BST 6200 Spatial Statistics and Disease Mapping

Steven E. Rigdon

Spring 2020

Chapter 6: Point Pattern Analysis in R

A **point process** is a set of point locations $(x_1, y_1),\ (x_2, y_2),\ \ldots,\ (x_n, y_n)$ in some domain.

A **marked point process** is a set of point locations $(x_1, y_1),\ (x_2, y_2),\ \ldots,\ (x_n, y_n)$ in some domain together with some "marked" variable z_i.

6.3 Kernel Density Estimation

Objective: estimate the probability density of points.

$$\hat{f}(x,y) = \frac{1}{n\ h_x h_y} \sum_{i=1}^{n} k\left(\frac{x - x_i}{h_x}, \frac{y - y_i}{h_y}\right)$$

k is called the **kernel**. Often k is the PDF of the bivariate normal distribution with mean $\boldsymbol{\mu}[0, 0]$ and covariance matrix

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Other choices: UNIF$(-1, 1)$, $t(1)$.

Multivariate Normal Distribution

$\boldsymbol{X} = [X_1, X_2, \ldots, X_p]^t$. $X \sim \mathrm{N}(\boldsymbol{\mu}, \Sigma)$ means that $\boldsymbol{X}$ has a multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ with PDF

$$f(\boldsymbol{x}) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^t\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

Choice of h_x and h_y

Usually $h_x = h_y$.

Larger h_x and h_y lead to smoother density estimates.

Smaller h_x and h_y lead to "choppier" density estimates.

Scott's (1997) rule:

$$h_x = \sigma_x \left(\frac{2}{3n}\right)^{1/6}$$

$$h_y = \sigma_y \left(\frac{2}{3n}\right)^{1/6}$$

Trial and error can be used to get appropriate h_x and h_y.

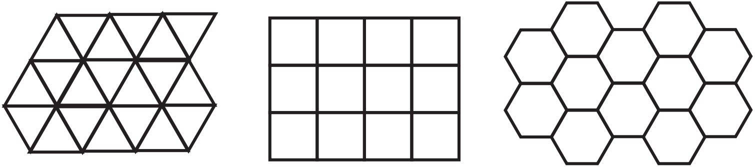
Using R for KDE

BC_Chapter6.R

Tesselations (or Tiling the Plane)

A tesselation or a tiling of a plane is a set of geometric objects that cover the plane with no gaps and no overlap.
A regular tesselation or tiling of a plane is a set of identical (congruent) regular polygons that cover the plane with no gaps and no overlap.
There are only three regular tesselatioins of the plane:

1. triangles
2. squares
3. hexagons

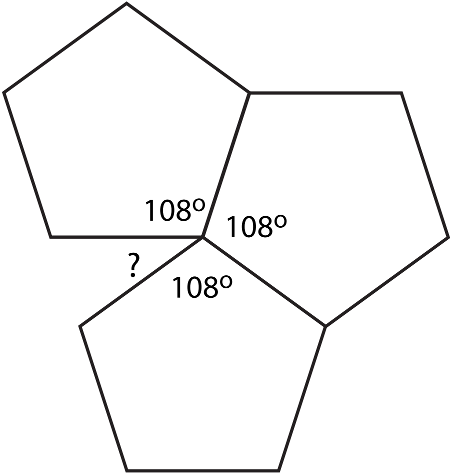


Why are there only three?

			If it is a Regular Polygon (all sides are equal, all angles are equal)	
Shape	**Sides**	**Sum of Interior Angles**	**Shape**	**Each Angle**
Triangle	3	180°		60°
Quadrilateral	4	360°		90°
Pentagon	5	540°		108°
Hexagon	6	720°		120°
Heptagon *(or Septagon)*	7	900°		128.57...°
Octagon	8	1080°		135°
...	...	..	...	...
Any Polygon	n	(n-2) × 180°	n	(n-2) × 180° / n

Source: https://www.dominatethegmat.com/2013/11/gmat-geometry-polygons/

Why Pentagonal Tiling Won't Work



Hexagonal Binning

BC_Chapter6.R

6.5 Second-Order Analysis of Point Patterns

N_d = number of events within a distance of d
from a randomly selected event point

Ripley's K function

$$K(d) = \frac{E(N_d)}{\lambda}, \qquad d > 0$$

Complete Spatial Randomness (CSR)

The point process exhibits **complete spatial randomness** if the joint (bivariate) distribution of event locations is uniform across the domain.

