Journal of Statistical Software

October 2021, Volume 99, Issue 13. doi: 10.18637/jss.v099.i13

Flexible Scan Statistics for Detecting Spatial Disease Clusters: The rflexscan R Package

Takahiro Otani
Nagoya City University

Kunihiko Takahashi
Tokyo Medical and Dental University

Abstract

The spatial scan statistic is commonly used to detect spatial disease clusters in epidemiological studies. Among the various types of scan statistics, the flexible scan statistic proposed by Tango and Takahashi (2005) is one of the most promising methods to detect arbitrarily-shaped clusters. In this paper, we introduce a new R package, **rflexscan** (Otani and Takahashi 2021), that provides efficient and easy-to-use methods for analyses of spatial count data using the flexible spatial scan statistic. The package is designed for any of the following interrelated purposes: to evaluate whether reported spatial disease clusters are statistically significant, to test whether a disease is randomly distributed over space, and to perform geographical surveillance of disease to detect areas of significantly high rates. The functionality of the package is demonstrated through an application to a public-domain small-area cancer incidence dataset in New York State, USA, between 2005 and 2009.

Keywords: cluster detection, hotspot cluster, flexible scan statistics, spatial epidemiology, R.

1. Introduction

Evaluating whether a disease is randomly distributed or tends to occur as clusters over space is among the most crucial aspects of epidemiological studies, and this can be primarily performed using disease mapping. As an example, Figure 1 shows a choropleth map of standardized incidence ratios (SIRs) of breast cancer (female) in the Manhattan borough of New York City, comprising 982 census blocks for the years 2005–2009 based on the age-specific incidence rates from the 2010 census counts for New York State (Boscoe, Talbot, and Kulldorff 2016). The total observed number of breast cancer cases for the 5 years was 6,219, and the SIR for the entire area was 1.07. Note that the age adjustment is based on the population structure of New York State, not that of Manhattan. One might interpret from the map that breast cancer cases are clustered in a certain area. However, such a map often does not indicate the existence of meaningful clusters clearly and identifying them objectively is still challenging.

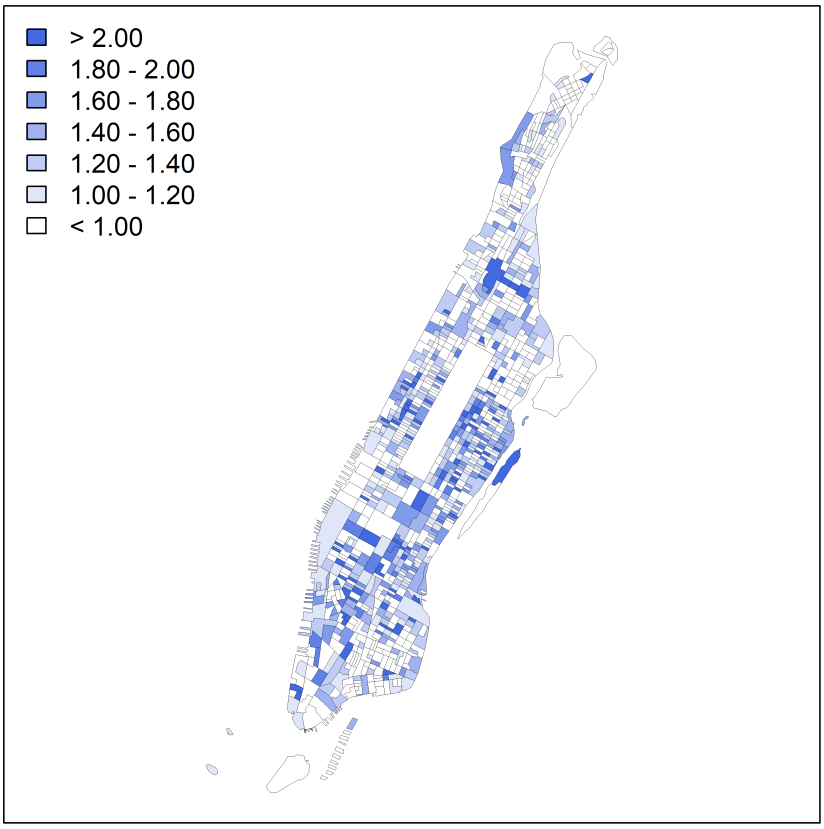


Figure 1: Standardized incidence ratios of breast cancer (female) in the Manhattan borough of New York City for the years 2005–2009 based on the 2010 census counts.

Various statistical tests have been proposed to address this issue (Kulldorff 2006; Rogerson and Yamada 2008; Tango 2010). Among those, cluster detection tests (CDTs) investigate whether a disease pattern is completely random over a space, without any prior information, while indicating regions with high disease prevalence. The spatial scan statistic (Kulldorff 1997) based on the maximum likelihood ratio is one of the most powerful methods of the CDT. However, Kulldorff's statistic considers only circular or elliptic shaped clusters and has difficulty in correctly detecting clusters with other shapes. To detect arbitrarily-shaped clusters, Tango and Takahashi (2005) proposed the flexible scan statistic, which is designed so that the detected cluster can be flexible in shape, while concurrently, the cluster is confined within relatively small neighborhoods of each region. Tango and Takahashi (2012) further proposed a flexible scan statistic with a restricted likelihood ratio that consumes much less computation time than the original one and tends to detect clusters of any shape reasonably well as the relative risk of the cluster increases.

Several software for performing tests based on the Kulldorff's scan statistic are available. The **SaTScan** software developed by Kulldorff and Information Management Services, Inc. (2018) is freely available and has been widely used. R (R Core Team 2021) packages such as **rsatscan** (Kleinman 2015), **SpatialEpi** (Kim and Wakefield 2021), **scanstatistics** (Allévius 2018), and **smerc** (French 2021) are also available from the Comprehensive R Archive Net-

DOHRegion	Latitude	Longitude	Observed no.	Expected no.
360610002011	40.71368	−73.98611	5	2.63
360610002012	40.71119	−73.98574	4	9.86
360610002021	40.71389	−73.98232	4	3.43
360610002022	40.71135	−73.98281	10	9.21
⋮	⋮	⋮	⋮	⋮

Table 1: Observed/expected number of breast cancer cases in Manhattan which comprises 982 regions (census blocks), 2005–2009. DOHRegion is a 12-digit geographic identifier that includes coding state, county, tract, and block group, based on the 2010 census. Centroid coordinates are also described by latitudes and longitudes.

work (CRAN). Other R packages for detection of clusters include the **DClusterm** package (Gómez-Rubio, Moraga, Molitor, and Rowlingson 2019) which uses a model-based approach and allows the inclusion of covariates, and the **SpatialEpiApp** package (Moraga 2017) which detects clusters using the spatial scan statistics implemented in **SaTScan** and allows to visualize the results through interactive maps and tables. Meanwhile, there is no R package that can efficiently conduct tests based on the flexible scan statistic, although a stand-alone application, **FleXScan** (Takahashi, Yokoyama, and Tango 2013), is freely available. Packages **scanstatistics** and **smerc** can detect clusters using the flexible scan statistic, and a wrapper of **smerc**, **FlexScan** (Du and Hao 2021), is also available. However, it is difficult to detect large clusters using these packages because of heavy computational load (see Section 5).

In this paper, we introduce a new R package **rflexscan** (Otani and Takahashi 2021), that is an R implementation of the **FleXScan** software for performing purely spatial analysis using the flexible scan statistic more efficiently. Package **rflexscan** is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=rflexscan`. The feasibility of the package is demonstrated through an application to the public domain small-area cancer incidence data for New York State during 2005–2009 (Boscoe *et al.* 2016), which are available at `https://www.satscan.org/datasets/nyscancer/`. Table 1 shows an example of breast cancer cases in Manhattan consisting of the observed and age-adjusted expected number of cases of breast cancer as well as centroid coordinates for each region. The whole dataset contains the observed and the age- and sex-adjusted expected number of cases for 23 anatomic sites of cancer diagnosed in New York State at the census block group level as well as geographic identifiers and centroid coordinates. The paper is organized as follows. Section 2 introduces the spatial scan statistics, and Section 3 describes the functionality of the **rflexscan** package. In Section 4, we demonstrate the feasibility of the package by applying it to cancer incidence data. Section 5 discusses the computational efficiency of the package and provides a benchmark result, and finally, Section 6 concludes the paper with some points to consider.

2. Methods

The spatial scan statistic (Kulldorff 1997) tries to identify the most likely cluster (MLC), defined as the set of connected regions, i.e., window, that achieves the maximum likelihood ratio by searching (scanning) over a set of candidates for the hotspot cluster with signifi-

cant elevated risk. In this paper, we describe the Poisson model that is based on the observed/expected number of disease cases and can adjust for potential confounders such as sex and age. In contrast, the Binomial model that is based on the observed number of disease cases and the background population at risk in each area is also implemented in **rflexscan**. For more information on the Binomial model, please refer to Kulldorff (1997).

Let us assume that the entire study area is divided into m regions (e.g., counties or enumeration districts). The number of cases of a particular disease in region i is denoted by the random variable N_i with observed value n_i $(i = 1, \ldots, m)$ and the total number of cases $n = n_1 + \cdots + n_m$ where m is the number of regions in the study area. Let $\mathcal{W}$ be the set of all potential scanning windows of any size. With the use of the notation of window $w \in \mathcal{W}$, let us assume that the relative risk is θ_w for regions inside of w and is $\theta_{\bar{w}}$ for regions outside of w. The N_i $(i = 1, \ldots, m)$ are independent Poisson variables such that

$$N_i \sim \text{Poisson}(\theta_w \times \xi_i), i \in w$$
$$N_i \sim \text{Poisson}(\theta_{\bar{w}} \times \xi_i), i \notin w$$

where Poisson(ξ) denotes the Poisson distribution with mean ξ, ξ_i is the age-adjusted expected number of cases in region i under the null hypothesis, and $n = \xi_i + \cdots + \xi_m$. For calculation of the expected number of cases adjusted for potential confounders such as age, we can use indirect standardization (Waller and Gotway 2004).

