# *GIST 4302/5302: Spatial Analysis and Modeling*

*Point Pattern Analysis*

Guofeng Cao

www.spatial.ttu.edu

Department of Geosciences
Texas Tech University

guofeng.cao@ttu.edu

Spring 2017

## Characteristics:

- set of $n$ point locations with recorded "events", e.g., locations of trees, disease or crime incidents $S = \{\mathbf{s}_1, \ldots, \mathbf{s}_i, \ldots, \mathbf{s}_n\}$
- point locations correspond to all possible events or to subsets of them
- attribute values also possible at same locations, e.g., tree diameter, magnitude of earthquakes (*marked point pattern*) $W = \{w_1, \ldots, w_i, \ldots, w_n\}$

## Analysis objectives:

- detect spatial clustering or repulsion, as opposed to complete randomness, of event locations (in space and time)
- if clustering detected, investigate possible relations with nearby "sources"

*Further issues:*

- analysis of point patterns over large areas should take into account distance distortions due to map projections
- boundaries of study area should not be arbitrary
- analysis of sampled point patterns can be misleading
- one-to-one correspondence between objects in study area and events in pattern

# Simple Descriptive Statistics

## *Mean center of a point pattern:*

- point with coordinates $\bar{s} = (\bar{x}, \bar{y})$:

$$\bar{x} = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i} \quad \text{and} \quad \bar{y} = \frac{\sum_{i=1}^{n} w_i y_i}{\sum_{i=1}^{n} w_i}$$

- center of point pattern, or point with average $x$ and $y$-coordinates

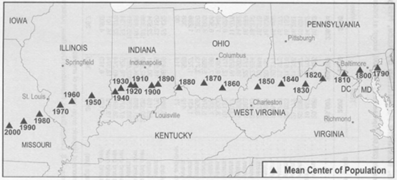
## *Median center of a point pattern:*

- *both* of the following two centers are called *median centers*, although they are essentially different (confusing!)
  - the intersection between the median of the $x$ and the $y$ coordinates
  - *center for minimum distance:* $\mathbf{s}_c \in \{\mathbf{s}_1, \ldots, \mathbf{s}_n\} s.t. min \sum_{i=1}^{n} |\mathbf{s}_i - \mathbf{s}_c|$
- the first type of *median center* is not unique, and there is no closed form for the second type
- $p$-median problem (a typical problem in spatial optimization ): the problem of locating $p$ "facilities" relative to a set of "customers" such that the sum of the shortest demand weighted distance between "customers" and "facilities" is minimized

## Changes of population center (year 1790-2000):

| Year | Median center | | Mean center | | |
|---|---|---|---|---|---|
| | Latitude-N | Longitude | Latitude-N | Longitude-W | Approximate location |
| 1790 (August 2) | (NA) | (NA) | 39 16 30 | 76 11 12 | In Kent County, MD, 23 miles E of Baltimore MD |
| 1850 (June 1). . | (NA) | (NA) | 38 59 00 | 81 19 00 | In Wirt County, WV, 23 miles SE of Parkersburg, WV[1] |
| 1900 (June 1). . | 40 03 32 | 84 49 01 | 39 09 36 | 85 48 54 | In Bartholomew County, IN, 6 miles SE of Columbus, IN |
| 1950 (April 1) . . | 40 00 12 | 84 56 51 | 38 50 21 | 88 09 33 | In Richland County, IL, 8 miles NNW of Olney, IL |
| 1960 (April 1) . . | 39 56 25 | 85 16 60 | 38 35 58 | 89 12 35 | In Clinton County, IL, 6.5 miles NW of Centralia, IL |
| 1970 (April 1) . | 39 47 43 | 85 31 43 | 38 27 47 | 89 42 22 | In St. Clair County, IL, 5.3 miles ESE of Mascoutah, IL |
| 1980 (April 1) . . | 39 18 60 | 86 08 15 | 38 08 13 | 90 34 26 | In Jefferson County, MO, .25 mile W of DeSoto, MO |
| 1990 (April 1) . . | 38 57 55 | 86 31 53 | 37 52 20 | 91 12 55 | In Crawford County, MO, 10 miles SE of Steelville, MO |
| 2000 (April 1) . . | 38 45 23 | 86 55 51 | 37 41 49 | 91 48 34 | In Phelps County, MO, 3 miles E of Edgar Springs, MO |

NA Not available. [1]West Virginia was set off from Virginia, Dec. 31, 1862, and admitted as a state, June 19, 1863.