Book's definition (equivalent to above only in the case of a rectangular domain). The point process exhibits **complete spatial randomness** if the x and y components are independent and the marginal distributions are uniform across the domain.

See BC_Chapter6.R

CSR is also called a **Poisson process** because the number of events that fall in a region A has a Poisson distribution with mean equal to $\lambda|A|$, where $|A|$ denotes the area of A.

K Function under CSR

If we have complete spatial randomness, then

$$K_{\text{CSR}}(d) = \frac{E(N_d)}{\lambda} = \frac{\lambda \times \text{Area(circle of rad. } d)}{\lambda} = \pi d^2$$

If $K(d) > K_{\text{CSR}}(d)$ for some d this "suggests that there is an excess of nearby points – or to put it another way, there is clustering at the spatial scale associated with the distance d.

If $K(d) < K_{\text{CSR}}(d)$ for some d this "suggests spatial dispersion at this scale – the presence of one point suggests other points are less likely to appear nearby ..."

Estimating K

Let d_{ij} denote the distance between points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

Let $\hat{\lambda} = \dfrac{n}{|A|}$.

$$\hat{K}(d) = \frac{1}{\hat{\lambda}} \sum_{i=1}^{n} \sum_{j=1; j \neq i}^{n} \frac{I(d_{ij} < d)}{n(n-1)}$$

Here $I(\texttt{logical}) = 1$ if `logical` is true and 0 if `logical` is false. This is called an indicator function.

Edge Effects

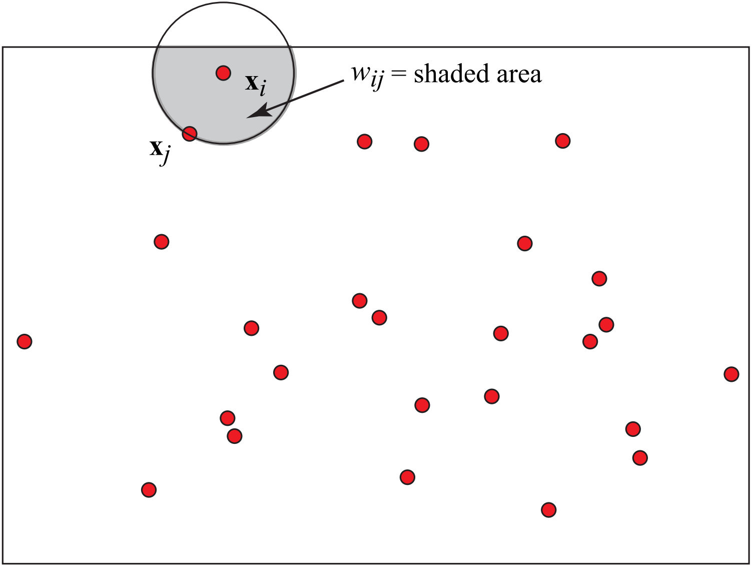
If points are close to the edge of the region A part of the circle of radius d may be outside of A, so fewer points should be expected within d.

Modified estimator:

$$\hat{K}(d) = \frac{1}{\hat{\lambda}} \sum_{i=1}^{n} \sum_{j=1; j \neq i}^{n} \frac{2I(d_{ij} < d)}{n(n-1) w_{ij}}$$

where w_{ij} is the area of that part of the circle centered at $\boldsymbol{x}_i$ passing through $\boldsymbol{x}_j$ that lies within the region A.