2.1. Spatial scan statistic

The null hypothesis of no clustering is expressed as

$$H_0 : \theta_w = \theta_{\bar{w}}, \forall w \in \mathcal{W}. \tag{1}$$

Meanwhile, under the alternative hypothesis H_1, there is at least one window w for which the underlying risk is higher inside the window than the outside, that is,

$$H_1 : \theta_w > \theta_{\bar{w}}, \exists w \in \mathcal{W}.$$

Under the Poisson assumption, the likelihood ratio for window w is given by

$$\lambda(w) = \begin{cases} \left(\frac{n(w)}{\xi(w)}\right)^{n(w)} \left(\frac{n-n(w)}{n-\xi(w)}\right)^{n-n(w)}, & n(w) > \xi(w), \\ 1, & \text{otherwise}, \end{cases} \tag{2}$$

where $n(w)$ is the observed number of cases in the window w. The MLC w^* is defined as

$$w^* = \arg\max_{w \in \mathcal{W}} \lambda(w),$$

and a test (scan) statistic to assess the statistical significance of w^* is defined as $\lambda^* = \lambda(w^*)$. The above statistic is widely used; however, Tango and Takahashi (2005) and Tango (2000) have shown that the scan statistic using the likelihood ratio given in Equation 2 tends to detect an MLC that is much larger than the true cluster by swallowing neighboring regions with non-elevated risk. To avoid or scale back such undesirable phenomena, Tango (2008) proposed the following restricted likelihood ratio by considering each region's risk.

$$\lambda'(w) = \begin{cases} \lambda(w) \prod_{i \in w} I(p_i < \alpha_1), & n(w) > \xi(w), \\ 1, & \text{otherwise}, \end{cases} \tag{3}$$

where $I()$ is the indicator function, and p_i is the one-tailed p value of the test for $H_0 : N_i = \xi_i$ given by the middle p value

$$p_i = \Pr\{N_i \geq n_i + 1 | N_i \sim \text{Poisson}(\xi_i)\} + \frac{1}{2}\Pr\{N_i = n_i | N_i \sim \text{Poisson}(\xi_i)\},$$

and α_1 is a prespecified significance level for the individual region. The MLC w^* using the restricted likelihood ratio is defined as

$$w^* = \arg\max_{w \in \mathcal{W}} \lambda'(w),$$

and the test statistic is $\lambda^* = \lambda(w^*)$. Note that it is equivalent to the original likelihood ratio in Equation 2 when $\alpha_1 = 1$.

2.2. Windows to be scanned

According to the different definition of $\mathcal{W}$, various scan statistics can be derived. The **rflexscan** package implements the flexible scan statistic (Tango and Takahashi 2005) and Kulldorff's scan statistic (Kulldorff 1997).

The flexible scan statistic imposes a flexibly shaped window on each centroid of the region by connecting its adjacent regions. For any given region i, the method creates the set of flexibly shaped windows with length k consisting of k connected regions including i and lets k move from 1 to the prespecified maximum length K. To avoid detecting a cluster of unlikely peculiar shape, i.e., over-sized oddly shaped with multiple narrow branches, the connected regions are restricted to the subsets of the set of regions i and K-nearest neighbors to the region i. In total, multiple different, but overlapping arbitrarily-shaped windows are created, each with a different location and size, and each being a potential cluster. Let w_{ik}, $(k = 1, \ldots, K)$ denote a window composed by the region i and $(k-1)$ nearest neighbors to i. Also, let w_{ikl} $(k = 1, \ldots, K,\ l = 1, \ldots, L_{ik})$ denote a flexibly shaped window that is a set of k regions connected starting from the region i, where L_{ik} is the number of windows satisfying $w_{ikl} \subseteq w_{iK}$. Then, all the windows to be scanned are included in the set

$$\mathcal{W}_f = \{w_{ikl} | 1 \leq i \leq m, 1 \leq k \leq K, 1 \leq l \leq L_{ik}\}.$$

Meanwhile, the Kulldorff's original scan statistic imposes a circular window on each centroid of regions. For any of those centroids, the radius of the circle varies from zero to an upper limit defined by the user. If the window contains the centroid of a region, then that whole region is included in the window. In total, as in the flexible spatial scan statistic, several different, but overlapping circular windows are created. In the **rflexscan**, the upper limit is determined by specifying the maximum length K of nearest neighbors, while the standard option in **SaTScan** is to specify it as 50% of the population at risk. With the use of the notation of window w_{ik}, all the windows to be scanned are included in the set

$$\mathcal{W}_c = \{w_{ik} | 1 \leq i \leq m, 1 \leq k \leq K\}.$$

The scan statistic based on $\mathcal{W}_c$ will be referred to as circular scan statistic.

2.3. Calculating p value

The Monte Carlo test (Dwass 1957) is used to determine the significance of w^*. Under the null hypothesis, we randomly generate B samples of the observed number of cases n_i in each region i using the Poisson distributed random number generator (`rpois` method) and calculate test statistics $\lambda_b = \lambda(w^*) \quad (b = 1, 2, \ldots, B)$ based on each sample. For the test statistic λ^* derived from the actual data, its p value is approximated by

$$\hat{p} = \frac{1 + \sum_{b=1}^{B} I(\lambda_b \geq \lambda^*)}{B + 1}.$$

Note that the distribution of the test statistic under the null hypothesis varies depending on the scan statistic used, and the parameters used in constructing the window, and the resulting p value also varies.

2.4. Detecting secondary clusters

The aforementioned procedure was intended to identify only the primary cluster, $w_1^* = w^*$. Kulldorff (1997) extended its use for detecting multiple clusters; the procedure was repeatedly used to identify other clusters, i.e., secondary clusters, $w_2^*, w_3^*, \ldots$, among which there were no overlaps, i.e., $w_k^* \cap w_{k'}^* = \emptyset$ for $k \neq k'$. Consequently, their likelihood ratios always followed a descending order, $\lambda(w_1^*) > \lambda(w_2^*) > \cdots > \lambda(w_k^*) > \cdots$. The statistical significance of secondary clusters was evaluated in the same way as that of the MLC, i.e., the likelihood ratio of each secondary cluster was compared with that calculated from randomly generated data sets.

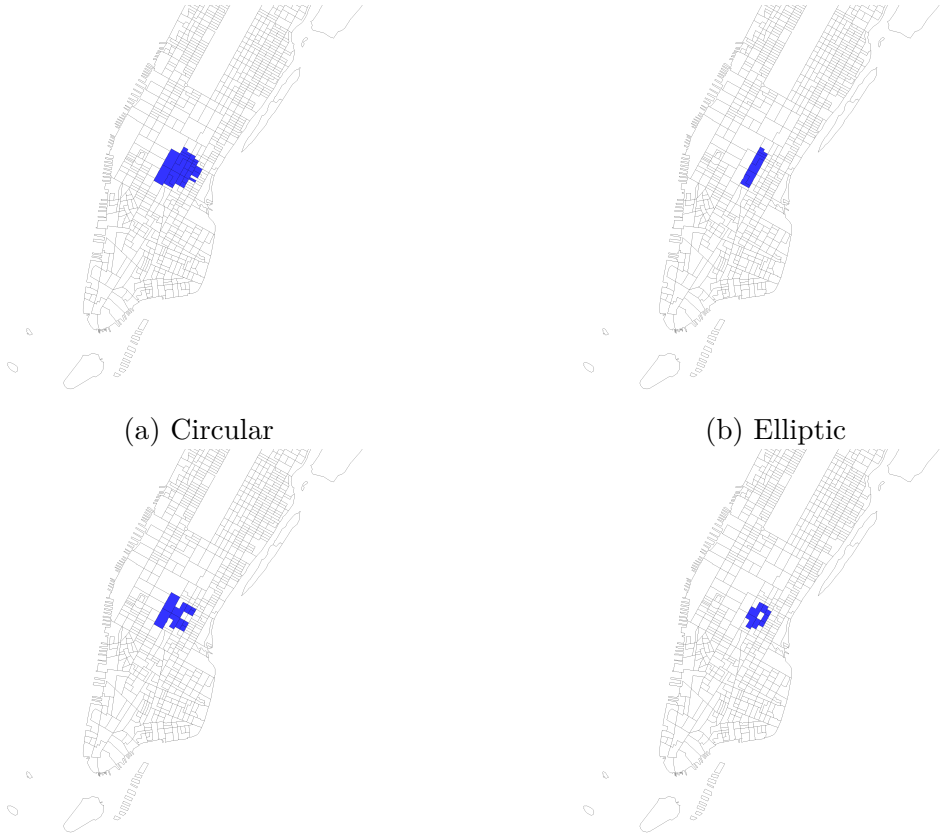
Note that the above procedure does not require a correction for multiple testing. The scan statistic avoids multiplicity by testing whether at least one cluster exists rather than repeating the test for the existence of each clusters. This is also the case for the detection of secondary clusters. The procedure for secondary clusters evaluates these clusters one by one, and each corresponding p value is calculated as if the cluster were the primary one, i.e., the MLC. The interpretation of this approach is that we are evaluating whether the secondary clusters are able to reject the null hypothesis in Equation 1 on their own strength, whether the MLC is a true cluster or not. A drawback of this approach is that the p values are conservative (Kulldorff 1997; Zhang, Assunção, and Kulldorff 2010) with a corresponding loss of statistical power.

