

Spatial Statistics

PM569 Lecture 9: Point Pattern 2

Meredith Franklin

Division of Biostatistics, University of Southern California

November 2, 2018

Review of point processes

- ▶ Simple stochastic models for point patterns do not have tractable distributions.
- ▶ To test models against data we use Monte Carlo tests (simulation-based).
- ▶ Monte Carlo steps:
 - Let u_1 be the observed value of a statistic U
 - Let u_i be the values of the statistic U generated by independent random sampling from the distribution of U under a simple hypothesis H_0 (the null hypothesis)
 - Let $u_{(j)}$ denote the j th largest among the u_i , $i = 1, \dots, s$
 - Then, under H_0 , $P\{u_1 = u_{(j)}\} = s^{-1}$, $j = 1, \dots, s$ and rejection of H_0 on the basis that u_1 ranks k th largest or higher gives an exact one sided test of size k/s

Review of point processes

- ▶ Monte Carlo methods are not precisely replicable since they rely on simulated data.
- ▶ An independent set of simulated realizations will result in a different estimated p-value than the first set of realizations.
- ▶ The larger number of simulations the more stable the resulting estimates.
- ▶ We use Monte Carlo methods to test whether our observations are a CSR with homogenous or inhomogeneous Poisson process, cluster process, regular process.

Point Pattern Data

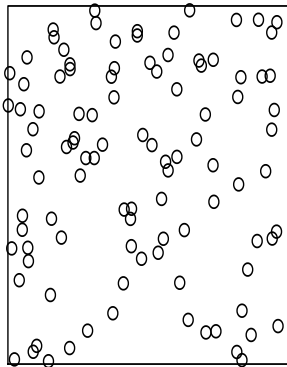
Review of point processes

- ▶ Testing for CSR:
 - Often want to adjust for edge effects.
 - We test for CSR with Ripley's K , which involves a search window with bandwidth h .
 - We test for CSR based on inter-event distances being less than a threshold δ with $H(h)$.
 - We test for CSR based on nearest-neighbour distances with $G(h)$.
- ▶ The types of spatial processes where the Poisson processes is the building block are:
 - Homogeneous Poisson process (constant intensity), used for testing CSR.
 - Inhomogeneous Poisson process (intensity varies across domain), used for testing CSR.
 - Poisson Cluster process (intensity varies for parents and/or children forming clusters), used for testing clustered patterns.
 - Simple inhibition processes, Markovian processes (Strauss and pairwise interaction), used for testing regular patterns.

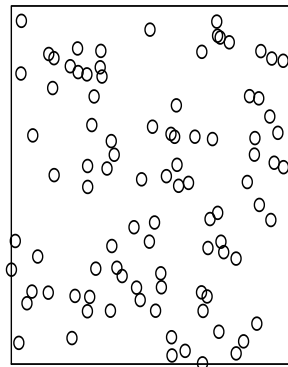
Point Pattern Data

Homogeneous Poisson Process (CSR)