Explanation of w_{ij}

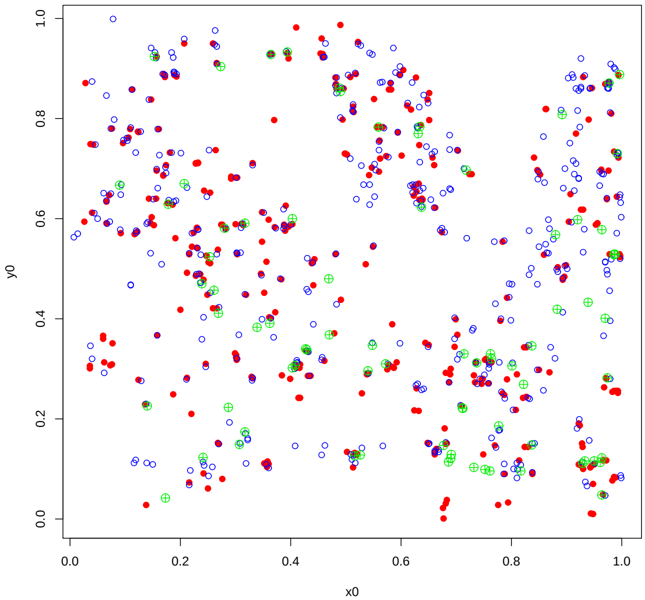


Is there evidence that the process is anything but CSR (Poisson process)

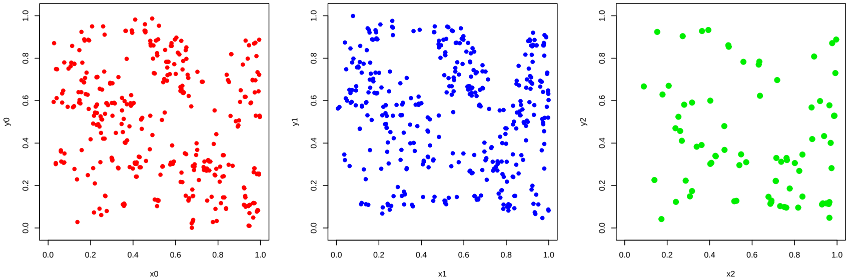
To test this, follow these steps

1. Simulate CSR on the same region A with the same number of points.
2. Compute $\hat{K}(d)$ for the simulated data set.
3. Repeat steps 1 and 2 many (say 100) times.
4. Plot the lower and upper (2.5% and 97.5%) percentiles; this is called the envelope.
5. Plot $\hat{K}(d)$ for the actual data on the same graph and see where it is not contained in the envelope.

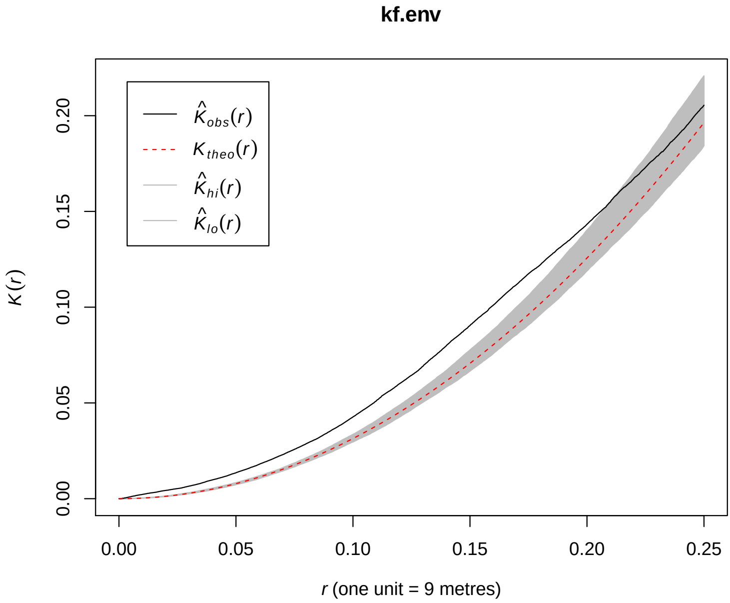
Bramblecane Data



Bramblecane Data



K Function Estimate with Envelope



Multiple Tests

If we compare $\hat{K}(d)$ with the envelope obtained by simulation, we are doing multiple tests. Thus the overall α (the probability of rejecting at least one test given CSR) is unknown.

