^{-1})$, $j = r+1, r+2, \cdots, m$, where λ_j are the eigenvalues of Q.
2. Let $\boldsymbol{x} = z_{r+1} e_{k+1} + z_{r+2} e_{r+2} + \cdots z_m e_m$.

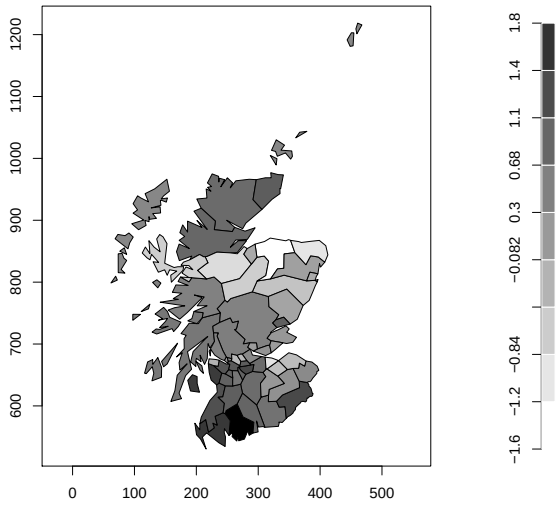

Figure 12: Simulated spatial random effect

To simulate spatial correlated data, we use function `sim.Q` in `SpatialEpi`. You need to provide the vector of number of neighbors, the vector of adjacency neighbors and the conditional variance σ_u^2. If your graph file is in the format as in `INLA`, you can use function `num.neighbors` and `num.neighbors` in `SpatialEpi`. These two functions will retrieve the vector of number of neighbors and the vector of adjacency neighbors from the graph file in the format in `INLA`. We take the scotland lip cancer data as an example. The following R code is used to generate samples from improper Gaussian random field with $\sigma_u^2 = 1$ using the scotland graph file from INLA. The simulated samples are in Figure 12.

```
> library(INLA)
> g <- system.file("demodata/scotland.graph", package = "INLA")
> m <- numNeighbors(g)
> adj <- listNeighbors(g)
> Q <- make.Q(m, adj, omega.sq = 1)
> tmp.x1 <- sim.Q(Q)
```

4.4.1 Choosing the Prior on the Fixed Effects

Informative priors can be independently specified on each of the fixed effects. The parameters in the normal priors are assigned via specification of two quantiles with associated probabilities. For logistic and log-linear models these quantiles may be given on the exponentiated scale since these are more interpretable. If θ_1 and θ_2 are the quantiles on the exponentiated scale, and p_1 and p_2 are the associated probabilities, then the parameters of the normal prior are given by:

$$\begin{aligned} \mu &= \frac{z_2 \log(\theta_1) - z_1 \log(\theta_2)}{z_2 - z_1} \\ \sigma &= \frac{\log(\theta_2) - \log(\theta_1)}{z_2 - z_1} \end{aligned}$$

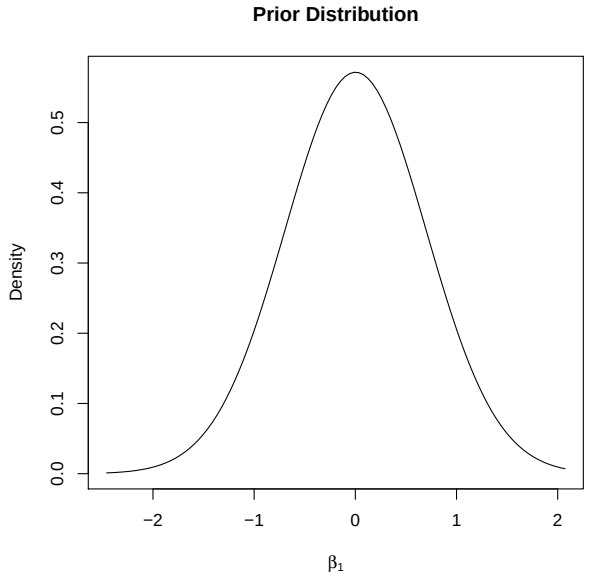

Figure 13: Prior distribution of fixed effect β_1

where z_1, z_2 are the p_1, p_2 quantiles of a standard normal distribution. For example, in an epidemiological context, we may wish to specify a prior on a relative risk parameter, $\exp(\beta_1)$, which has a median of 1 and a 95% point of 3 (if we think it is unlikely that the relative risk associated with a unit increase in exposure exceeds 3). These speciÃdcations lead to $\beta_1 \sim N(0, 0.6682^2)$.

In R, the above calculation can be obtained by calling function `LogNormalPriorCh`.

```
> param <- LogNormalPriorCh(theta1 = 1, theta2 = 3, prob1 = 0.5,
+     prob2 = 0.95)
> param

$mu
[1] 0

$sigma
[1] 0.6679088
```

The resulting log normal distribution with parameter $\mu = 0$ and $\sigma = 0.6682$ is shown in Figure 13.