intensity = 100, unit square

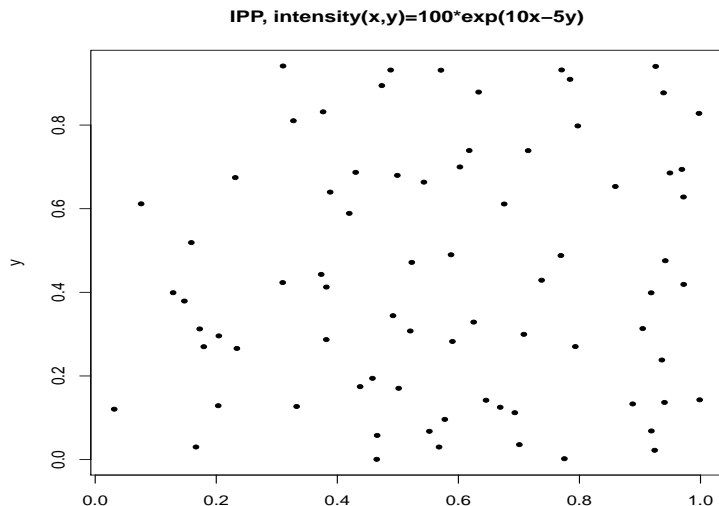


intensity = 1, 10 x 10 square



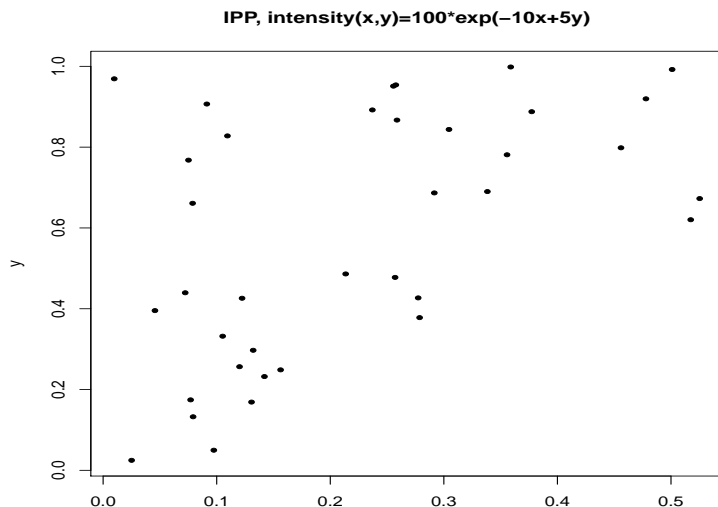
Point Pattern Data

Inhomogeneous Poisson Process



Point Pattern Data

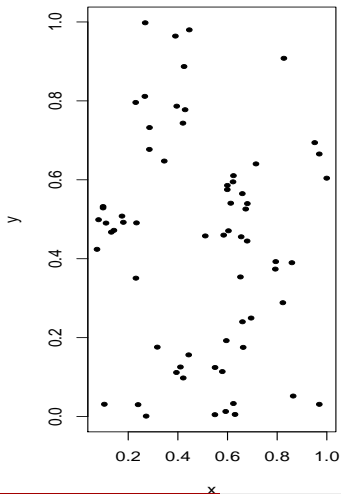
Inhomogeneous Poisson Process



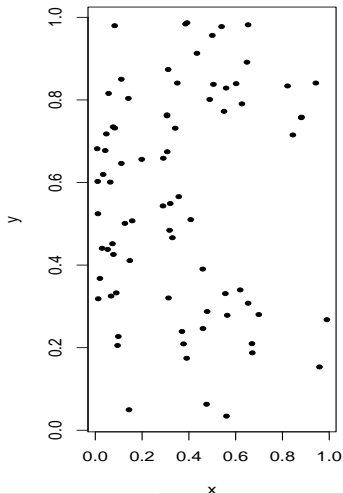
Point Pattern Data

Poisson Clustered Process

PCP, (P,O,Spread)=(25,4,0.0025)



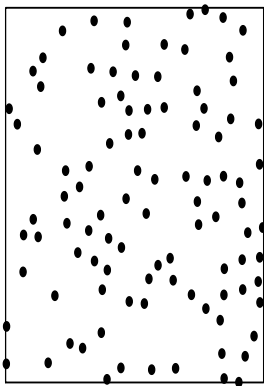
PCP, (P,O,Spread)=(25,4,0.005)



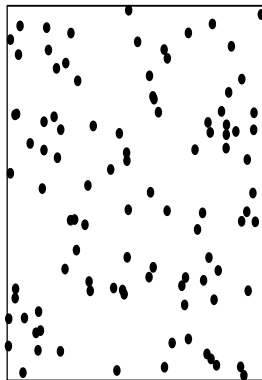
Point Pattern Data

Simple Inhibition Process

SIP, distance 0.05



SIP, distance 0.005



Point Pattern Data

- ▶ Inhomogeneous Point Processes, where the intensity, λ , is not constant.
- ▶ Properties of a spatial point process in terms of the intensity function.
 - First order properties are described by the intensity function.

$$\lambda(x) = \lim_{|dx| \rightarrow 0} \frac{E[N(dx)]}{|dx|}$$

- The first order properties are the mean properties of the random process, describing the expected density of events in any location of the region
- Clusters appear in areas of high intensity
- Under IPP and CPC, clusters occur due to heterogeneities in the intensity function and individual event locations remain independent of one another

Point Pattern Data

- ▶ First order properties are described by the intensity function
- ▶ Example: consider the constant risk hypothesis
 - Each person has the same risk of disease, but we expect more cases in areas with more people at risk
 - Clusters of cases in high population areas will violate CSR but not the constant risk hypothesis
 - We are interested in clustering of disease events after accounting for known variations in population density
 - This requires a generalization of the intensity where we define it as a spatially varying function over the study area
 - As population size increases, so should the expected number of cases