Method 1 to compare $\hat{K}(d)$ with that expected under CSR (πd^2)

$$MAD = \max_d |\hat{K}(d) - K_{\mathrm{CSR}}(d)| = \max_d |\hat{K}(d) - \pi d^2|$$

Method 2 finds the average squared deviation between $\hat{K}(d)$ and $K_{\mathrm{CSR}}(d)$. This is called the dclf test (Diggle, Cressie, Loosmore, Ford).

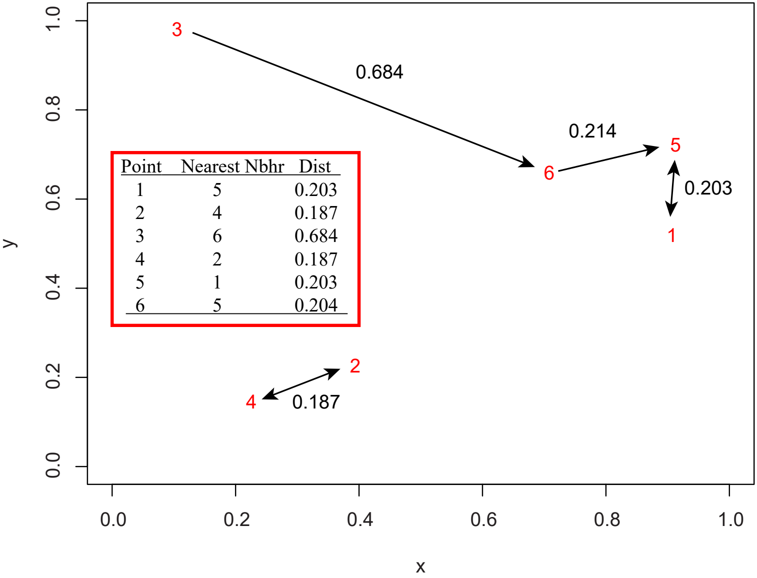
The L Function

$$L(d) = \sqrt{\frac{K(d)}{\pi}}$$

This is just a transformation of the $K(d)$ so it seems to offer no new information, but under CSR

$$L_{\mathsf{CSR}}(d) = \sqrt{\frac{K_{\mathsf{CSR}}(d)}{\pi}} = \sqrt{\frac{\pi d^2}{\pi}} = d$$

The G Function



Point	Nearest Nbhr	Dist
1	5	0.203
2	4	0.187
3	6	0.684
4	2	0.187
5	1	0.203
6	5	0.204

The G Function

$$\hat{G}(d) = \frac{\#\text{ of pairs where min distance is } \leq d}{\text{number of points}}$$

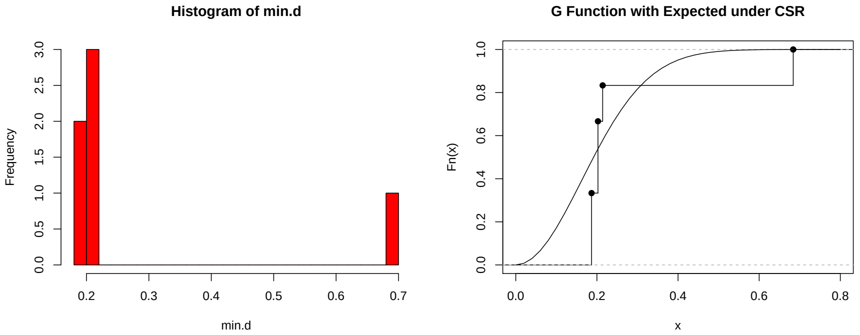
$$= \text{empirical CDF of shortest distances}$$

The probability of at least one event within a distance d is

$$P(\text{nearest point is within } d) = 1 - P(X = 0)$$

$$= 1 - \frac{(\lambda A)^0 \exp(-(\lambda A))}{0!} = 1 - \exp(-\lambda \pi d^2)$$

Notice error in equation (6.11) in textbook.



