Chapter 13
Spatial and Spatiotemporal Patterns

13.1 Introduction

Spatial and spatiotemporal data analysis is of great importance in disease dynamics for a number of reasons such as looking for space-time clustering, hot-spot detection, characterizing invasion waves, and quantifying spatial synchrony. Spatial synchrony—the level of correlation in outbreak dynamics at different locations—is of particular significance to acute immunizing infections, because asynchrony may permit regional persistence of infections despite local chains-of-transmission breaking during post-epidemic troughs (Keeling et al. 2004). Conversely, spatial synchrony can exacerbate the economic and public health burden because the resulting regionalized outbreaks can overwhelm logistical capabilities as was evident in the early part of the 2013–2014 West African ebola outbreak.

Spatial statistics is also important in order to correct for the problem of spurious associations between incidence and environmental data because spatial autocorrelation violates the assumption of independence. We will discuss this in Sect. 15.2.

13.2 A Plant-Pathogen Case Study

Jennifer Koslow carried out an experiment with a foliar, nonsystemic rust (*Coleosporium asterum*) infecting the flat-top goldenrod (*Euthamia graminifolia*). The `gra` data present the severity of disease expression (`$score`, from 0 to 10) on host-plants planted within mesocosms (`$plot`) in an old field near Ithaca, NY, USA. The mesocosms were in a checkerboard grid with locations specified by

This chapter uses the following R-package: `ncf`.

O. N. Bjørnstad, *Epidemics*, Use R!, https://doi.org/10.1007/978-3-319-97487-3_13

coordinates `$xloc` and `$yloc`. Each mesocosm contained three focal *E. graminifolia* plants. The field also contained naturally occurring *E. graminifolia*, as well as several other hosts of the rust notably the Canada goldenrod (*Solidago canadensis*). Two different treatments, species composition (`$comp`, with three levels) and watering treatment (`$water`, with two levels), were applied to the mesocosms in a fully factorial design. Finally, to account for spatial variation across the field, there were four blocks with treatment combinations randomly assigned within each block.

We have to jitter the coordinates for some of the analyses because the three plants within each plot were not given separate coordinates. Figure 13.1 the spatial layout of the study. The vertical lines mark the blocks.

```
data(gra)
gra$jx=jitter(gra$xloc)
gra$jy=jitter(gra$yloc)
symbols(y=gra$xloc, x=gra$yloc, circles=gra$score,
     inches=0.1, xlab="y", ylab="x")
abline(v=47.5,col=2)
abline(v=97.5,col=2)
abline(v=147.5,col=2)
```

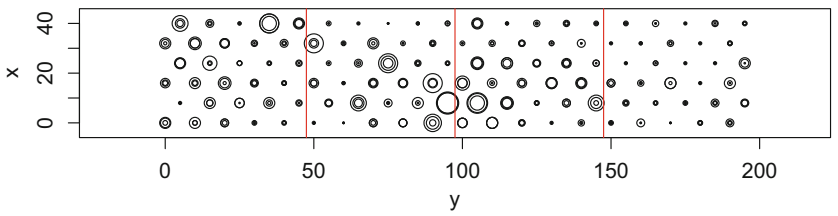

Fig. 13.1 Rust scores from Keslow's experiment

13.3 Spatial Autocorrelation

Spatial statistics is a very rich field. We will focus on a subset of methods that are more (or less) commonly used in disease ecology. Many of these involve the notion of spatial autocorrelation in one form or another. Legendre (1993) is a great introduction to the use of spatial autocorrelation in ecological studies in general. While all the methods we will be discussing—such as Mantel tests, parametric and nonparametric correlation functions, local indicators of spatial association, etc.—come in canned packages (this chapter uses the `ncf`-package), it is useful to spend a bit of time on the underlying ideas.