4.4.2 Choosing Prior on the Non-Spatial Residuals

The approach for choosing a prior for a single random effect is based on Wakefield(2009). The idea is to specify a range for the more interpretable marginal distribution of the random effect b_i, and use this to drive specification of prior parameters. It can be proven that if $b|\tau \sim N(0, \tau^{-1})$ and $\tau \sim Ga(a, b)$, then the marginal distribution of b is $T_1(0, b/a, 2a)$. For a range of b and a pre-specified degree of freedom d, we can solve for a and b. For example, if the range of b is $(-R, R)$, a and b can be solved by:

$$\begin{aligned} a &= d/2 \\ b &= R^2/2(t^d_{1-(1-q)/2})^2 \end{aligned}$$

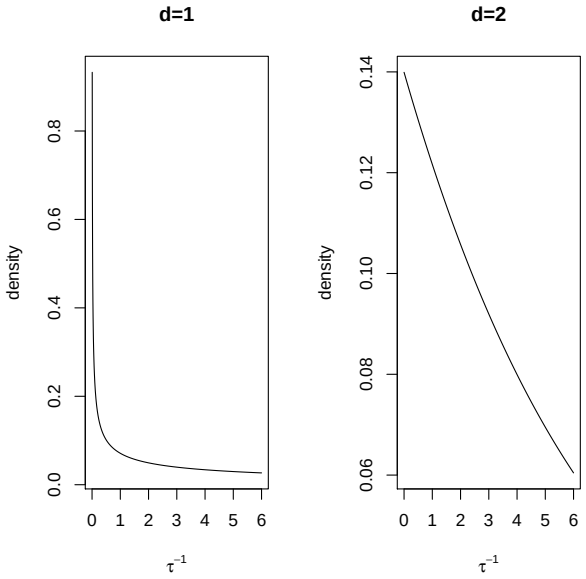

Figure 14: Prior distribution of non-spatial residual error precision τ^{-1}

where t_q^d is the $100\times q^{th}$ quantile of a Student t distribution with d degrees of freedom. For example, if we choose $d = 1$ and a 95% range of $\exp(b)[0.1, 10]$ we take $R = \log(10)$ and obtain $a = 0.5, b = 0.0164$.

Function `GammaPriorCh` can be used for this calculation:

```
> param1 <- GammaPriorCh(log(10), 0.975, 1)

Gamma Parameters:  0.5 0.01641987

> param2 <- GammaPriorCh(log(5), 0.975, 2)

Gamma Parameters:  1 0.1399187
```

4.4.3 Choosing the Prior on the Spatial Residuals

References

[1] J. Besag, P.J. Green, D. Higdon, and K. Mengersen. Bayesian computation and stochastic systems (with discussion). *Statistical Science*, 10:3–66, 1995.

[2] J. Besag and J. Newell. The detection of clusters in rare diseases. *Journal of the Royal Statistical Society: Series A*, 154(1):143–155, 1991.

[3] R. S. Bivand, E. J. Pebesma, and V. Gomez-Rubio. *Applied Spatial Data Analysis with R*. Springer Series in Statistics. Springer, New York, 1st edition, 2008.

[4] D. Clayton and J. Kaldor. Empirical Bayes estimates of age-standardized relative risks for use in disease mapping. *Biometrics*, 43(3):671–681, Sep. 1987.

[5] P. Elliot and D. Wartenberg. Spatial epidemiology: Current approaches and future challenges. *Environmental Health Perspectives*, 112(9):998–1006, Jun. 2004.

[6] P. Elliott, J. Wakefield, N. Best, and D. Briggs. *Spatial Epidemiology: Methods and Applications*. Oxford Medical Publications. Oxford University Press, Oxford, 2nd edition, 2000.

[7] I. Kemp, P. Boyle, M. Smans, and C. Muir. *Atlas of cancer in Scotland, 1975-1980, incidence and epidemiologic perspective*. Number 72 in IARC Scientific Publications. International Agency for Research on Cancer, Lyon, 1985.

[8] M. Kulldorff. A spatial scan statistic. *Communication in Statistics: Theory and Methods*, 26(6):1481–1496, 1997.

[9] M. Kulldorff and N. Nagarwalla. Spatial disease clusters: detection and inference. *Statistics in Medicine*, 14:799–810, 1995.

[10] N. Mantel and C. R. Stark. Computation of indirect-adjusted rates in the presence of confounding. *Biometrics*, 24:23–37, 1968.

[11] E. J. Pebesma and R. S. Bivand. Classes and methods for spatial data in R. *R News*, 5(2):9–13, Nov. 2005.

[12] H. Rue and L. Held. *Gaussian Markov Random Fields:Theory and Application*. Monographs on Statistics & Applied Probability Statistical Science. Chapman and Hall/CRC, 1st edition, 2005.

[13] J. Wakefield and S. Haneuse. Overcoming ecologic bias using the two-phase study design. *American Journal of Epidemiology*, 167:908–916, 2008.