Point Pattern Data

Inhomogeneous Poisson Process intensity function

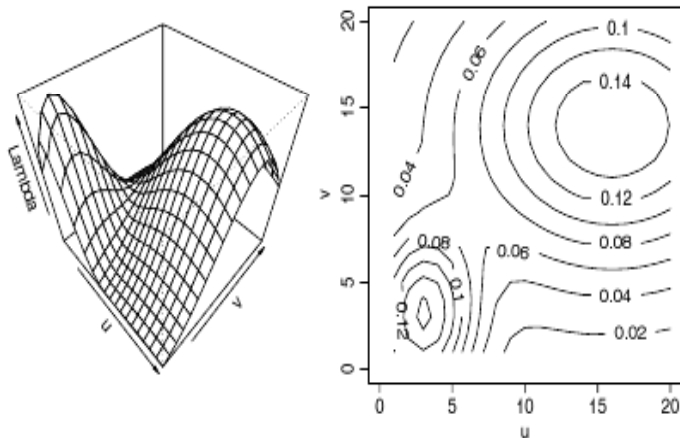


FIG. 5.5 Example intensity function, $\lambda(s)$, for a heterogeneous Poisson point process defined for $s = (u, v)$ and $u, v \in (0, 20)$.

Point Pattern Data

- ▶ The inhomogeneous Poisson process shows lack of events between the modes
- ▶ More events around the mode $(16,14)$ and a narrower peaked area around $(3,3)$
- ▶ Collections of events suggest areas of higher intensity
- ▶ Single realizations make it hard to identify the specific areas of these modes
- ▶ Useful to simulate multiple realizations of the process

Point Pattern Data

- ▶ Second order properties are described by the inter-relationships between events

$$\lambda(x, y) = \lim_{|dx|, |dy| \rightarrow 0} \frac{E[N(dx)N(dy)]}{|dx||dy|}$$

- ▶ This allows us to describe how often events occur within a given distance of other events
- ▶ The second order properties are similar to variance/covariance of the process
- ▶ Allows us to summarize the spatial dependence between events over a wide range of possible spatial scales
- ▶ The Ripley's K function is a second-order statistic

Point Pattern Data

- ▶ Recall the K function for distance h:

$$K(h) = \frac{E[\# \text{ events within } h \text{ of randomly chosen event}]}{\lambda}$$

- ▶ The second order properties gives us insight into the global aspects of the point pattern
- ▶ Are there general patterns of clustering or regularity with respect to CSR or another pattern?

Cox processes

- ▶ Spatial clustering with a spatially varying intensity function of the inhomogeneous Poisson process
- ▶ Varying $\lambda(x)$ and $\lambda(x)$ is a realization of a stochastic process
- ▶ Property 1) it is a non-negative valued stochastic process

$$\{\Lambda(x); x \in \mathbb{R}^2\}$$

- ▶ Property 2) the events for an inhomogeneous Poisson process with intensity function $\lambda(x)$

$$\{\Lambda(x) = \lambda(x); x \in \mathbb{R}^2\}$$

Cox processes

- ▶ The Cox process is homogeneous iff $\Lambda(x)$ is homogeneous:

$$E[\Lambda(x)] = \lambda \forall x$$

$$E[\Lambda(x)\Lambda(x+h)] \text{ depends only on } \|h\|$$

Cox processes

- ▶ The Cox process is linked to the clustered Poisson process
- ▶ Aggregation into clusters may be a result of environmental heterogeneity
- ▶ Clusters of events in regions of high intensity
- ▶ Cox processes are considered doubly stochastic, intensity is heterogeneous but also may be a random quantity
- ▶ $\lambda(x)$ can be drawn from some probability distribution of possible intensity functions over the study area

Cox processes

$$\Lambda(x) = \mu \sum_{i=1}^{\infty} h(x - X_i)$$

- ▶ $\mu > 0$, $h(\cdot)$ is a bivariate pdf, and X_i are points from a Poisson process
- ▶ The Cox process can also be thought of as a specific case of a Poisson cluster process with number of offspring having intensity μ and dispersion around parents with pdf $h(\cdot)$