Histogram of min.d
Frequency
min.d
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0.2
0.3
0.4
0.5
0.6
0.7
G Function with Expected under CSR
Fn(x)
x
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8

The Cross K Function

For a marked point process with objects of type i and j, consider how many points of type i are within d units of points of type j.

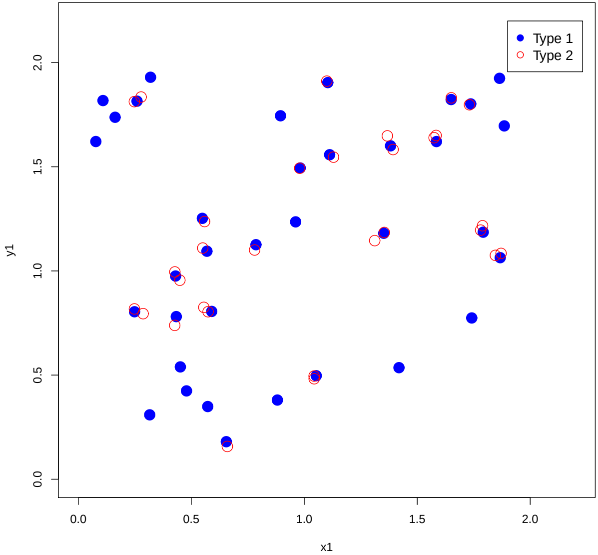
$$\hat{K}_{ij}(d) = \frac{1}{\hat{\lambda}_j} \sum_{k=1}^{n_i} \sum_{\ell=1}^{n_j} \frac{I(d_{k\ell} < d)}{n_i n_j}$$

Notes:

1. This is not symmetric in i and j.
2. There is an error in equation (6.12) in the book. The first expression to the right of the equal sign should be λ_j.

Example of Cross K Function

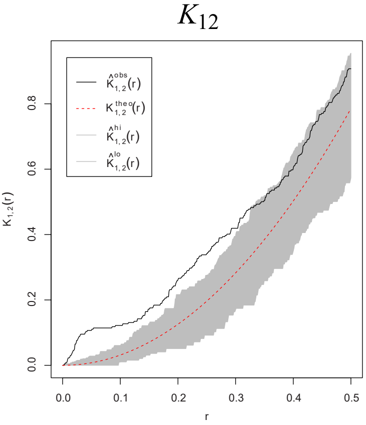
Most type 2s are near a type 1, but not conversely.

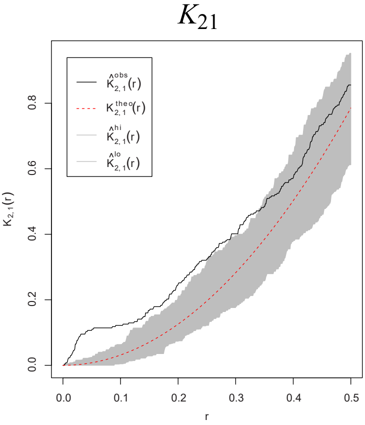


Most type 2s are near a type 1, but not conversely.

K_{12}: Do 2s tend to be near a type 1? Yes.

K_{21}: Do 1s tend to be near a type 2? Not necessarily.





K_{12}

```
Diggle-Cressie-Loosmore-Ford test of CSR
Monte Carlo test based on 500 simulations
Summary function: Kcross["1", "2"](r)
Reference function: theoretical
Alternative: two.sided
Interval of distance values: [0, 0.5]
Test statistic: Integral of squared
absolute deviation
Deviation = observed minus theoretical

data: crossEx.ppp
u = 0.011543, rank = 21, p-value = 0.04192
```

K_{21}

```
Diggle-Cressie-Loosmore-Ford test of CSR
Monte Carlo test based on 500 simulations
Summary function: Kcross["2", "1"](r)
Reference function: theoretical
Alternative: two.sided
Interval of distance values: [0, 0.5]
Test statistic: Integral of squared
absolute deviation
Deviation = observed minus theoretical

data: crossEx.ppp
u = 0.0047363, rank = 82, p-value = 0.1637
```