3. The rflexscan package

The R package **rflexscan** includes functions for analyzing spatial count data using the flexible spatial scan statistics as well as the circular scan statistic. This package can be installed and loaded as usual:

```
R> install.packages("rflexscan", dependencies = TRUE)
R> library("rflexscan")
```

Figure 2 shows examples of flexibly shaped clusters that can be detected using **rflexscan**. **SaTScan** software and **rsatscan** package can detect circular and elliptical clusters, but not



(a) Circular (b) Elliptic

(c) Cross (d) Loop

Figure 2: Examples of flexible shapes that can be constructed.

cross and loop shaped clusters. In contrast, the **rflexscan** package can detect clusters of any shape shown in Figure 2.

The main function of the package is `rflexscan`, which takes the following arguments:

```
R> rflexscan(x, y, lat, lon, name, observed, expected, population, nb,
+    clustersize = 15, radius = 6370, stattype = "ORIGINAL",
+    scanmethod = "FLEXIBLE", ralpha = 0.2, simcount = 999,
+    rantype = "MULTINOMIAL", comments = "", verbose = FALSE)
```

Centroid coordinates for each region should be specified by Cartesian coordinates using arguments `x` and `y` or by latitudes and longitudes using arguments `lat` and `lon`. `name` specifies identifiers for each region and `observed` specifies the observed number of cases. For the Poisson model described in Section 2, the expected number of cases under the null hypothesis should be specified using `expected`. In addition, for the Binomial model (Kulldorff 1997), the background population at risk in each area should be specified using `population`. These arguments should be specified by vectors that have length m. The i-th element of the vector represents the data of region i.

`nb` is a list of neighbors or an adjacency matrix that expresses a structure of the connection between regions. When a list is specified by users, it should contain m integer vectors, where

the i-th vector contains either the indices in the range from 1 to m of the neighbors of region i, or `as.integer(0)` to signal no neighbors (see Section 4.1 for details). When users specify an adjacency matrix A, it should be a symmetric $m \times m$ matrix with zeros on its diagonal. Its element A_{ij} is one when region i and j ($i \neq j$) are connected, and zero when there is no connection.

In addition to the above necessary input data, users can specify the following parameters to control functionality.

- `clustersize`: The number of maximum spatial cluster size to scan (K), i.e., the maximum number of regions included in the detected cluster.

- `radius`: Radius of the earth in kilometers to calculate a distance between two sets of latitude and longitude. It is approximately 6370 km in Japan. This is deprecated. The distance calculated using this parameter is not accurate. This feature is implemented to maintain compatibility with **FleXScan**. It is recommended to transform latitude and longitude onto the Cartesian coordinate system beforehand and use the `x` and `y` parameters that are projected coordinates.

- `stattype`: Statistic type to be used (case-insensitive). If `"ORIGINAL"` is specified, the likelihood ratio statistic by Kulldorff and Nagarwalla (1995) is used. If `"RESTRICTED"` is specified, the restricted likelihood ratio statistic by Tango (2008) is used with a preset parameter `ralpha` for restriction.

- `scanmethod`: Scanning method to be used (case-insensitive). If `"FLEXIBLE"` is specified, the flexible scan statistic by Tango and Takahashi (2005) is used. If `"CIRCULAR"` is specified, the circular scan statistic by Kulldorff (1997) is used.

- `ralpha`: The prespecified significance level for the individual region used for the restricted likelihood ratio statistic (α_1).

- `simcount`: The number of Monte Carlo replications to calculate a p value for the statistical test.

- `rantype`: The type of random number for Monte Carlo simulation (case-insensitive). If `"MULTINOMIAL"` is specified, the total number of cases in the whole area is fixed. This option can be chosen in either Poisson or Binomial model. If `"POISSON"` is specified, the total number of cases is not fixed. This option can be chosen only in the Poisson model.

- `comments`: Comments for the analysis which will be written in a log of processing.

- `verbose`: Print progress messages.

The return value of the `rflexscan` function is an object of '`rflexscan`' class that contains analysis results and parameters specified by the user. To output summaries of results and to visualize detected clusters using a graph representation, S3 methods `summary` and `plot` are available for this class, respectively. Also, the `choropleth` method can be used to make a choropleth map displaying detected clusters.

4. Examples

In this section, we demonstrate the feasibility of the **rflexscan** package through an application to the small-area cancer incidence dataset (Boscoe *et al.* 2016), which is a public-domain dataset containing data for 23 anatomic sites of cancer diagnosed in New York State, USA between 2005 and 2009 at the census block group level. Here, we use a dataset provided by the ESRI shapefile format that is freely available from the **SaTScan** (Kulldorff and Information Management Services, Inc. 2018) website at https://www.satscan.org/datasets/nyscancer/. The dataset contains the following information for each region as well as geometric information:

- Geographical identifier consisting of 12 digits (coded from the state, county, tract, and census block group) based on the 2010 census.
- Centroid coordinates by latitude and longitude.
- Number of diagnosed (observed) cases for specific cancer.
- Expected number of cases for specific cancer adjusted for sex and 5-year age groups up to 85+ years, using the 2010 census counts for New York State.

First, let us load the dataset into the R environment. Although we use the **rgdal** (Bivand, Keitt, and Rowlingson 2021) package here, other packages for spatial analysis such as the **sf** (Pebesma 2018) package can also be used.

```
R> library("rgdal")
R> nys <- readOGR("NYS_Cancer/NYSCancer_region.shp",
+    stringsAsFactors = FALSE)
```

For example, the observed and expected number of breast cancer cases (`OBREAST` and `EBREAST`) as well as geographic identifier (`DOHREGION`) and centroid coordinates (`LATITUDE` and `LONGITUDE`) are contained as follows:

```
R> head(nys@data[c("DOHREGION", "LATITUDE", "LONGITUDE",
+    "OBREAST", "EBREAST")])
```

```
     DOHREGION LATITUDE LONGITUDE OBREAST EBREAST
0 360010001001 42.66806 -73.73442       0 3.56308
1 360010001002 42.67386 -73.74066       2 3.32389
2 360010002001 42.66768 -73.75101       9 6.70407
3 360010002002 42.65996 -73.75472       3 2.44295
4 360010003001 42.68651 -73.80743       3 4.13941
5 360010003002 42.67033 -73.77373       1 4.81561
```

Here, we try to analyze the spatial count data of breast cancer cases in Manhattan. Because a prefix of the geographic identifiers for Manhattan is `"36061"` (`"36"` is a state code for New York State and `"061"` is a county code for Manhattan), we can easily extract data of Manhattan from the whole dataset as follows:

```
R> manhattan <- nys[startsWith(nys$DOHREGION, "36061"),]
```

Manhattan has 982 regions, which are shown in Figure 1. Also, we transform the longitude and latitude coordinates of each region to Cartesian coordinates (UTM zone 18N) as follows:

```
R> coord <- data.frame(x=manhattan$LONGITUDE, y=manhattan$LATITUDE)
R> coordinates(coord) <- c("x", "y")
R> proj4string(coord) <- proj4string(manhattan)
R> coord <- spTransform(coord, CRS("+init=epsg:32618"))
```

Note that some useful web sites are available to construct appropriate coordinate reference system objects. For example, **What UTM Zone am I in ?** (MangoMap Limited 2020) can be used to find UTM zone numbers and **epsg.io** (MapTiler Team 2019) can be used to find EPSG codes.

Although the `rflexscan` method has a function to calculate the distance between two sets of latitude (`lat`) and longitude (`lon`), it is deprecated because of accuracy issues. This feature is implemented to maintain compatibility with **FleXScan**. We recommend transforming latitude and longitude onto the Cartesian coordinate system beforehand and use the `x` and `y` parameters.

4.1. Basic usage

For performing spatial data analysis using the flexible scan statistics, a neighbors list or an adjacency matrix that expresses a connection status between regions should be constructed. Although it is desirable to determine connections based on the actual geographical and/or social factors, here we automatically construct the neighbors list from the `SpatialPolygonsDataFrame` object using the `poly2nb` function in the **spdep** (Bivand and Piras 2015; Bivand, Hauke, and Kossowski 2013) package as follows:

```
R> library("spdep")
R> nb <- poly2nb(manhattan, queen = T)
R> print(nb)

Neighbour list object:
Number of regions: 982
Number of nonzero links: 6614
Percentage nonzero weights: 0.6858691
Average number of links: 6.735234
1 region with no links:
6796
```

The `nb` object is a list of integer vectors that contain indices of neighboring regions:

```
R> head(nb)

[[1]]
[1]  2  3  4 11 12 13 30 31
```

Figure 3: Map of Manhattan.

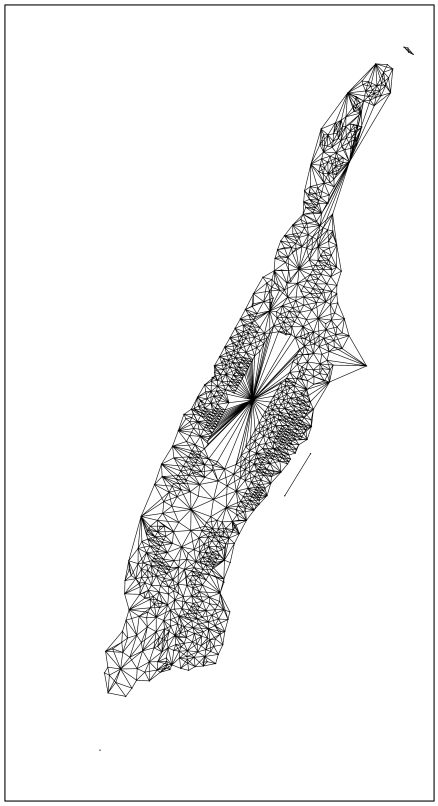


Figure 4: Connections via `poly2nb`.

```
[[2]]
[1]  1  3  4 12 13 14

[[3]]
 [1]  1  2  4  5  6  7 21 26 27 31

[[4]]
[1] 1 2 3 5 6 7

[[5]]
[1] 3 4 6

[[6]]
[1] 3 4 5 7
```

In this case, region 1 is connected with regions 2, 3, 4, 11, 12, 13, 30, and 31. Figure 4 shows the resulting neighbors list via a graph representation. The object contains 6,614 connections for the 982 regions.