Cox processes

- ▶ The log-Gaussian Cox process is another form of the Cox process

$$\Lambda(x) = \exp(Z(x))$$

- ▶ $Z(x)$ is a Gaussian process.
- ▶ If $Z(x)$ is stationary with mean μ , variance σ^2 and correlation $\rho(h)$:
 - $\lambda = \exp(\mu + 0.5\sigma^2)$
 - $\gamma(h) = \exp(\sigma\rho(h))$
- ▶ The log-Gaussian Cox process can be fit in R spatstat with the `rLGCP()` function

Simple Inhibition Process

- ▶ This process is used to describe regular patterns
- ▶ Often related to interactions or contagions where the occurrence of an event raises or lowers the probability of subsequent events nearby
- ▶ Useful for modeling the spread of infectious disease (contagion) or an application where an event precludes the occurrence of other events in a nearby area such as animal territories (inhibition)
- ▶ **Contagion** typically refers to the increased likelihood of events occurring near other events
- ▶ **Inhibition** may be absolute, where there is a specified distance around which *no* other events may occur, or it may be probabilistic where there is small but positive probability of an event occurring near other events

Simple Inhibition Process

- ▶ Models for inhibition or contagion processes are Markov point processes or Gibbs processes
- ▶ The general idea is to take a CSR and "delete" points within a distance less than a threshold δ
- ▶ Under a Markov process, the existence of an event in a region depends on the locations of events in a neighbourhood (where neighbourhoods are within regions)
- ▶ There are two ways to do this: 1) to simulate CSR then delete all within a distance δ , and 2) to simulate CSR, record when event was simulated, then delete an event if it is within distance δ of an older event

Simple Inhibition Process

- ▶ We use the packing intensity to describe simple inhibition processes:

$$\tau = \lambda \pi (\delta/2)^2$$

Where λ is the intensity, giving τ to be the proportion of the region A covered by non-overlapping discs of diameter δ

Simple Inhibition Process

- ▶ For simple inhibition process 1) we take a Poisson process with intensity ρ and thin it by the deletion of pairs of events that are less than δ apart
- ▶ In this case, the probability that an event "survives" is $\exp(-\pi\rho\delta^2)$ giving the intensity of a simple inhibition process as:

$$\lambda = \rho \exp(-\pi\rho\delta^2)$$

- ▶ The second order properties can be expressed as:

$$\lambda(h) = \rho^2 \exp(-\rho U_\delta(h)) \qquad h \geq \delta$$

- ▶ $\lambda(h) = 0$ when $0 < h < \delta$, and $U_\delta(h)$ is the area of the union of two discs with equal radius δ and centers distance h apart

Point Pattern Data

Simple Inhibition Process

- ▶ For simple inhibition process 2) we take a a Poisson process with intensity ρ and thin it by the deletion of pairs of "older" events that are less than δ apart
- ▶ The expressions are the same as for process 1) but with the addition of the sequential piece (this process is referred to as the simple sequential inhibition process)
- ▶ Let X_i be a sequence of n events in A , and $d(x, y)$ be the distance between two points x and y . Then:
 - X_1 is simulated from a uniform distribution in A
 - Given (past) $\{X_j = x_j, j = 1, \dots, (i-1)\}$, then X_i (present) is uniformly distributed on the intersection of A with $\{y : d(y, x_j) \geq \delta, j = 1, \dots, (i-1)\}$
- ▶ So the simple sequential inhibition process has packing intensity:

$$\tau = \frac{n\pi(\delta/2^2)}{|A|}$$

Point Pattern Data

- ▶ In R (spatstat), the functions for thinning processes 1) and 2) described above are called `rMaternI` and `rMaternII`
- ▶ The simple sequential inhibition process, called `rSSI` is similar but slightly different:
 - Each new point is generated uniformly in the window and independently of preceding points
 - If a new point lies within distance δ from an existing point then it is rejected and another random point is generated
 - The SSI process ends when no points can be added

Markov point processes

- ▶ The general idea of a Markov point process lies in conditioning, whereby the existence of an event in a finite region A depends on the locations of events in a neighbourhood
- ▶ Inhibition processes are a special form of Markov process: the conditional intensity of an event at a point x given the realization of the process in the remainder of the region A depends on the existence (or otherwise) of an event within distance δ of x
- ▶ General Markov processes were introduced by Ripley and Kelly (1977)
- ▶ Markov point processes are characterized by the likelihood ratio with respect to a Poisson process of unit intensity