Many geostatistical methods to describe spatial pattern are focused on either spatial *variance* (Gary's C) or spatial *correlation* (Moran's I). We will discuss the family of "correlational" methods. We start off with considering the regular (Pearson's) product-moment correlation between two random variables, Z_1 and Z_2, which we denote by ρ_{12} and defined as:

$$\rho_{12} = \frac{(Z_1 - \mu_1)}{\sigma_1} \frac{(Z_2 - \mu_2)}{\sigma_2}$$

where μ's are expectations and σ's are standard deviations. *Auto*correlation has exactly the same definition and is used when the Z's are measurements of the same quantity (e.g., prevalence, incidence, presence/absence, etc.) at different spatial locations (or different times).

To calculate the autocorrelation we need to know (or have an estimate of) the values of the μ's and σ's. In the case of single snapshot spatial data we use the *marginal* mean and *marginal* standard deviation.[1] Let's explore using the graminifolia rust data (Fig. 13.1).

```
n = length(gra$score)
# marginal mean:
mu = mean(gra$score)
# marginal MLE  sd:
sig = sd(gra$score) * (n - 1)/n
```

The estimated "autocorrelation matrix" (`rho`) among all 360 plants is then[2]:

```
# rescale Zs
zscale = (gra$score - mu)/sig
# autocorrelation  matrix
rho = outer(zscale, zscale)
```

Note that these individual values are not constrained to be between -1 and 1. This is not a worry, though, because the various geostatistical methods we will be discussing involve relatively simple manipulations of this matrix. For several of the methods we also need some sort of spatial distance matrix. Most commonly used is the Euclidian distance for UTM coordinates and greater-circle distance for latitude/longitude coordinates. The Euclidean distance matrix among all 360 plants is:

```
dst = as.matrix(dist(gra[, c("xloc", "yloc")]))
```

[1] Note that the geostatistical methods usually use the Maximum Likelihood Estimator (MLE) of the sd rather than the Best Linear Unbiased estimator (BLUE): i.e., the denominator is n rather than $n-1$.

[2] The `outer`-function provides all pairwise products of two vectors.

To understand the different geostatistical methods we will consider the plot of the first 1000 pairs as a function of their spatial distance (Fig. 13.2). Plotting all the 64,620 pairs would clutter up the screen.

```
plot(dst[1:1000], rho[1:1000], ylab="Pairwise rho",
     xlab="Pairwise distance (m)")
```

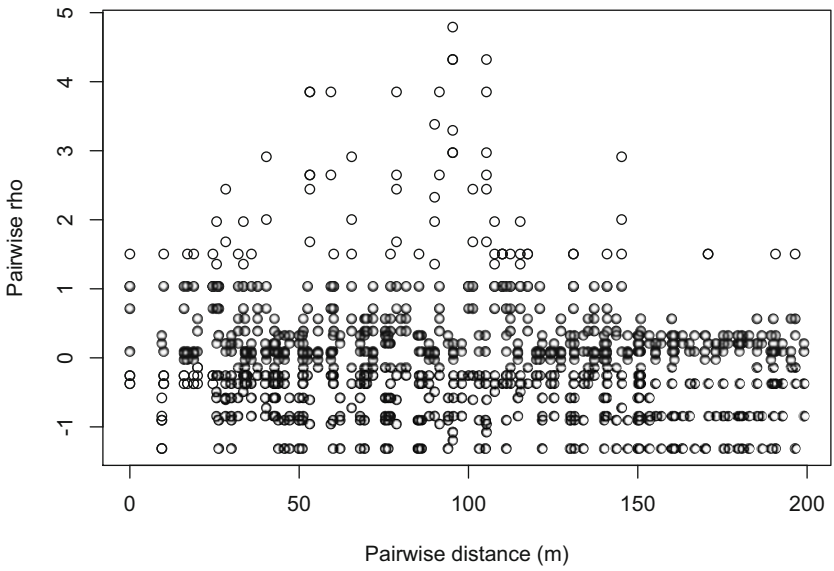

Fig. 13.2 Scatterplot of pairwise-ρ *versus* pairwise-distance

With this we are ready to conceptually understand many different geostatistical methods:

- **Mantel test:** An overall test for whether there is any significant relationship between the elements in the two matrices. This is essentially a test for significant correlation between ρ and distance.
- **Correlogram:** The most classic tool of testing how autocorrelation depends on distance without assuming any particular function—hack the x-axis into segments (given by specifying some distance increment) and calculate the average within each distance class.[3]
- **Parametric correlation functions:** Assume the relationship follows some parametric relationship—such as Exponential, Gaussian, or Spherical functions—and do the appropriate nonlinear regression of ρ on distance; Sect. 15.2 provides an example of such fitting via the `lme`-function of the `nlme`-library.

[3] The semivariogram is similar to the correlogram but instead of using the "autocorrelation similarity" measure it uses the "semivariance dissimilarity" measure: $(Z_i - Z_j)^2/2$.

- **Nonparametric correlation function:** Fit a "nonparametric regression" (usually a smoothing spline or a kernel smoother) to the relationship (Hall and Patil 1994). This also goes by the name of the "spline.correlogram" (Bjørnstad and Falck 2001).
- **LISA:** Local indicators of spatial association (Anselin 1995): A test for "hotspots." Specify a neighborhood size, and for each location calculate the average ρ with all the other locations that belongs to its neighborhood to find areas of significant above-average values.

There are a bunch of other named methods that are variations of these. Several of which are extensions to when there is multiple observations at each location (such as a time series), in which case it is natural to estimate the "autocorrelation matrix" using the regular correlation matrix. The "modified correlogram" of Koenig (1999) is the multivariate extension of the correlogram (e.g., Bjørnstad et al. 1999b). The "time-lagged spatial cross-correlation function" has been used to study waves of spread (see below and Sect. 11.7). Various directional versions allow the spatial correlation function to vary by cardinal direction (so-called anisotropic correlograms) to investigate directional patterns (e.g., Bjørnstad et al. 2002b).

13.4 Testing and Confidence Intervals

An important reason why specialized methods are needed for these analyses—despite most being conceptually simple—is because while the n original data-points may (or may not) be statistically independent, the n^2 numbers in the autocorrelation matrix is obviously very statistically not-independent and the interdependence is very intricate. None of the usual ways of testing for significance or generating confidence intervals are therefore applicable. Testing is usually done using permutation tests under the null-hypothesis of no spatial patterns. The correlogram (or Mantel test, or ...) of the real data should look no different than that of a random reallocation of observations to the spatial coordinates if the null hypothesis is true. Statistical significance is calculated by comparing the observed estimate to the distribution of estimates for, say, 999 different randomized data sets.[4] If the observed is more extreme than 950 (990) of the randomized we conclude that there is significant deviation from spatial randomness at a nominal 5%-level (1%-level). For some of the methods it is possible to generate *confidence intervals* using bootstrapping (resampling with replacement) (e.g., Bjørnstad and Falck 2001).

All the above methods are available in the `ncf`-package.

```
require(ncf)
```

[4] This produces a total of 1000 known possible outcomes; The 999 we randomly generated + the one nature provided.

13.5 Mantel Test

We continue using Keslow's data as a case study.

```
test1 = mantel.test(M1 = rho, M2 = dst)
```

```
test1

  ## $correlation
  ## [1] -0.04603662
  ##
  ## $p
  ## [1] 0.000999001
  ##
  ## $call
  ## [1] "mantel.test(M1 = rho, M2 = dst)"
  ##
  ## attr(,"class")
  ## [1] "Mantel"
```

We see that there is a significant negative association between similarity and distance. This is a crude tool but it does reveal that locations near each other tend to be more similar in disease status than those separated by a greater distance.