Now, we are ready to perform the main analysis using the flexible scan statistic. This can be easily done with the following lines of code (which might take several seconds):

```
R> fls <- rflexscan(name = manhattan$DOHREGION,
+    x = coord$x, y = coord$y, nb = nb,
+    observed = manhattan$OBREAST, expected = manhattan$EBREAST)
```

The return value `fls` is an object of '`rflexscan`' class. The S3 method `print` for the class briefly shows the results:

```
R> print(fls)

Call:
rflexscan(x = coord$x, y = coord$y, name = manhattan$DOHREGION,
    observed = manhattan$OBREAST, expected = manhattan$EBREAST,
    nb = nb)

Most likely cluster (P-value: 0.001):
360610140001 360610142001 360610148011 360610148022 360610148024
360610148025 360610150012 360610150021 360610150022 360610150023
Number of secondary clusters: 14
```

The first part of the output displays the function call specified by the user. The next part displays identifiers of regions included in the MLC with the p value. In addition, the last part displays the number of secondary clusters.

The member variable `cluster` in the '`rflexscan`' object is a list of '`rflexscanCluster`' objects. The '`rflexscanCluster`' class contains properties of detected clusters. The following code display properties of the MLC w_1 using the `print` method for this class:

```
R> print(fls$cluster[[1]])

Areas included ...........:
360610140001 360610142001 360610148011 360610148022 360610148024
360610148025 360610150012 360610150021 360610150022 360610150023
Maximum distance .........:  616.8445
(areas: 360610140001 to 360610150021)
Number of cases ..........: 133
Expected number of cases .: 67.01175
Overall relative risk ....: 1.984727
Statistic value ..........: 25.53593
Monte Carlo rank .........: 1
P-value ..................: 0.001
```

The properties include areas in the cluster, the maximum distance between areas in the cluster, the number of cases, the expected number of cases, the relative risk of the disease, the likelihood ratio statistic, the rank obtained in the Monte Carlo simulation, and the p value.

Properties of secondary clusters can also be displayed with a similar code. For example, the following code displays properties of w_2:

```
R> print(fls$cluster[[2]])
```

```
Areas included ..........:
360610070001 360610072001 360610072005 360610072007 360610074001
360610080003 360610082002 360610082003 36061DOH0022
Maximum distance .........:  669.9477
(areas: 360610070001 to 360610082002)
Number of cases ..........: 79
Expected number of cases .: 39.91084
Overall relative risk ....: 1.979412
Statistic value ..........: 14.97593
Monte Carlo rank .........: 11
P-value ..................: 0.011
```

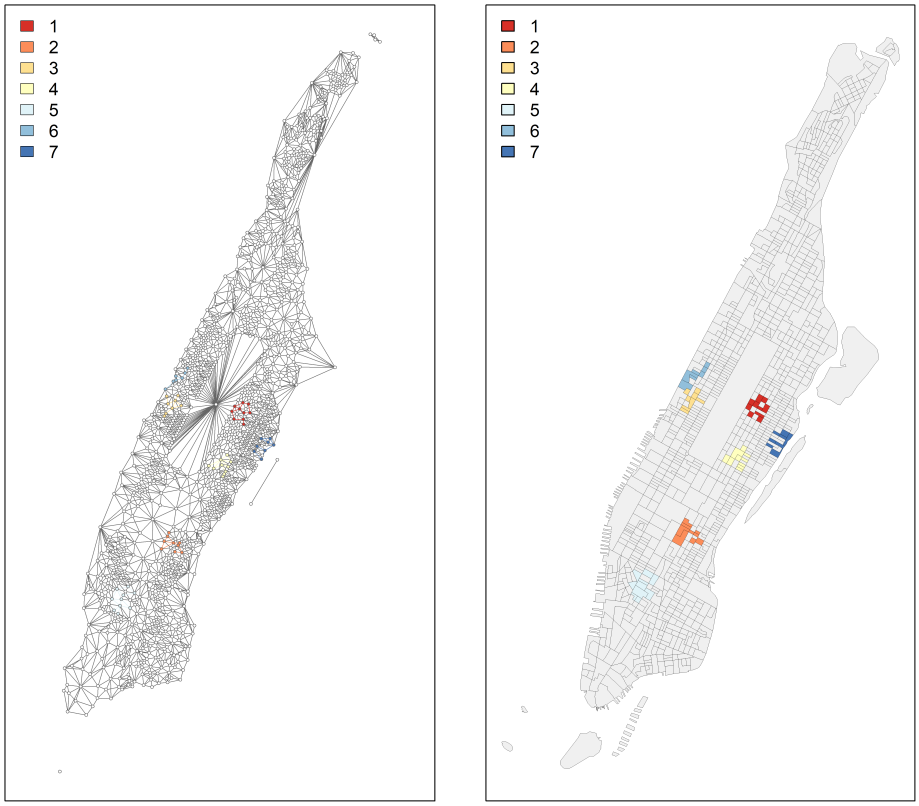
The summary of the analysis can be extracted using the S3 method `summary` for the '`rflexscan`' class:

```
R> summary(fls)

Call:
rflexscan(x = coord$x, y = coord$y, name = manhattan$DOHREGION,
    observed = manhattan$OBREAST, expected = manhattan$EBREAST,
    nb = nb)

Clusters:
    NumArea   MaxDist  Case  Expected     RR   Stats      P
1        10   616.844   133    67.012  1.985  25.536  0.001  ***
2         9   669.948    79    39.911  1.979  14.976  0.011    *
3         8   701.937    81    41.688  1.943  14.617  0.011    *
4        10   636.156    98    55.244  1.774  13.567  0.019    *
5        10   825.005    90    49.393  1.822  13.527  0.020    *
6         7   870.788    77    40.260  1.913  13.300  0.022    *
7         8   637.761   101    58.118  1.738  13.085  0.027    *
8         8   623.125    82    50.179  1.634   8.533  0.698
9         9   575.958    88    55.290  1.592   8.274  0.766
10        7   617.428    47    24.459  1.922   8.198  0.783
11        7   529.104    70    41.574  1.684   8.111  0.801
12        3   641.974    28    11.830  2.367   7.974  0.823
13        7  1145.401    69    42.682  1.617   6.881  0.977
14        9  1049.749    67    41.940  1.598   6.378  0.997
15        1     0.000    16     5.941  2.693   5.800  1.000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Limit length of cluster: 15
Number of areas: 982
Total cases: 6219
Coordinates: Cartesian
Model: POISSON
```



(a) Graph representation (b) Choropleth map

Figure 5: Significant clusters via the flexible scan statistic ($p < 0.05$).

```
Scanning method: FLEXIBLE
Statistic type: ORIGINAL
```

The first part is the function call as with the output of the `print` method for the '`rflexscan`' class. The next part labeled as `Clusters` displays a list of detected clusters. The first row of the list is the MLC, and others are secondary clusters. **rflexscan** reports clusters with p values calculated by the Monte Carlo test less than one. For each cluster, the number of areas (`NumArea`), the maximum distance between areas in the cluster (`MaxDist`), the observed number of cases (`Case`), the expected number of cases (`Expected`), the relative risk of disease (`RR`), the likelihood ratio statistic (`Stats`), and p value (`P`) are presented. Statistically significant clusters are marked; in this case, we detected seven significant clusters ($p < 0.05$). The final part displays some summary statistics and model parameters used.

Users can evaluate whether a disease is randomly distributed over space from the `Clusters` part. In this case, p value corresponding to the MLC is 0.001 (see the first row of the list), which implies breast cancer cases are clustered rather than randomly distributed over the study area. Further, p values are reported for the MLC and secondary clusters that enable us to evaluate which cluster is statistically significant.

Users can simply visualize the detected clusters via a graph representation using the S3 method `plot` for the '`rflexscan`' object (using color palettes from **RColorBrewer** (Neuwirth 2014) package) as follows:

```
R> library("RColorBrewer")
R> plot(fls, rank = 1:7, col = brewer.pal(7, "RdYlBu"))
R> box()
R> legend("topleft", legend = 1:7, fill = brewer.pal(7, "RdYlBu"), bty="n")
```

The result is shown in Figure 5 (a). The `rank` argument of the `plot` method specifies rankings of clusters to be displayed in the graph. Here, we highlighted the top 7 clusters that were statistically significant ($p < 0.05$). Furthermore, the `choropleth` method displays a choropleth map displaying detected clusters as follows:

```
R> choropleth(manhattan, fls, rank = 1:7, col = brewer.pal(7, "RdYlBu"))
R> legend("topleft", legend = 1:7, fill = brewer.pal(7, "RdYlBu"), bty="n")
```

The result is shown in Figure 5 (b). In a similar way to the `plot` method, users can specify clusters to be displayed (the top 7 clusters, in this case).

The MLC at the east of Central Park (colored by black) corresponds to a previously reported cluster with unusually high cancer incidence (Boscoe *et al.* 2016). However, the shape of the MLC is not circular, whereas the reported clusters determined based on the scan statistic of Kulldorff (1997) are all circular. The shapes of secondary clusters are also diverse, although they are located in reported block groups with high incidence (Boscoe *et al.* 2016).