## Standard distance of a point pattern:

- average squared deviations of $x$ and $y$ coordinates from their respective mean:

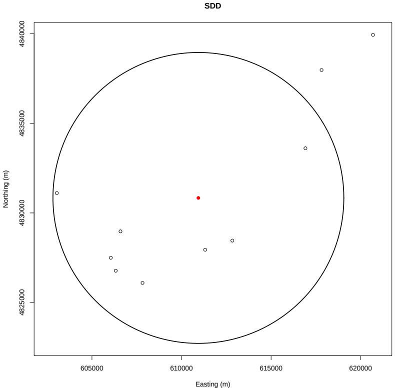
$$d_{std} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2 + \sum_{i=1}^{n}(y_i - \bar{y})^2}{n-2}}$$

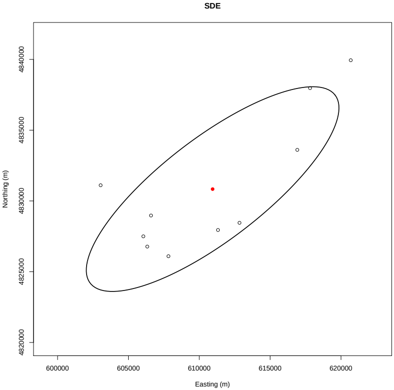
- related to standard deviation of coordinates, a summary circle (centered at $\bar{s}$ with radius $d_{std}$) of a point pattern

## Standard deviational ellipse:

- Taking directional effects into account for *anisotropy* cases
- Please refer to Levine and Associates, 2004 for calculations

## Examples:




## Remarks:

- indicates overall shape and center of point pattern
- *do not suffice to fully specify a spatial point pattern*

*1st order (i.e., intensity): absolute location of events on map:*

- Quadrat methods
- Density Estimation (KDE)
- Moran's I and Geary's C

*2nd order (i.e., interactions): interaction of events:*

- Nearest neighbor distance
- Distance functions G, K, F, L
- Getis-Ord Gi* and Anselin local Moran's I

# Quadrat methods

Consider a point pattern with $n$ events within a study region $A$ of area $|A|$

*Global intensity:*

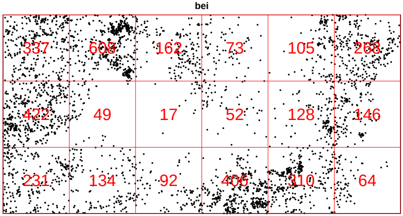
$$\hat{\lambda} = \frac{n}{|A|} = \frac{\#\text{of events within} A}{|A|}$$

*Local intensity via quadrats*

1. partition $A$ into $L$ sub-regions $A_l, l = 1, \ldots, L$ of equal area $|A_l|$ (also called quadrats)
2. count number of events $n(A_l)$ in each sub-region $A_l$
3. convert sample counts into estimated intensity rates as:

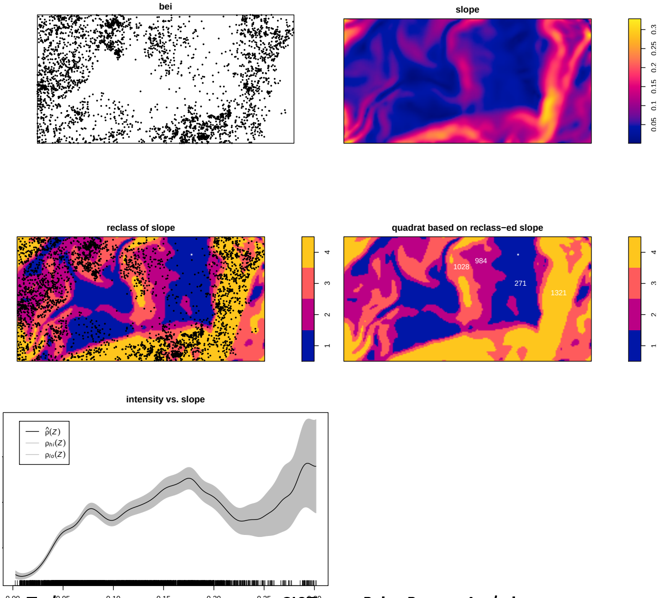
$$\hat{\lambda}(A_l) = \frac{n(A_l)}{|A_l|}$$