If we, instead of having two matrixes, have spatial coordinates and observations, the syntax is:

```
test = mantel.test(x = ..., y = ..., z = ...)
```

13.6 Correlograms

The correlogram shows how the autocorrelation is a function of distance (Fig. 13.3). The shape of the correlogram can indicate random *versus* diffusive *versus* clinal patterns. Legendre and Fortin (1989) provide probes for patterns using various visual characteristics of the correlogram.

```
test2=correlog(x=gra$xloc, y=gra$yloc, z=gra$score,
     increment=10)
plot(test2)
```

The first distance class is significantly positive, and the estimated distance to which the local positive distance decays to zero (the `x-intercept`) is 44 m, indicative of significant local similarity. There is further evidence of significantly *negative* autocorrelation at long distances suggestive of a gradient (Legendre and Fortin 1989) across the field (Fig. 13.3).

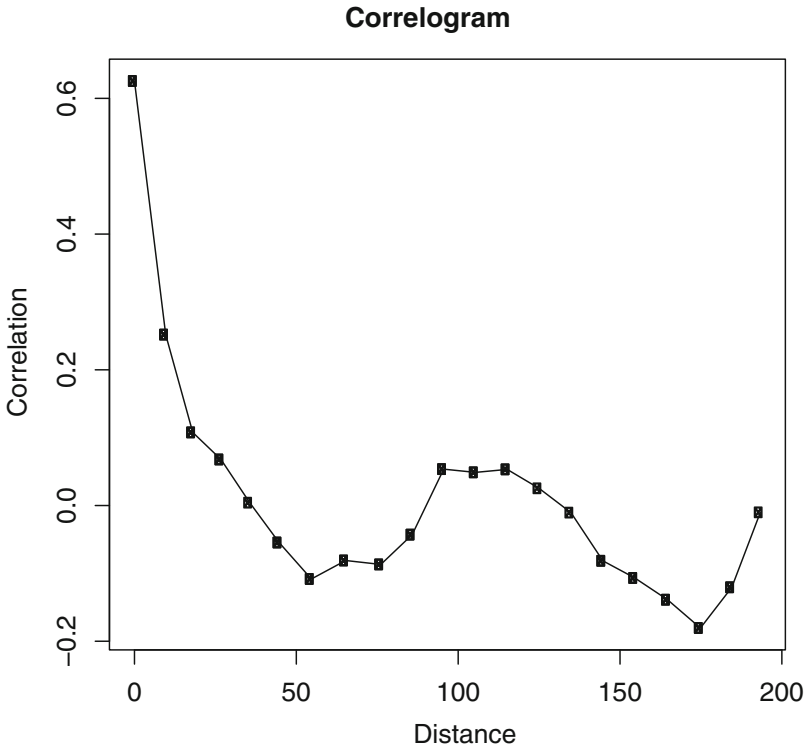

Fig. 13.3 The spatial correlogram of Keslow's rust data. Values that significantly deviate from that expected under the null hypothesis of complete spatial randomness are represented by filled black circles

13.7 Nonparametric Spatial Correlation Functions

We can get a bit finer resolution and confidence intervals for the underlying spatial correlation function using a nonparametric spatial covariance function (Hall and Patil 1994) as implemented in the spline correlogram (Bjørnstad and Falck 2001).

```
test3=spline.correlog(x=gra$xloc, y=gra$yloc,
   z=gra$score)
```

```
summary(test3)
```

```
## $call
## [1] "spline.correlog(x = gra$xloc, y = gra$yloc,
## z = gra$score)"
##
## $estimate
##                   x        e         y
## estimate 36.53433 5.981471 0.5824953
##
## $quantiles
##                  x         e           y
## 0      -1.805418  0.000000 -0.02511758
## 0.025 23.252053  0.000000  0.14654935
## 0.25   33.067324  0.000000  0.28755750
## 0.5    36.555305  1.316775  0.38985499
## 0.75   39.924299  5.880383  0.48984864
## 0.975 44.163650 11.783945  0.75428625
## 1      49.466042 14.797740  0.98590369
```