Point Pattern Data

Markov point processes

- ▶ Let's call the likelihood ratio $f(\cdot)$
- ▶ If $\mathbf{X} = \{x_1, \dots, x_n\}$ denotes a finite set of points in A then $f(\mathbf{X})$ indicates how much more likely is the configuration of events \mathbf{X} than a homogeneous point process (with unit intensity)
- ▶ We can factorize the likelihood ratio to:

$$f(\mathbf{X}) = \alpha \prod_{i=1}^n g_i(x_i) \prod_{j>i} g_{ij}(x_i, x_j) \dots g_{12\dots n}(x_1, x_2, \dots, x_n)$$

- ▶ Where α is a normalizing constant
- ▶ We also define two points x and y in A to be neighbours if $d(x, y) < \delta$ for some $\delta > 0$ where $d(x, y)$ is the distance between x and y
- ▶ We also define a clique (recall areal data) as a set of mutual neighbours, and the neighbourhood of x to be the set of points $\{y \in A : 0 < d(x, y), \delta\}$

Markov point processes

- ▶ The point process with these definitions is Markov with range δ if the conditional intensity at the point x given the configuration of the other events in A depends only on the configuration in the neighbourhood of x
- ▶ The g -functions from the above equation are unity *unless* the x form a clique

Examples of Markov point processes: the Strauss process

$$f(\mathbf{X}) = \alpha \beta^n \gamma^p$$

- ▶ Where α is the normalizing constant, β is the intensity of the process, γ is the interaction between neighbours, and p is the number of distinct pairs of neighbours in \mathbf{X}
- ▶ If $\gamma = 1$ then the Strauss process gives a Poisson process with intensity β
- ▶ if $\gamma = 0$ then the Strauss process gives a simple inhibition process because no two events may be neighbours
- ▶ In R spatstat, the Strauss process is simulated with `rStrauss`

Point Pattern Data

Examples of Markov point processes: the pairwise interaction process

$$f(\mathbf{X}) = \alpha \beta^n \prod_{i \neq j} h\{d(x_i, x_j)\}$$

- ▶ Where α is the normalizing constant, β is the intensity of the process, $h(d)$ is non-negative for all distances and the product is over all pairs of distinct points in \mathbf{X}
- ▶ The additional restriction is that $h(d)$ is bounded and that $h(d) = 0$ for all distances less than some $\delta > 0$
- ▶ This restriction limits the number of events in A by imposing a minimum allowable distance δ between any two events
- ▶ The pairwise interaction process may be fit in R `spatstat` using the `rmh` function

Examples of Markov point processes: the pairwise interaction process

- ▶ The pairwise interaction process may be simulated using the following steps (MCMC):
 1. For the initial realization, consider n points $\{x_1, \dots, x_n\}$
 2. Delete one of the points in $\{x_1, \dots, x_n\}$
 3. Generate a point y from a uniform distribution in A , and accept y with probability $p(y)$
 4. Repeat 2-3 until the MCMC converges

Review of point processes

- ▶ Testing for CSR:
 - Adjusting for edge effect
 - Testing for CSR with Ripley's K .
 - Testing for CSR based on inter-event distances, $H(h)$.
 - Testing for CSR based on nearest-neighbour distances, $G(h)$.
- ▶ Spatial processes, Poisson processes are the building block:
 - Homogeneous Poisson process (constant intensity).
 - Inhomogeneous Poisson process (intensity varies across domain).
 - Poisson Cluster process (intensity varies for parents and/or children forming clusters), Cox process when the cluster intensity is spatially varying.
 - Simple inhibition processes, Markovian processes (Strauss and pairwise interaction) for regular patterns.
- ▶ In this lecture we will focus on fitting point process models and on methods for detecting clusters.

Fitting point process models

- ▶ Given our set of observed point events $\{x_1, \dots, x_n\}$ in region A we wish to fit a model (which is stationary and isotropic)
- ▶ Model fitting is approached by estimating the parameters of the particular process
 - Example: fitting the parameters ρ , μ and σ^2 of a clustered process
 - Example: fitting a parametric form of the intensity of an inhomogeneous poisson process
- ▶ We use familiar fitting methods: Least squares, Maximum Likelihood and non-parametric methods.