# Quadrat methods




- estimated rates $\hat{\lambda}(A_l)$ over set of quadrats
- reveal large-scale patterns in intensity variation over $A$
- larger quadrats yield smoother intensity maps; smaller quadrats yield 'spiky' intensity maps
- size, origin, and shape of quadrats is critical (recall: *MAUP*)
- *only first-order effects are captured*

# Dependence of intensity on a covariate (Inhomogeneous cases)

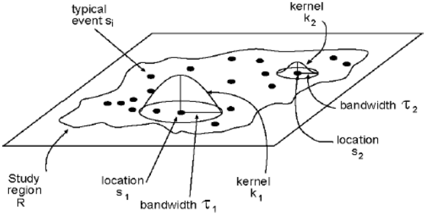


## Procedure of Kernel Density Estimation (KDE)

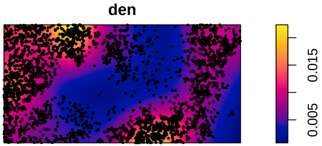
1. define a kernel $K(\boldsymbol{s}; r)$ of radius (or bandwidth) $r$ centered at any arbitrary location $\boldsymbol{s}$
2. estimate local intensity at $\boldsymbol{s}$ as:

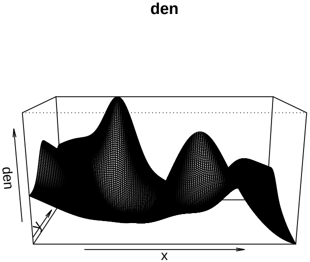
$$\hat{\lambda}(\boldsymbol{s}) = \frac{1}{n}\sum_{i=1}^{n} K(\boldsymbol{s}_i - \boldsymbol{s}; r)$$

3. repeat estimation for all points s in the study region to create a density map



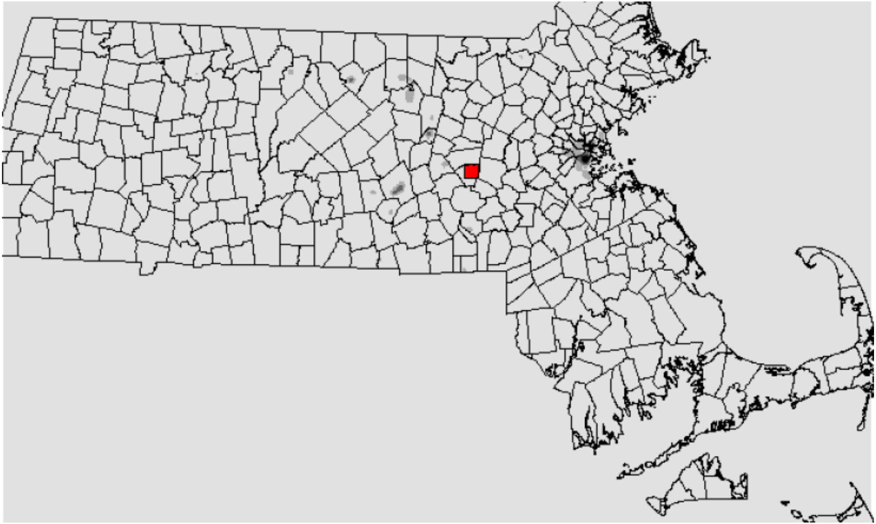
## Example for the previous dataset:



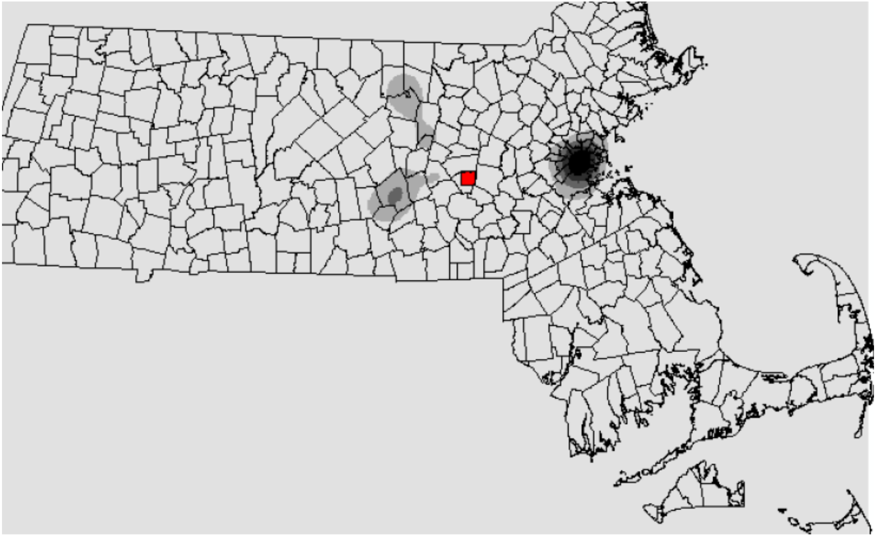
# Kernel Density Estimation

## Example with 2km bandwidth

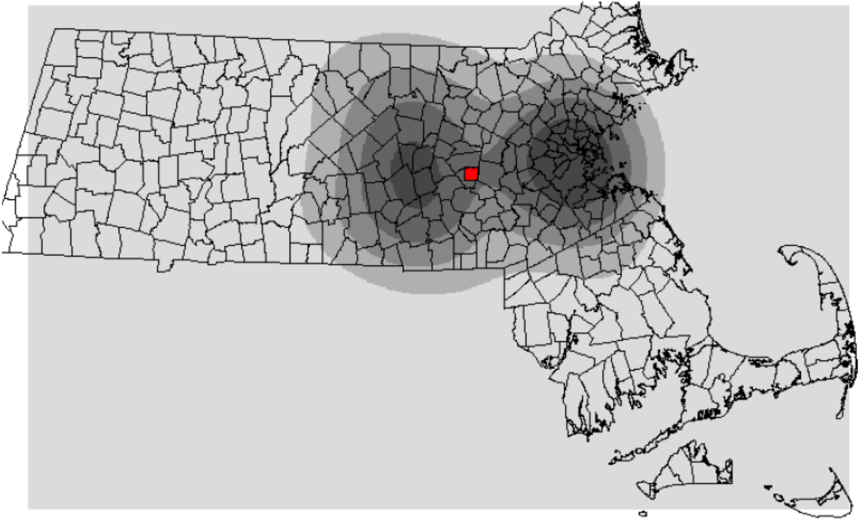


# Kernel Density Estimation

## Example with 10km bandwidth



## Example with 40km bandwidth



## Comments

- Choice of kernel function is not critical (Diggle, 1985)
- Choice of bandwidth, or degree of smoothing critical:
  - Small bandwidth $\rightarrow$ spiky results
  - Large bandwidth $\rightarrow$ loss of detail
- Multi-scale analyses can use these bandwidth characteristics to investigate both broad trends and localized variation
- How to choose bandwidth: choose the degree of smoothing subjectively, by eye, or by formula (Diggle)
- could define local bandwidth based on function of presence of events in neighborhood of $\boldsymbol{s}$ (i.e., adaptive kernel estimation)

## What does the output of KDE means?

# Distance-based Descriptors of Point Patterns

- Distances: accessing second order effects
  - Event-to-event distance: distance $d_{ij}$ between event at arbitrary location $s_i$ and another event at another arbitrary location $s_j$:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

  - Point-to-event distance: distance $\tilde{d}_{pj}$ between a randomly chosen point at location $\tilde{s}_p$ and an event at location $s_j$:

$$\tilde{d}_{pj} = \sqrt{(\tilde{x}_p - x_j)^2 + (\tilde{y}_i - y_j)^2}$$

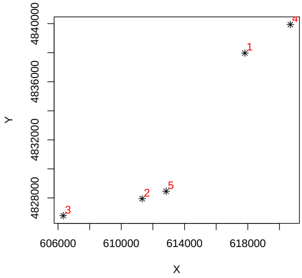
  - Event-to-nearest-neighbour distance: distance $d_{min}(s_i)$ between an event at location $s_i$ and its *nearest neighbor* event:

$$d_{min}(s_i) = min\{d_{ij}, j \neq i, j = 1, \ldots, n\}$$

  - Point-to-nearest-neighbour distance (i.e., *empty space distance*): distance $d_{min}(\tilde{s}_p)$ between a randomly chosen point at location $\tilde{s}_p$ and its *nearest neighbor* event:

$$\tilde{d}_{min}(\tilde{s}_p) = min\{\tilde{d}_{pj}, j = 1, \ldots, n\}$$

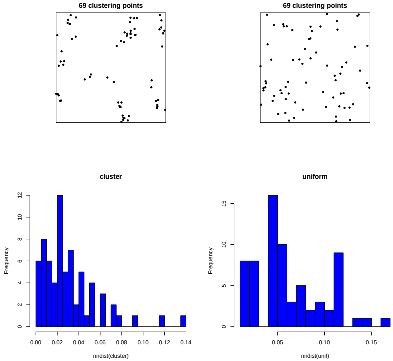
# Event-to-Nearest-Neighbor Distances




| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0.00 | 11947.70 | 16042.65 | 3481.22 | 10742.98 |
| 2 | 11947.70 | 0.00 | 5126.79 | 15219.58 | 1599.07 |
| 3 | 16042.65 | 5126.79 | 0.00 | 19481.59 | 6720.59 |
| 4 | 3481.22 | 15219.58 | 19481.59 | 0.00 | 13913.70 |
| 5 | 10742.98 | 1599.07 | 6720.59 | 13913.70 | 0.00 |

*Table :* Euclidean distance matrix

## Nearest neighbour distances




- Mean nearest neighbour distance: Average of all $d_{min}(s_i)$ values

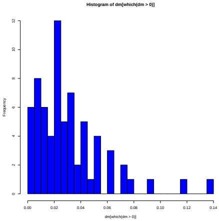
$$\bar{d}_{min} = \frac{1}{n} \sum_{i=1}^{n} d_{min}(s_i)$$

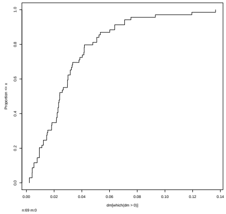
## The G function

- Definition: nearest neighbour distance function, i.e., proportion of event-to-nearest-neighbor distances $d_{min}(s_i)$ no greater than given distance cutoff $d$, estimated as:

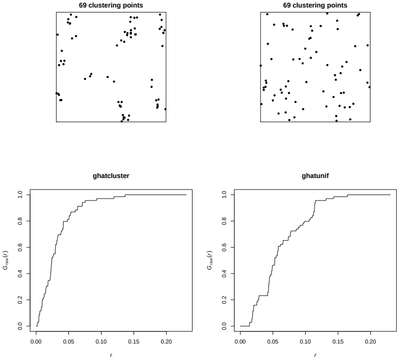
$$\hat{G}(d) = \frac{\#\{d_{min}(s_i) < d, i = 1, \ldots, n\}}{n}$$

- alternative definition: cumulative distribution function (CDF) of all $n$ event-to-nearest-neighbor distances; instead of computing average $\bar{d}_{min}$ of $d_{min}$ values, compute their CDF
- the G function provides information on event *proximity*
- example for previous clustering point pattern:



# Examples of G function




## Expected plot:

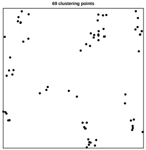
- for clustered events, $\hat{G}(d)$ rises sharply at short distances, and then levels off at larger $d$-values
- for randomly-spaced events, $\hat{G}(d)$ rises gradually up to the distance at which most events are spaced, and then increases sharply

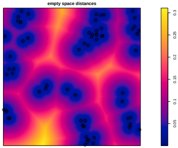
## Definition

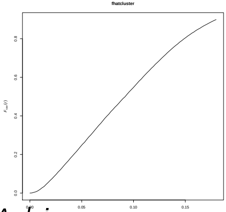
- proportion of point-to-nearest-neighbor distances (i.e., *empty space distances*) $\tilde{d}_{min}(s_p)$ no greater than given distance cutoff $d$, estimated as:

$$\hat{F}(d) = \frac{\#\{\tilde{d}_{min}(\tilde{s}_p) < d, p = 1, \ldots, m\}}{m}$$