The spline correlogram returns a bunch of stuff—in fact all the summary statistics I thought might be of relevance in some previous spatial analyses. These are:

- estimates: a vector of benchmark statistics
- x: is the lowest value at which the function is = 0.[5]
- e: is the lowest value at which the function is = 1/e (i.e., the spatial scale parameter in the presence of exponential or Gaussian spatial correlation).
- y: is the extrapolated value at x = 0.
- quantiles: A matrix summarizing the quantiles in the bootstrap distributions of the benchmark statistics. The 2.5- and 97.5-percentiles represent the 95% confidence interval.

```
plot(test3)
```

Figuer 13.4 shows the estimated correlation function with its bootstrap 95% confidence intervals. The confidence intervals allows us to compare correlation functions for different data sets to test for significant differences (e.g., Bjørnstad et al. 1999a).

[5] If correlation is initially negative, the distance calculated appears as a negative measure. This may seem a little strange, but some locally inhibitory processes predict significant negative local auto- or cross-correlation (e.g., Seabloom et al. 2005).

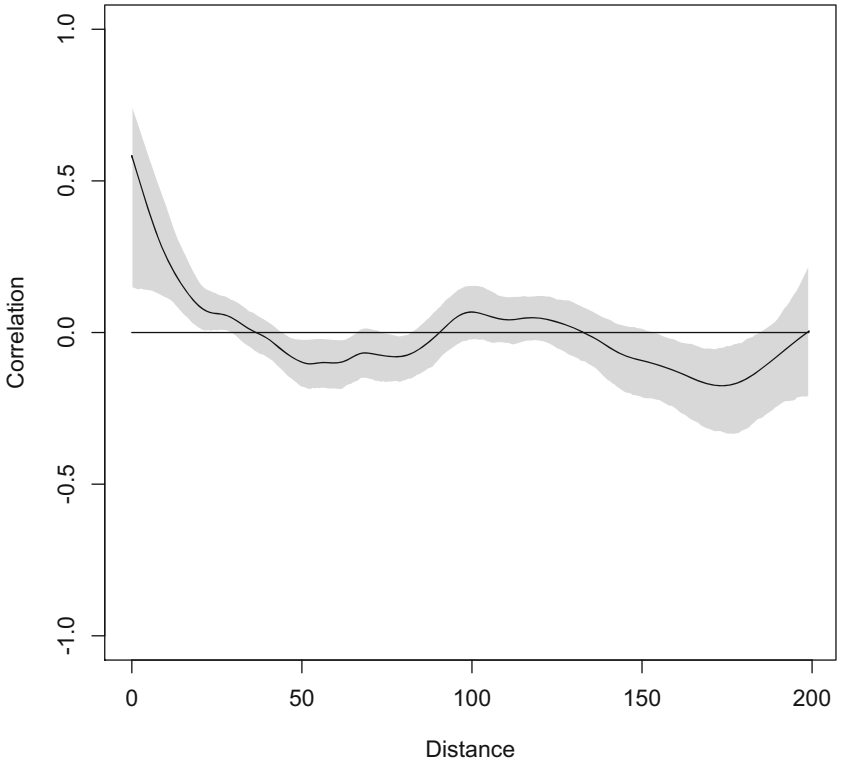

Fig. 13.4 The spline correlogram of Keslow's rust data. The outer lines represent the 95% bootstrap confidence interval

13.8 LISA

The previous methods average across all locations to study how similarity depends on distance. Local indicators of spatial association (Anselin 1995) quantify how similar observations are within neighborhoods of each observation—this can be used to test for *significant* spatial hot-/cold-spots of disease (Fig. 13.5). For this we have to define the radius of the neighborhood. Spatial dependence in the Koslow-data decay to zero at around 40 m (Fig. 13.4), so we use 20 m:

```
test4=lisa(x=gra$yloc, y=gra$xloc, z=gra$score,
    neigh=20)
```

```
plot(test4)
```