Fitting point process models: Least Squares

- ▶ We start with $K(h)$ and the estimator $\hat{K}(h)$ (or L, or G, or H) for parameter fitting
- ▶ This is useful when the mathematical form of $K(h)$ is known either explicitly or as an integral (which is true for some point processes)
- ▶ If $K(h)$ is not known we use the simulated realizations
- ▶ Example, to fit a homogeneous poisson cluster process we have parameters $\theta = (\rho, \sigma)$ and the Ripley's K function is:

$$K(h, \theta) = \pi h^2 + \frac{1}{\rho}(1 - \exp(-h^2/(4\sigma^2)))$$

- ▶ And we estimate $\hat{K}(h)$ from the data

Fitting point process models: Least Squares

- ▶ Given the theoretical K-function and the estimator $\hat{K}(h)$ we minimize the deviance:

$$D(\theta) = \int_0^{h_0} [(\hat{K}(h))^c - (K(h, \theta))^c]^2 dh$$

- ▶ Where h_0 is the maximum distance which is typically chosen as 1/3 to 1/2 of the width of a rectangular region, and c is the power transformation
- ▶ The power transformation controls the sampling fluctuations in $\hat{K}(h)$ which can increase with h and have influence on $\hat{\theta}$ (i.e. it is a variance stabilizer)
- ▶ Examples of c are $c=0.5$ for a pattern that is not too different from CSR, $c=0.25$ for cluster patters. However, choose a variety of c values in practice in order to see how sensitive the results are

Fitting point process models: Least Squares Estimation Steps

1. Compute the edge corrected $\hat{K}(h)$

$$\hat{K}(h) = \frac{|A|}{n^2} \sum_{i=1}^n \sum_{j \neq i} I(h_{i,j} \leq h)$$

2. Choose a theoretical model for $K(h, \theta)$ where θ are the parameters of the model
3. Find $\hat{\theta}$ that minimizes the deviance for a given c

$$D(\theta) = \int_0^{h_0} [(\hat{K}(h))^c - (K(h, \theta))^c]^2 dh$$

Fitting point process models: Least Squares Estimation

- ▶ When $K(h, \theta)$ is unknown because there is no closed form, use the simulated method (for s simulations):

$$\bar{K}_s(h, \theta) = \frac{1}{n} \sum_{i=1}^s \hat{K}_i(h, \theta)$$

- ▶ Finding $\bar{K}_s(h, \theta)$ for each value of θ can be prohibitive computationally
 1. Start with a small number of simulations, s
 2. Find a first approximation of $\hat{\theta}$
 3. Repeat with a larger value for s

Fitting point process models: Least Squares Estimation Steps

- ▶ A weighted version of the deviance, shown to have asymptotic properties (consistency and asymptotic normality), is often used:

$$D(\theta) = \int_0^{h_0} w(h)[(\hat{K}(h))^c - (K(h, \theta))^c]^2 dh$$

- ▶ The weight $w(h)$ is a weight on the distance also controls the variance
- ▶ When $c = 0.5$ and $w(h) = 1$ we have the Poisson cluster process
- ▶ See Guan and Sherman, J R Stat Soc (2007) for asymptotic properties

Fitting point process models: Least Squares Estimation

- ▶ In R `spatstat`, cluster or Cox point process models are fit with least squares estimation through the `kppm` function with the `method="mincontrast"` option
- ▶ To fit a log-Gaussian Cox point process, use the function `lgcp.estK`
- ▶ To fit the Matern cluster process (type I or II), use the function `matclust.estK`

Fitting point process models: Maximum Likelihood

- ▶ To fit inhomogeneous Poisson, Strauss and pairwise interaction processes we need to rely on likelihood methods
- ▶ Recall for the inhomogeneous Poisson process:
 - $N(A)$ is Poisson with mean $\int_A \lambda(x)dx$
 - Conditional on $N(A) = n$, the n events in A form an independent random sample from A with a probability distribution function proportional to $\lambda(x)$
- ▶ We can define the process based on its conditional intensity
- ▶ Namely, the conditional probability of finding a point of the process inside an infinitesimal neighbourhood du of the location u given the complete point pattern \mathbf{x} is $\lambda(u, \mathbf{x})du$