4.2. Restricted likelihood ratio

The **rflexscan** package also implements the flexible scan statistic with the restricted likelihood ratio (Tango and Takahashi 2012). The following codes can be used to analyze breast cancer data with the significance level $\alpha_1 = 0.2$ for the individual region, the `stattype` argument specifies a likelihood ratio statistic to be used, and the `ralpha` specifies the significance level α_1:

```
R> fls2 <- rflexscan(name = manhattan$DOHREGION,
+    x = coord$x, y = coord$y, nb = nb,
+    observed = manhattan$OBREAST, expected = manhattan$EBREAST,
+    stattype = "RESTRICTED", ralpha = 0.2)
R> summary(fls2)

Call:
rflexscan(x = coord$x, y = coord$y, name = manhattan$DOHREGION,
    observed = manhattan$OBREAST, expected = manhattan$EBREAST,
    nb = nb, stattype = "RESTRICTED", ralpha = 0.2)

Clusters:
    NumArea  MaxDist  Case  Expected     RR   Stats      P
1        10  616.844   133    67.012  1.985  25.536  0.001  ***
2         9  669.948    79    39.911  1.979  14.976  0.007   **
3         5  560.999    59    27.135  2.174  14.044  0.016    *
4         8  602.440   103    60.412  1.705  12.515  0.038    *
5         7  661.019    74    40.871  1.811  10.890  0.110
```

```
6          4  408.005    50    24.380  2.051  10.346  0.156
7          6  505.077    60    31.549  1.902  10.183  0.177
8          7  529.104    70    41.574  1.684   8.111  0.603
9          3  641.974    28    11.830  2.367   7.974  0.639
10         5  542.966    54    30.107  1.794   7.702  0.715
11         5  503.001    63    37.739  1.669   7.074  0.842
12         7  728.530    50    29.207  1.712   6.123  0.971
13         1    0.000    16     5.941  2.693   5.800  0.991
14         4  831.273    35    18.767  1.865   5.602  0.997
15         2  286.998    14     4.926  2.842   5.555  0.997
16         4  438.215    40    22.634  1.767   5.435  0.999
17         4  338.163    46    27.621  1.665   5.111  1.000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Limit length of cluster: 15
Number of areas: 982
Total cases: 6219
Coordinates: Cartesian
Model: POISSON
Scanning method: FLEXIBLE
Statistic type: RESTRICTED
```

In this case, we detected four significant clusters ($p < 0.05$) and a suggestive cluster ($p < 0.1$). Also, Figure 6 (a) shows a choropleth map of significant clusters. As mentioned in Section 2.3, the distribution of the test statistic of clusters under the null hypothesis varies depending on the parameters used, and the resulting p value also varies. In this case, the MLC is the same as the one detected by the original flexible scan statistic, while the secondary clusters are slightly more compact than those of the original statistic. The restricted likelihood ratio eliminates neighboring regions with non-elevated risk of disease occurrence (Tango and Takahashi 2012) and avoids creating over-sized clusters that are oddly shaped with multiple narrow branches, i.e., the octopus effect (Costa, Assunção, and Kulldorff 2012; Duczmal and Assunção 2004).

The choice of α_1 varies depending on the situation and/or the user's specific consideration. Tango and Takahashi (2012) shows the following guidance regarding the choice of α_1 for a restricted likelihood ratio statistic of the nominal α level of 0.05: (1) α_1 between 0.10 and 0.20 to detect small clusters with a sharp increase in risk; (2) α_1 between 0.20 and 0.30 to detect small to middle-sized clusters with a moderate increase in risk; and (3) α_1 between 0.30 and 0.40 to detect larger clusters with a slight increase in risk. Tango (2008) further recommends $\alpha_1 = 0.20$ as a default.

4.3. Cluster size

Users can specify the maximum number K of nearest neighbors to be scanned using the `clustersize` argument of the `rflexscan` function (default is 15). For the flexible scan statistic with the original likelihood ratio (Tango and Takahashi 2005) given in Equation 2, we recommend $K \leq 30$ because of heavy computational load. Meanwhile, the flexible scan statistic with the restricted likelihood ratio (Tango and Takahashi 2012) given in Equation 3

allows us to consider larger clusters efficiently (see Section 5 for details).

For example, the following code performs a cluster detection with $K = 50$ using the flexible scan statistic with the restricted likelihood ratio:

```
R> system.time({
+    fls3 <- rflexscan(name = manhattan$DOHREGION,
+      x = coord$x, y = coord$y, nb = nb,
+      observed = manhattan$OBREAST, expected = manhattan$EBREAST,
+      stattype = "RESTRICTED", ralpha = 0.2,
+      clustersize = 50)
+  })

   user  system elapsed
   2.86    0.03    2.90
```

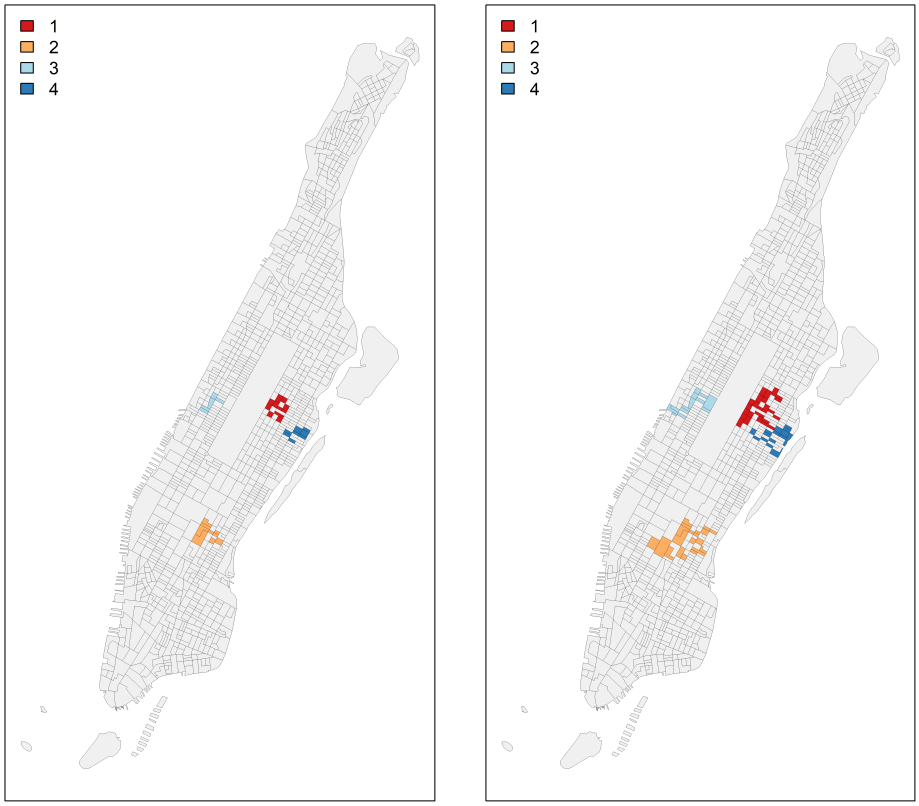
As described above, the analysis was completed within about 3 seconds on a laptop with an Intel Core i7-8565U CPU at 1.80GHz and 16GB RAM. The detected clusters are slightly different to those detected using the default parameter of $K = 15$:

```
R> summary(fls3)

Call:
rflexscan(x = coord$x, y = coord$y, name = manhattan$DOHREGION,
    observed = manhattan$OBREAST, expected = manhattan$EBREAST,
    nb = nb, clustersize = 50, stattype = "RESTRICTED", ralpha = 0.2)

Clusters:
    NumArea   MaxDist  Case  Expected     RR   Stats      P
1        23  1383.701   261   144.838  1.802  38.660  0.001  ***
2        20  1815.655   162    87.953  1.842  25.350  0.001  ***
3        13  1135.491   142    75.427  1.883  23.627  0.002   **
4        15  1070.199   170    97.614  1.742  22.356  0.004   **
5         9  1436.430    88    48.628  1.810  12.950  0.273
6        11  1451.106    81    45.647  1.774  11.203  0.475
7         8   900.217    83    47.170  1.760  11.176  0.477
8        10  1093.712    93    55.361  1.680  10.718  0.544
9         4   408.005    50    24.380  2.051  10.346  0.606
10        8   904.880    80    46.133  1.734  10.266  0.616
11        7   625.847    69    38.062  1.813  10.187  0.637
12        1     0.000    16     5.941  2.693   5.800  0.999
13        2   286.998    14     4.926  2.842   5.555  0.999
14        5   510.609    37    21.717  1.704   4.450  1.000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Limit length of cluster: 50
Number of areas: 982
```



(a) $K = 15$ (b) $K = 50$

Figure 6: Significant clusters via the flexible scan statistic with the restricted likelihood ratio ($p < 0.05$).

```
Total cases: 6219
Coordinates: Cartesian
Model: POISSON
Scanning method: FLEXIBLE
Statistic type: RESTRICTED
```

The MLC has 23 regions that are not considered in the previous analysis. These clusters are irregularly shaped (see Figure 6 (b)), and thus will not be detected by the circular scan statistic.

As shown in Figure 6, different values of K produce different results. If prior information is not available, set K to as large a value as possible. Package **rflexscan** will then look for clusters of both small and large sizes without any pre-selection bias in terms of the cluster size. However, K should be set to no more than half of the number of regions m. A cluster of larger size would indicate areas of exceptionally low risks outside the window rather than an area of exceptionally high risks within the window. Besides, Tango and Takahashi (2005) pointed that it seems to be unlikely that the size of the true cluster would be larger than 10%–15% of the total number of regions.