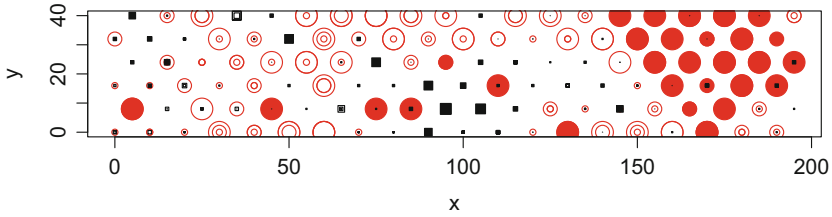

Fig. 13.5 LISA analysis of Koslow's rust data (with a 20 m neighborhood). Filled red circles are significant spatial hot-spots. Squares are cold-spots

Significant hot-spots show up as filled red circles and cold-spots as filled squares. The size of the symbols reflects how much the disease-score deviates from the mean.

13.9 Cross-Correlations

Janis Antonovics and his colleagues have done road-side surveys of antler smut disease counting number of healthy and diseased wild campions (*silene alba*) at the Mountain Lake Biological field station for more than 20 years (Antonovics 2004). The `silene2`-data contains the mean number of healthy (`$hmean`) and diseased (`$dmean`) individuals for each road segment, as well as latitude (`$lat`) and longitude (`$lon`) (Fig. 13.6).

```
data(silene2)
symbols(silene2$lon, silene2$lat, circles =
     sqrt(silene2$dmean), inches=.2, xlab="Longitude",
     ylab="Latitude")
```

Most geostatistical methods can be extended to consider spatial *cross*-correlation between different variables. We can use the silene data set to investigate if prevalence is spatially cross-correlated with abundance using the spline cross-correlogram (Fig. 13.7).

```
silene2$ab=silene2$dmean+silene2$hmean
silene2$prev=silene2$dmean/(silene2$dmean+silene2$hmean)
```

We square-root transform the abundance measure before analyses. There is significant positive cross-correlation within a 1 km range (95% CI: $\{0.6, 2.9\}$ km).

```
testcc=spline.correlog(x=silene2$lon, y=silene2$lat,
   z=silene2$prev, w=sqrt(silene2$ab),
   latlon=TRUE, na.rm=TRUE)
plot(testcc)
```

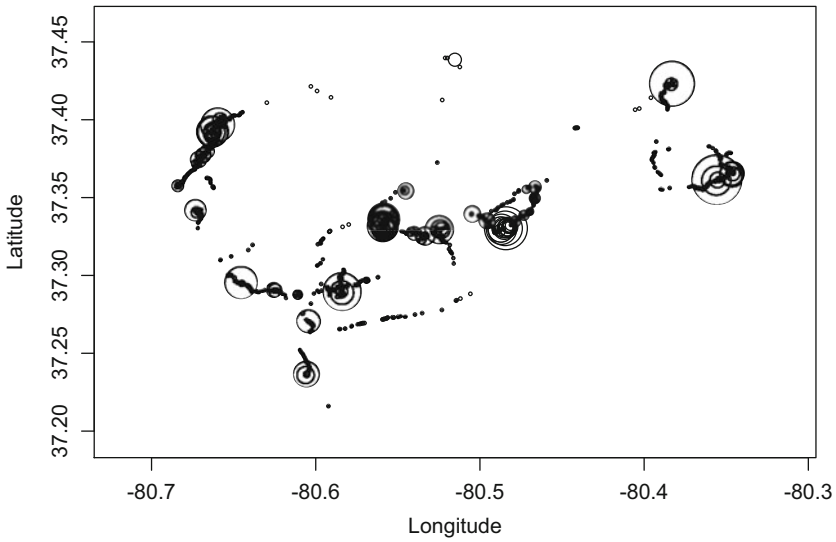

Fig. 13.6 Burden of antler smut on wild campion at Mt. Lake field station (Antonovics 2004)