Fitting point process models: Conditional Intensity

- ▶ For example, CSR has conditional intensity $\lambda(u, \mathbf{x}) = \lambda$
- ▶ The IPP has conditional intensity $\lambda(u, \mathbf{x}) = \lambda(u)$
- ▶ Sometimes the IPP trend is denoted as $\beta(u)$ and indicates "spatial trend"
- ▶ The Strauss process has conditional intensity $\lambda(u, \mathbf{x}) = \beta^n \gamma^p$ where β is the intensity, γ is the interaction parameter, and p is the number of points of \mathbf{x} that lie within a distance δ of u (i.e. pairs of neighbours)
- ▶ For example, the Strauss process with $\gamma < 1$ dependence between points is reflected in the fact that the conditional probability of finding a point of the process at the location u is reduced if other points of the process are present within a distance δ . And when $\gamma = 0$, the conditional probability of finding a point at u is zero if there are any other points of the process within a distance δ of this location.

Fitting point process models: Pseudolikelihood

- ▶ Because maximum likelihood is difficult for point process models, the log of the pseudolikelihood is maximized, using the conditional intensity
- ▶ For a point process governed by parameter θ the pseudolikelihood is:

$$PL(\theta; x) = \prod_{i=1}^n \lambda_{\theta}(x_i; x) \exp\left(-\int_A \lambda_{\theta}(u; x) du\right)$$

- ▶ The maximum pseudolikelihood estimate of θ minimizes the above equation

Fitting point process models: Pseudolikelihood

- ▶ We need to take the log of the pseudolikelihood equation and approximate the integral using a "quadrature" scheme (see Berman and Turner, 1992)

$$\int_A \lambda_\theta(u; x) du \approx \sum_{j=1}^m \lambda_\theta(u_j, x) w_j$$

- ▶ Where u_j are "quadrature points" in A and $w_j \geq 0$ are the "quadrature weights"

Fitting point process models: Pseudolikelihood

- ▶ The quadrature points can be chosen as all data points, x_i and the addition of some dummy points u_j , i.e. $\{x_1, \dots, x_n\} \subset \{u_1, \dots, u_m\}$
- ▶ Then the log pseudolikelihood can be written:

$$\log PL(\theta; x) = \sum_{j=1}^m z_j \log \lambda_{\theta}(u_j; x) - w_j \lambda_{\theta}(u_j; x)$$

- ▶ Where $z_j = 1$ if u_j is a data point, and $z_j = 0$ if u_j is a dummy point

Fitting point process models in R

- ▶ In R spatstat the function `ppm` fits models by pseudolikelihood based on the conditional intensity $\lambda_{\theta}(u, x)$
- ▶ The model must be loglinear in the parameters θ
- ▶ For example, the Strauss process can be written:

$$\log \lambda(u, x) = \log \beta + \log \gamma p$$

- ▶ So $\theta = (\log \beta, \log \gamma)$ are the "regular parameters" and the parameter driving the interaction, p is the "irregular parameter"

Fitting point process models in R

- ▶ Thus in `spatstat` `ppm` the conditional intensity is split into first and higher order terms:

$$\log \lambda_{\theta}(u, x) = \eta S(u) + \phi V(u, x)$$

- ▶ The first order term $S(u)$ describes the spatial inhomogeneity of the intensity (including covariate effects) and the higher order term $V(u, x)$ describes the interactions between points

Fitting point process models in R

- ▶ The general form of ppm is `ppm(X, trend, interaction,...)`
- ▶ The trend argument specifies any spatial trend or covariate effects and is written as an R formula
- ▶ The default trend formula is ~ 1 , which indicates $\lambda(u) = 1$, corresponding to a process without spatial trend or covariate effects. The formula $\sim x$ indicates the vector statistic $\lambda(x, y) = (1, x)$ corresponding to a spatial trend of the form $\exp(\alpha + \beta x)$, where α, β are coefficient parameters to be estimated, while $\sim x + y$ indicates $\lambda(x, y) = (1, x, y)$ corresponding to $\exp(\alpha + \beta x + \gamma y)$

Fitting point process models in R

- ▶ The general form of ppm is `ppm(X, trend, interaction,...)`
- ▶ The interaction term represents the interaction function $V(u, x)$
- ▶ For example, the Strauss function with interaction distance $\delta = 0.1$ is fit with `ppm(X, 1, Strauss(r=0.1))`
- ▶ Note that the ppm with a specified higher order term calls the first order term for the intensity β rather than λ
- ▶ `spatstat` automatically creates a quadrature scheme but it can be controlled by the user through the function `quadscheme`

Point Pattern Data: Detecting Clusters

- ▶ So far we have simulated point processes, found statistics that indicate a global measure of what pattern there may be, and fit models to specific types of point patterns.
- ▶ If we want to find where clusters of observations are located, we need a different set of tools called scan statistics.
- ▶ Goals of scan statistics:
 - To determine areas where the number of events is inconsistent with the number observed over the rest of the study area.
 - Compare local rates of events (or case/control ratios) to detect clusters.