4.4. Monte Carlo replications

The `simcount` argument specifies the number of Monte Carlo replications to calculate p values of detected clusters (default value is 999). For example, the following codes perform a similar analysis as Section 4.3, with 9,999 replications.

```
R> system.time({
+    fls4<- rflexscan(name = manhattan$DOHREGION,
+      x = coord$x, y = coord$y, nb = nb,
+      observed = manhattan$OBREAST, expected = manhattan$EBREAST,
+      stattype = "RESTRICTED", ralpha = 0.2,
+      clustersize = 50,
+      simcount = 9999)
+  })

   user  system elapsed
  17.96    0.17   18.12
```

Note that the computation time increases as the number of replications increases. The resulting '`rflexscan`' object includes p values estimated more precisely than with the default value as follows:

```
R> summary(fls4)

Call:
rflexscan(x = coord$x, y = coord$y, name = manhattan$DOHREGION,
    observed = manhattan$OBREAST, expected = manhattan$EBREAST,
    nb = nb, clustersize = 50, stattype = "RESTRICTED", ralpha = 0.2,
    simcount = 9999)

Clusters:
    NumArea   MaxDist  Case  Expected     RR   Stats       P
1        23  1383.701   261   144.838  1.802  38.660  0.0001  ***
2        20  1815.655   162    87.953  1.842  25.350  0.0003  ***
3        13  1135.491   142    75.427  1.883  23.627  0.0016   **
4        15  1070.199   170    97.614  1.742  22.356  0.0034   **
5         9  1436.430    88    48.628  1.810  12.950  0.2874
6        11  1451.106    81    45.647  1.774  11.203  0.5067
7         8   900.217    83    47.170  1.760  11.176  0.5104
8        10  1093.712    93    55.361  1.680  10.718  0.5785
9         4   408.005    50    24.380  2.051  10.346  0.6399
10        8   904.880    80    46.133  1.734  10.266  0.6518
11        7   625.847    69    38.062  1.813  10.187  0.6642
12        1     0.000    16     5.941  2.693   5.800  0.9989
13        2   286.998    14     4.926  2.842   5.555  0.9995
14        5   510.609    37    21.717  1.704   4.450  0.9999
15        1     0.000    16     6.999  2.286   4.234  1.0000
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Limit length of cluster: 50
Number of areas: 982
Total cases: 6219
Coordinates: Cartesian
Model: POISSON
Scanning method: FLEXIBLE
Statistic type: RESTRICTED
```

4.5. Circular scan statistic

In addition to the flexible scan statistic, the **rflexscan** package implements the original spatial scan statistic proposed by Kulldorff and Nagarwalla (1995) and Kulldorff (1997) that considers only the set of circular-shaped windows $\mathcal{W}_c$. In this case, the `scanmethod` argument should be specified as `"CIRCULAR"` as follows:

```
R> fls5 <- rflexscan(name = manhattan$DOHREGION,
+    lat = manhattan$LATITUDE, lon = manhattan$LONGITUDE, nb = nb,
+    observed = manhattan$OBREAST, expected = manhattan$EBREAST,
+    scanmethod = "CIRCULAR")
R> summary(fls5)

Call:
rflexscan(lat = manhattan$LATITUDE, lon = manhattan$LONGITUDE,
    name = manhattan$DOHREGION, observed = manhattan$OBREAST,
    expected = manhattan$EBREAST, nb = nb, scanmethod = "CIRCULAR")

Clusters:
    NumArea  MaxDist  Case  Expected     RR   Stats      P
1        14    0.650   160    92.917  1.722  20.240  0.001  ***
2        13    0.621    92    52.237  1.761  12.437  0.004   **
3        15    0.949   157   107.184  1.465  10.314  0.017    *
4        15    0.687   134    89.173  1.503   9.910  0.027    *
5        14    0.823   135    90.206  1.497   9.799  0.027    *
6         7    0.574    60    33.006  1.818   8.924  0.059    .
7         6    0.349    70    41.543  1.685   8.132  0.124
8        12    0.622   103    71.091  1.449   6.363  0.498
9         5    0.432    49    28.786  1.702   5.884  0.657
10       15    0.968   109    77.348  1.409   5.820  0.680
11       11    0.775   104    73.966  1.406   5.481  0.780
12       15    0.576   114    85.053  1.340   4.515  0.980
13        1    0.000    16     6.999  2.286   4.234  0.996
14        5    0.566    44    27.496  1.600   4.205  0.996
15        1    0.000     9     3.116  2.888   3.664  0.999
```

```
16        1    0.000    10     3.829  2.612   3.432  1.000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Limit length of cluster: 15
Number of areas: 982
Total cases: 6219
Coordinates: Latitude/Longitude
Model: POISSON
Scanning method: CIRCULAR
Statistic type: ORIGINAL
```

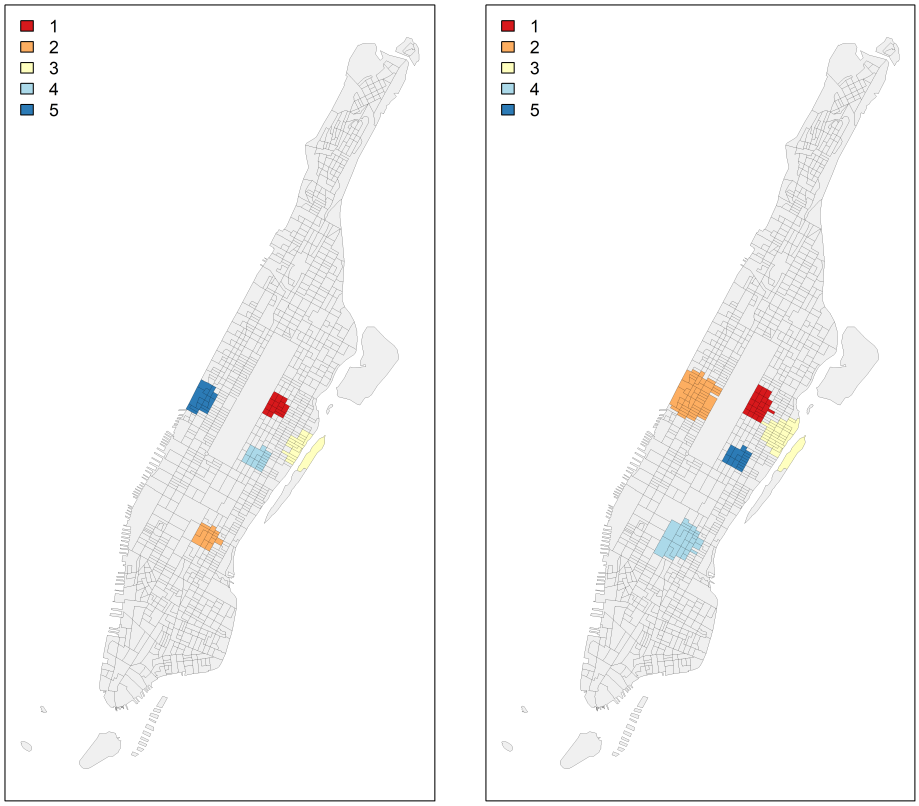
In this case, we detected five significant clusters ($p < 0.05$) and a suggestive cluster ($p < 0.1$). As shown in Figure 7 (a), all detected clusters are circular and different from those detected via the flexible scan statistic.

Similar with the flexible scan statistic, users can specify the maximum number K of nearest neighbors to be scanned using the `clustersize` argument:

```
R> fls6 <- rflexscan(name = manhattan$DOHREGION,
+    lat = manhattan$LATITUDE, lon = manhattan$LONGITUDE, nb = nb,
+    observed = manhattan$OBREAST, expected = manhattan$EBREAST,
+    scanmethod = "CIRCULAR",
+    clustersize = 50)
R> summary(fls6)

Call:
rflexscan(lat = manhattan$LATITUDE, lon = manhattan$LONGITUDE,
    name = manhattan$DOHREGION, observed = manhattan$OBREAST,
    expected = manhattan$EBREAST, nb = nb, clustersize = 50,
    scanmethod = "CIRCULAR")

Clusters:
    NumArea  MaxDist  Case  Expected     RR   Stats      P
1        21    0.903   222   134.981  1.645  24.061  0.001  ***
2        45    1.413   366   261.185  1.401  19.602  0.001  ***
3        30    1.092   284   203.524  1.395  14.691  0.002   **
4        23    1.123   147    92.212  1.594  14.010  0.003   **
5        15    0.687   134    89.173  1.503   9.910  0.034    *
6         7    0.574    60    33.006  1.818   8.924  0.083    .
7        11    0.775   104    73.966  1.406   5.481  0.839
8        24    1.079   149   113.734  1.310   5.079  0.925
9         1    0.000    16     6.999  2.286   4.234  0.995
10        5    0.566    44    27.496  1.600   4.205  0.995
11        9    0.476    85    61.681  1.378   3.983  0.999
12        6    0.328    52    34.888  1.491   3.665  1.000
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



(a) $K = 15$ (b) $K = 50$

Figure 7: Significant clusters via the circular scan statistic ($p < 0.05$).