We can use a spatial cross-correlogram (using 25 m distance increments) to study if presence/absence of rust is spatiotemporally cross-correlated between 1994 and 1995 in the filipendula data set we discussed in Sect. 11.2.

```
data(filip)
testcc2=correlog(x=filip$X, y=filip$Y, z=filip$y94,
     w=filip$y95, increment=25)
```

The local inter-year correlation (`corr0`) is 0.75 and the first cross-correlation is significantly positive with a cross-correlogram x-intercept of 148 m[6]:

```
testcc2$corr0

  ## [1] 0.7651124

testcc2$x.intercept

  ## (Intercept)
  ##     148.939
```

Locations heavily affected in 1994 were thus also heavily affected in 1995 (testifying to the importance of local contagion and/or habitat heterogeneity in infection risk). This is an example of a "time-lagged cross-correlogram" (e.g., Bjørnstad et al. 2002b).

[6] The spline cross-correlogram would give bootstrap confidence intervals on these quantities.

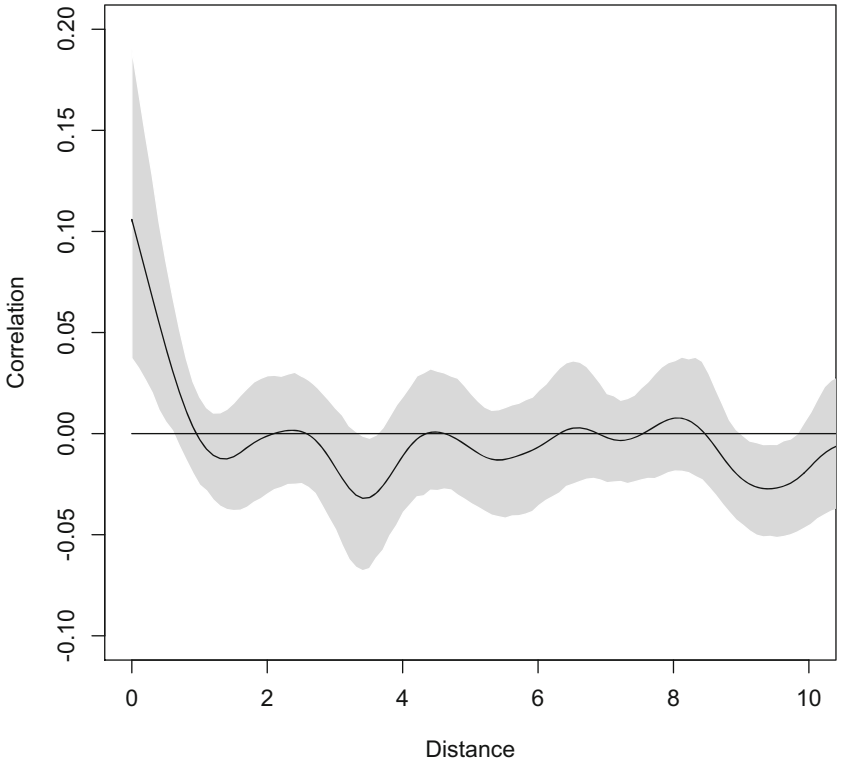

Fig. 13.7 Spatial cross-correlation of prevalence and abundance in the silene data