Point Pattern Data: Detecting Clusters

- ▶ First attempts at cluster detection methods were developed in the 1980s: geographical analysis machine and the cluster evaluation permutation procedure.
- ▶ Early cluster detection methods:
 - Graphical in nature
 - Divides the region up into a fine grid
 - Uses a search window, which is a circle of predefined radius, larger than the grid spacing in order for circles to overlap
 - Centers the circle over each grid cell, then moves across the region
 - The number of cases occurring within the search window are counted
 - The circle is drawn on the map if the count observed within the circle exceeds some tolerance level
 - The tolerance level may be defined as the observed count exceeding all of the counts associated with that circle under random selection ($N=499$)

Point Pattern Data: Detecting Clusters

- ▶ The circle in these cases is considered a circular uniform kernel
- ▶ In kernel estimation, the kernel is centered on the data locations, in scan statistics, the circle is centered on the grid

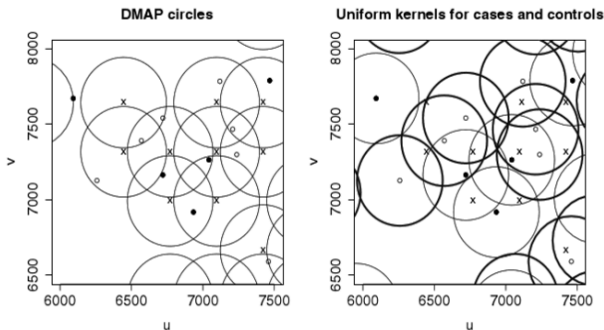


FIG. 6.8 Equivalence between the case/control ratio within circles surrounding grid points (left-hand plot) and the ratio of intensity (density) functions based on circular uniform kernels (right-hand plot). Case locations appear as filled small circles, control locations as open small circles, and grid points as “x” symbols. In the left-hand plot, large circles represent radii of 300 units around each grid point. In the right-hand plot, dark circles represent control kernel radii, and lighter circles represent case kernel radii (both 300 units).

Point Pattern Data: Spatial Scan Statistics

- ▶ A scan statistics involves the definition of a moving window and a statistical comparison measurement (count or rate) within the window to the same measurement outside the window.
- ▶ Kulldorff (1995, 1997) defines a spatial scan statistic that is similar to the geographical analysis machine, but with an inferential framework.
- ▶ Scan statistics tend to consider circular windows with variable radii ranging from the smallest distance between a pair of cases to the user-defined upper bound.
- ▶ The circles can be centered on either grid locations (like earlier methods) or the set of locations. Different results will be seen depending on what is chosen.

Point Pattern Data: Spatial Scan Statistics

Setup:

Let $N_{1,in}$ represent the number of case locations and $N_{in} = N_{0,in} + N_{1,in}$ be the total number of people at risk (or number of case and control locations) inside a particular window.

Let $N_{1,out}$ represent the number of case locations and $N_{out} = N_{0,out} + N_{1,out}$ be the total number of people at risk (or number of case and control locations) outside the window.

The test statistic is:

$$T_{scan} = \max \left(\frac{N_{1,in}}{N_{in}} \right)^{N_{1,in}} \left(\frac{N_{1,out}}{N_{out}} \right)^{N_{1,out}} I \left(\frac{N_{1,in}}{N_{in}} > \frac{N_{1,out}}{N_{out}} \right)$$

$I(\cdot)$ is the indicator function (i.e. we only maximize over windows where the observed rate inside the window exceeds that outside the window)

Point Pattern Data: Spatial Scan Statistics

- ▶ The maximum observed likelihood ratio statistic provides a test of overall general clustering and an indication of the most likely clusters with significance determined by Monte Carlo testing of the constant risk hypothesis.
- ▶ SaTScan is a software package that enables this.
- ▶ There is also a new R package called scanstatistics that looks promising <https://cran.r-project.org/web/packages/scanstatistics/vignettes/introduction.html>