```
Limit length of cluster: 50
Number of areas: 982
Total cases: 6219
Coordinates: Latitude/Longitude
Model: POISSON
Scanning method: CIRCULAR
Statistic type: ORIGINAL
```

Detected clusters are shown in Figure 7 (b).

Users might hesitate in deciding whether it is better to use the scan method `"FLEXIBLE"` or `"CIRCULAR"`. Although the best choice depends on the situation and/or the user's specific consideration, the power characteristics of each method might be helpful. The circular spatial scan statistic shows better power for detecting circular-shaped clusters, but it has zero power for accurately detecting regions as a single non-circular cluster without any false positive regions or false negative regions, while the flexible scan statistic shows better powers for detecting non-circular clusters. Users can choose a more powerful method according to the expectation of the shape of clusters (based on the shape of regions, for example). Performance evaluations of the flexible scan statistic have been conducted in Tango and Takahashi (2005) and Tango and Takahashi (2012). Package **rflexscan** gives the same output as presented in

these studies, resulting in the same performance for type I error rate and statistical power (and for sensitivity, specificity, and positive predictive value).

5. Computational load

Because numerous windows will be scanned in the flexible scan statistic for large clusters, the computational load is one of the main concerns in practical uses. The **rflexscan** package implements an efficient procedure (Tango and Takahashi 2005) used in the original **FleXScan** software through the **Rcpp** (Eddelbuettel and François 2011) package. Note that the procedure correctly detects the MLC and secondary clusters considering all possible windows, although unnecessary cluster candidates can be pruned during the scanning. Furthermore, the package also implements the flexible scan statistic with the restricted likelihood ratio (Tango and Takahashi 2012), allowing us to consider large clusters more efficiently.

As a demonstration, we conducted benchmarks using the breast cancer data in Manhattan described in Section 4. Again, Manhattan has 982 regions, and there are 6,614 connections. We measured computation times of the `rflexscan` method varying the parameter K, the maximum size of clusters to be considered. Other parameters were set to default values, the number of Monte Carlo replication was 999, and the significance level α_1 used for the restricted likelihood ratio was 0.2. Also, we conducted the same analysis using existing R packages **scanstatistics** (Allévius 2018) and **smerc** (French 2021).

Figure 8 shows the measured computation times of each procedure for each value of K. The computation times of **scanstatistics** and **smerc** increase exponentially as K increases because the package enumerates all possible windows without pruning of unnecessary candidates to search the MLC. The `rflexscan` method with the original likelihood ratio takes comparably lower time than these packages because of the pruning procedure; however, its computation time also increases exponentially. In addition, **smerc** and **scanstatistics** required larger amount of memory since these packages constructed a list of the complete set of windows W_f. The size of the list of windows increases exponentially with respect to the maximum size of the cluster (K), and therefore, these packages resulted in out of memory for a large K. **rflexscan** does not construct the list of the complete set of windows and then searches for MLCs, but instead sequentially scans windows using depth-first search to find the MLC avoiding out of memory. Due to this reason, **rflexscan** does not have a method implemented in **smerc** and **scanstatistics** that returns a list of the complete set of windows.

Meanwhile, a marked improvement was achieved by using the restricted likelihood ratio that eliminates neighboring regions with non-elevated risk. The `rflexscan` method required an almost constant calculation time of <5 seconds regardless of the setting of K in this case. Note that the effect of the restriction depends on the choice of α_1. Longer computation time will be required when α_1 is increased; setting $\alpha_1 = 1$ is equivalent to the use of the original likelihood ratio in Equation 2. To illustrate this point, we conducted another benchmark varying the threshold parameter α_1. The maximum size of clusters K was 50 to have no impact on the computational burden, and the number of Monte Carlo replication was 999. Also, we conducted the same analysis using `rflex.test` method in **smerc** (French 2021) that implements the restricted likelihood ratio statistic.

Figure 9 shows the measured computation times of each procedure as a function of α_1. Since the size of the clusters is primarily limited by the value of α_1, the computation time of

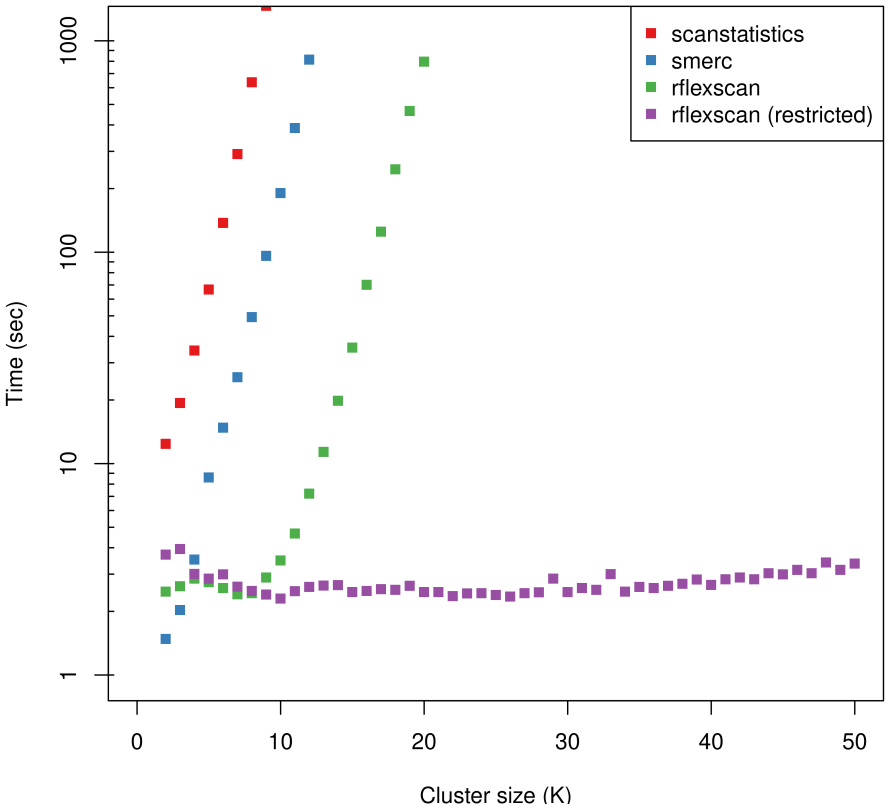


Figure 8: Computation time of each method for the flexible scan statistic ($\alpha_1 = 0.2$ and 999 Monte Carlo replications).

rflexscan was less than 3 seconds for $\alpha_1 \leq 0.2$ while it increased exponentially as α_1 increased. `rflex.test` in **smerc** required more time and was unable to perform the analysis for $\alpha_1 > 0.17$ because memory usage increased exponentially as α_1 increased.

In addition, we conducted the same analysis using **rsatscan** (Kleinman 2015) with **SaTScan** version 9.6 and measured computation times. We used a standard setting for the maximum cluster size, which specifies the upper limit of cluster size as 50% of the population at risk.

For detecting circular-shaped clusters, **rsatscan** took about 3 seconds as described below.

```
R> system.time({satscan("NYS_Cancer/", "breast_circular")})

   user  system elapsed
   0.08    0.03    3.05
```

In contrast, for detecting elliptically-shaped windows, it took more computation time because of the increase in the number of candidate windows.

```
R> system.time({satscan("NYS_Cancer/", "breast_elliptic")})
```

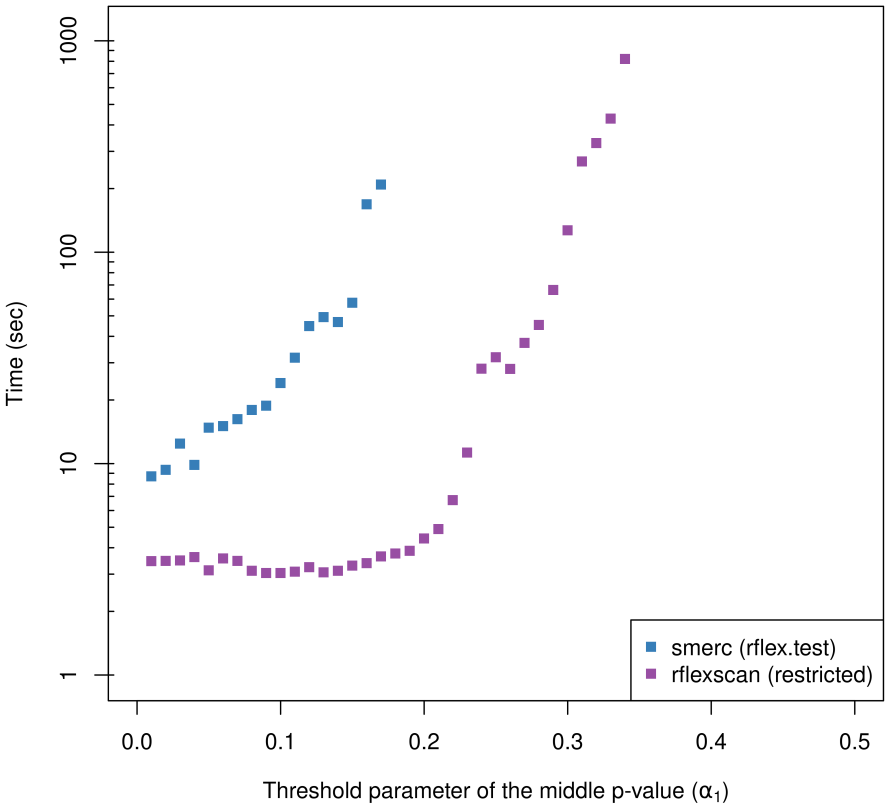


Figure 9: Computation time of each method for the flexible scan statistic ($K = 50$ and 999 Monte Carlo replications).