13.10 Gypsy Moth

The gypsy moth was introduced to the northeastern USA in the late 1860s and has spread at a rate of 10–20 km/year since. The larvae eats leaves of a wide range of trees and shrubs and reach outbreak (defoliating) densities usually around every 10 years. The outbreaks end through epizootics of the *Lymantria dispar* nuclear polyhedrosis virus and more recently the entomopathogenic fungus *Entomophaga maimaiga* that together kills virtually all larvae following outbreaks. Bjørnstad et al. (2010) used the nonparametric spatial covariance function to study the spatiotemporal patterns in these outbreaks. The `gm`-data set contains UTM coordinates and fraction of forests defoliated each year between 1975 and 2002 in 20×20 km grid cells across the northeastern USA. We characterize the patterns of synchrony and time-lagged cross correlation in the outbreak time series.

```
data(gm)
sel=apply(gm[3:30],1,sum)!=0
#Synchrony:
```

```
fit1=Sncf(gm[sel,1]/1000, gm[sel,2]/1000,
     gm[sel,3:30], resamp=500)
#Lag 1 cross-correlation
fit2=Sncf(gm[sel,1]/1000, gm[sel,2]/1000,
     z=gm[sel,3:29], w=gm[sel,4:30], resamp=500)
#Lag 2 cross-correlation
fit3=Sncf(gm[sel,1]/1000, gm[sel,2]/1000,
     z=gm[sel,3:28], w=gm[sel,5:30], resamp=500)
```

The outbreaks are highly synchronized out to 200 km, with a regional average outbreak correlation of around 0.2. The time lagged cross-correlation function shows significant local cross-correlation at the 1-year lag but not 2-year lag, indicating that outbreaks tend to persist spatially for 2 years before collapsing (Fig. 13.8):

```
par(mfrow = c(1, 3))
plot(fit1, ylim = c(-0.1, 1))
plot(fit2, ylim = c(-0.1, 1))
title("Lag 1")
plot(fit3, ylim = c(-0.1, 1))
title("Lag 2")
```

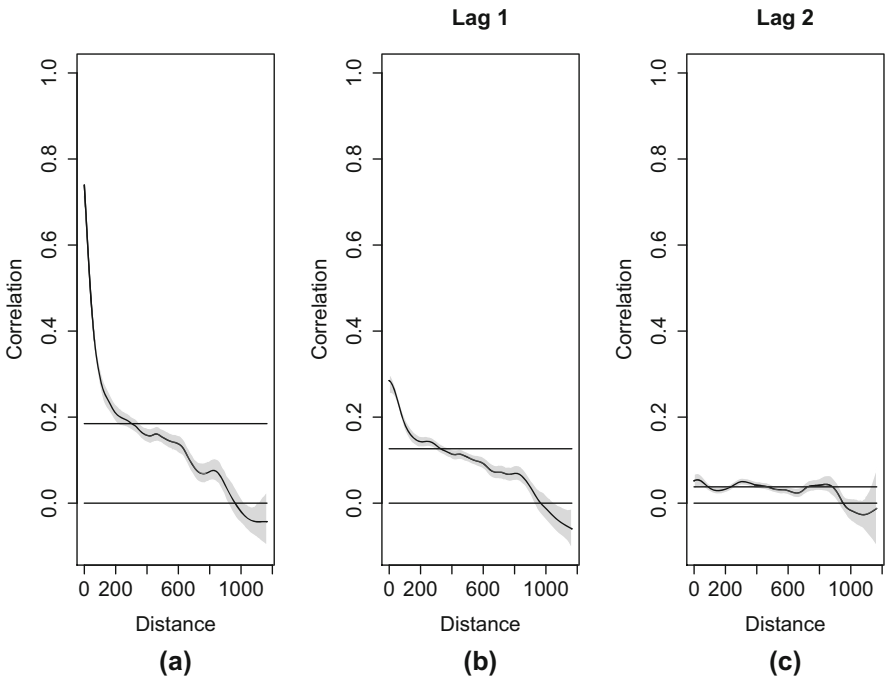

Fig. 13.8 The (**a**) nonparametric spatial covariance function, (**b**) lag-1, and (**c**) lag-2 cross-correlation function of gypsy moth outbreak data from the northeastern USA between 1975 and 2002