```
user  system elapsed
0.04    0.02   52.31
```

With the standard setting for the maximum cluster size, **rsatscan** looks for clusters of both small and large sizes exhaustively. Nevertheless, it is highly optimized and is able to detect clusters very fast as described above. However, note that is has a limitation on statistical power to detect non-circular or non-elliptic clusters. Although **rflexscan** tends to take more computation time than **rsatscan**, it can detect arbitrarily-shaped clusters.

6. Conclusions

We presented the **rflexscan** package that was designed for analyzing spatial count data using the flexible scan statistic. The package is designed for any of the following interrelated purposes: to evaluate whether reported spatial disease clusters are statistically significant, to test whether a disease is randomly distributed over space, and performing geographical surveillance of disease to detect areas of significantly high incidence and prevalence. It is implemented in R, and no external program has to be installed. By combining **rflexscan** with many useful R packages such as **rgdal** (Bivand *et al.* 2021), **sf** (Pebesma 2018) or **spdep** (Bivand

and Piras 2015; Bivand *et al.* 2013), users will be able to perform analysis and visualization easily and rapidly.

Some points should be noted for practical uses. As demonstrated, the clusters detected vary depending on the scan statistics and the control parameters used in constructing the window, and the test statistics and the p values will also vary. We recommend checking the robustness of your results using various statistic and control parameters. To check the robustness, users should conduct sensitivity analysis varying values of K and α_1. When the results of the hypothesis testing are same on any parameter settings, i.e., p value of the MLC are similar, the result will be robust. Users also need to make sure that the clusters are detected in the same location. As mentioned earlier, some guidelines for choosing values of K and α_1 exist. We recommend to perform sensitivity analysis within these ranges. Further, the spatial count data and the geographical information to be input should be constructed appropriately. For example, although we automatically constructed the connections between regions using the `poly2nb` method for demonstration in the paper, the actual geographical and/or social factors in real-world practice should be carefully considered. However, owing to the presence of obstacles that separate the districts, such as mountains and rivers, it may not be appropriate to treat them as a single cluster, even if the polygons are in contact with each other. There are also areas where polygons are defined but not actually inhabited (e.g., Central Park in Manhattan). Therefore, simply determining a connection from a shapefile can be problematic. The use of inappropriate data and parameters will produce strange results and will lead you to false decisions.

Besides, the use of flexible scan statistic has some disadvantages such as the need to specify K and α_1 in advance, and the high computation time when the method is set to search clusters with a numerous areas. An alternative approach to detect areas of unusually high risk is to detect areas with high exceedance probabilities, i.e., areas that have a high probability that the relative risk exceeds a given threshold (Moraga 2019). When computing exceedance probabilities, the user will obtain the probability that the region exceeds a given threshold. If the probability is small, say less than 0.05, they can highlight the risk of the region as unusual. This approach has the advantage that the highlighted areas can be of any shape, allow the assessment of the effects of covariates, and time is not a restriction when numerous areas are detected. In contrast, the advantage of the spatial scan statistics is that it can evaluate the significance of clusters via statistical hypothesis testing, i.e., it can calculate p value corresponding to the detected clusters.

Currently, the **rflexscan** package only implements methods for performing purely spatial analysis, while the **SaTScan** software and **scanstatistics** allow space-time analysis. The flexibly shaped space-time scan statistic (Takahashi, Kulldorff, Tango, and Yih 2008) will be included in future versions.

Computational details

The analyses were performed on a laptop with an Intel Core i7-8565U CPU at 1.80GHz and 16GB RAM.

Acknowledgments

This work was supported by the Environment Research and Technology Development Fund (S-17) of the Ministry of the Environment, Japan. The authors thank the Institute of Statistical Mathematics for the facilities and the use of HPE SGI 8600.

References

Allévius B (2018). "**scanstatistics**: Space-Time Anomaly Detection Using Scan Statistics." *Journal of Open Source Software*, **3**(25), 515. doi:10.21105/joss.00515.

Bivand R, Hauke J, Kossowski T (2013). "Computing the Jacobian in Gaussian Spatial Autoregressive Models: An Illustrated Comparison of Available Methods." *Geographical Analysis*, **45**(2), 150–179. doi:10.1111/gean.12008.

Bivand R, Keitt T, Rowlingson B (2021). ***rgdal**: Bindings for the 'Geospatial' Data Abstraction Library*. R package version 1.5-27, URL https://CRAN.R-project.org/package=rgdal.

Bivand R, Piras G (2015). "Comparing Implementations of Estimation Methods for Spatial Econometrics." *Journal of Statistical Software*, **63**(18), 1–36. doi:10.18637/jss.v063.i18.

Boscoe FP, Talbot TO, Kulldorff M (2016). "Public Domain Small-Area Cancer Incidence Data for New York State, 2005-2009." *Geospatial Health*, **11**(1), 2005–2009. doi:10.4081/gh.2016.304.

Costa MA, Assunção RM, Kulldorff M (2012). "Constrained Spanning Tree Algorithms for Irregularly-Shaped Spatial Clustering." *Computational Statistics & Data Analysis*, **56**(6), 1771–1783. doi:10.1016/j.csda.2011.11.001.

Du Z, Hao Y (2021). ***FlexScan**: Flexible Scan Statistics*. R package version 0.2.1, URL https://CRAN.R-project.org/package=FlexScan.

Duczmal L, Assunção R (2004). "A Simulated Annealing Strategy for the Detection of Spatial Clusters of Irregular Shape." *Computational Statistics & Data Analysis*, **45**(2), 269–286. doi:10.1016/s0167-9473(02)00302-x.

Dwass M (1957). "Modified Randomization Tests for Nonparametric Hypotheses." *The Annals of Mathematical Statistics*, **28**(1), 181–187.

Eddelbuettel D, François R (2011). "**Rcpp**: Seamless R and C++ Integration." *Journal of Statistical Software*, **40**(8), 1–18. doi:10.18637/jss.v040.i08.

French J (2021). ***smerc**: Statistical Methods for Regional Counts*. R package version 1.5, URL https://CRAN.R-project.org/package=smerc.

Gómez-Rubio V, Moraga P, Molitor J, Rowlingson B (2019). "**DClusterm**: Model-Based Detection of Disease Clusters." *Journal of Statistical Software*, **90**(14), 1–26. doi:10.18637/jss.v090.i14.

Kim AY, Wakefield J (2021). ***SpatialEpi**: Methods and Data for Spatial Epidemiology*. R package version 1.2.5, URL https://CRAN.R-project.org/package=SpatialEpi.

Kleinman K (2015). ***rsatscan**: Tools, Classes, and Methods for Interfacing with **SaTScan** Stand-Alone Software*. R package version 0.3.9200, URL https://CRAN.R-project.org/package=rsatscan.

Kulldorff M (1997). "A Spatial Scan Statistic." *Communications in Statistics – Theory and Methods*, **26**(6), 1481–1496. doi:10.1080/03610929708831995.

Kulldorff M (2006). "Tests of Spatial Randomness Adjusted for an Inhomogeneity: A General Framework." *Journal of the American Statistical Association*, **101**(475), 1289–1305. doi:10.1198/016214506000000618.

Kulldorff M, Information Management Services, Inc (2018). ***SaTScan** V9.6: Software for the Spatial and Space-Time Scan Statistics*. URL https://www.satscan.org/.

Kulldorff M, Nagarwalla N (1995). "Spatial Disease Clusters: Detection and Inference." *Statistics in Medicine*, **14**(8), 799–810. doi:10.1002/sim.4780140809.

MangoMap Limited (2020). "**What UTM Zone Am I in?**" URL https://mangomap.com/robertyoung/maps/69585/what-utm-zone-am-i-in-#.

MapTiler Team (2019). "**epsg.io**." URL https://epsg.io/.

Moraga P (2017). ***SpatialEpiApp**: A Shiny Web Application for the Analysis of Spatial and Spatio-Temporal Disease Data*. R package version 0.3, URL https://CRAN.R-project.org/package=SpatialEpiApp.

Moraga P (2019). *Geospatial Health Data: Modeling and Visualization with R-INLA and Shiny*. Chapman & Hall/CRC.

Neuwirth E (2014). ***RColorBrewer**: **ColorBrewer** Palettes*. R package version 1.1-2, URL https://CRAN.R-project.org/package=RColorBrewer.

Otani T, Takahashi K (2021). ***rflexscan**: The Flexible Spatial Scan Statistic*. R package version 1.0.0, URL https://CRAN.R-project.org/package=rflexscan.

Pebesma E (2018). "Simple Features for R: Standardized Support for Spatial Vector Data." *The R Journal*, **10**(1), 439–446. doi:10.32614/RJ-2018-009.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Rogerson P, Yamada I (2008). *Statistical Detection and Surveillance of Geographic Clusters*. Chapman & Hall/CRC. doi:10.1201/9781584889366.

Takahashi K, Kulldorff M, Tango T, Yih K (2008). "A Flexibly Shaped Space-Time Scan Statistic for Disease Outbreak Detection and Monitoring." *International Journal of Health Geographics*, **7**, 1–14. doi:10.1186/1476-072x-7-14.

Takahashi K, Yokoyama T, Tango T (2013). ***FleXScan** V3.1.2: Software for the Flexible Scan Statistics*. URL https://sites.google.com/site/flexscansoftware/home.

Tango T (2000). "A Test for Spatial Disease Clustering Adjusted for Multiple Testing." *Statistics in Medicine*, **19**(2), 191–204. doi:10.1002/(sici)1097-0258(20000130)19:2<191::aid-sim281>3.0.co;2-q.

Tango T (2008). "A Spatial Scan Statistic with a Restricted Likelihood Ratio." *Japanese Journal of Biometrics*, **29**(2), 75–95. doi:10.5691/jjb.29.75.

Tango T (2010). *Statistical Methods for Disease Clustering*. Statistics for Biology and Health. Springer-Verlag, New York. doi:10.1007/978-1-4419-1572-6.

Tango T, Takahashi K (2005). "A Flexibly Shaped Spatial Scan Statistic for Detecting Clusters." *International Journal of Health Geographics*, **4**(c), 11. doi:10.1186/1476-072x-4-11.

Tango T, Takahashi K (2012). "A Flexible Spatial Scan Statistic with a Restricted Likelihood Ratio for Detecting Disease Clusters." *Statistics in Medicine*, **31**(30), 4207–4218. doi:10.1002/sim.5478.

Waller LA, Gotway CA (2004). *Applied Spatial Statistics for Public Health Data*. John Wiley & Sons Series in Probability and Statistics. John Wiley & Sons, Hoboken. doi:10.1002/0471662682.

Zhang Z, Assunção R, Kulldorff M (2010). "Spatial Scan Statistics Adjusted for Multiple Clusters." *Journal of Probability and Statistics*, p. 11 pages.

Affiliation:

Takahiro Otani
Department of Public Health
Graduate School of Medical Sciences
Nagoya City University
1 Kawasumi, Mizuho-cho, Mizuho-ku, Nagoya 467-8601, Japan
E-mail: otani@med.nagoya-cu.ac.jp

Kunihiko Takahashi
Department of Biostatistics
M&D Data Science Center
Tokyo Medical and Dental University
1-5-45 Yushima, Bunkyo-ku, Tokyo 113-8510, Japan
E-mail: kunihikot.dsc@tmd.ac.jp

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
October 2021, Volume 99, Issue 13 *Submitted:* 2019-07-12
doi:10.18637/jss.v099.i13 *Accepted:* 2021-02-26