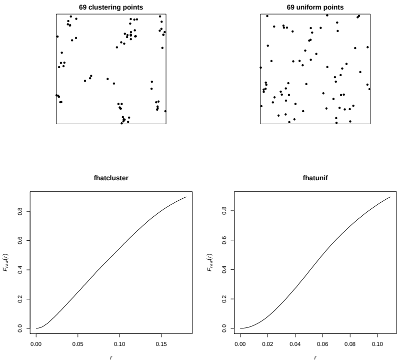
- alternative definition: cumulative distribution function (CDF) of all $m$ point-to-nearest-neighbor distances
- the F function provides information on event proximity to voids
- Examples for previous clustering point pattern:



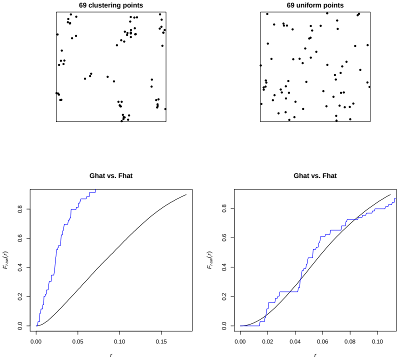



# Examples of F function




## Expected plot:

- for clustered events, $\hat{F}(d)$ rises sharply at short distances, and then levels off at larger $d$-values
- for randomly-spaced events, $\hat{F}(d)$ rises rapidly up to the distance at which most events are spaced, and then levels off (there are more nearest neighbors at small distances from randomly placed points)




## *Expected plot:*

- for clustered events, $\hat{G}(d)$ rises faster
- for randomly-spaced events, $\hat{F}(d)$ tends to be close to $\hat{G}(d)$

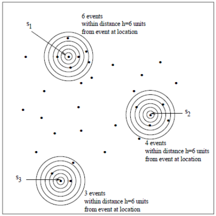
# The $K$ function

*Working with pair-wise distances&looking beyond nearest neighours*

## Concept

1. construct set of concentric circles (of increasing radius $d$) around each event
2. count number of events in each distance "band"
3. cumulative number of events up to radius $d$ around all events becomes the sample $K$ function $\hat{K}(d)$

**Example of K function estimation**



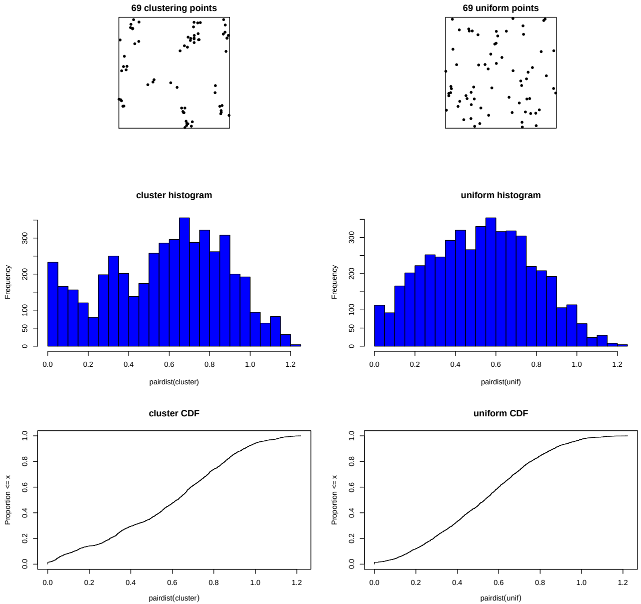
*Working with pair-wise distances&looking beyond nearest neighours*

- Formal definition:

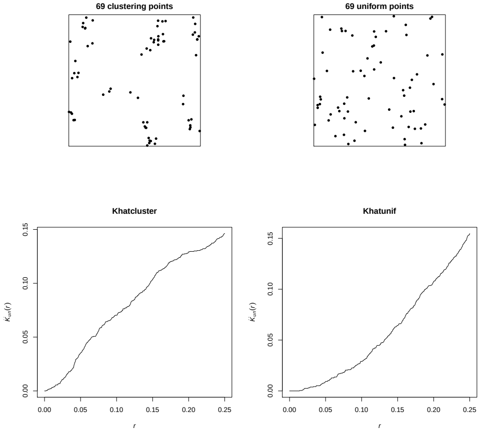
$$
\begin{aligned}
K(d) &= \frac{1}{\lambda}\frac{\#\{d_{ij} \leq d, i,j = 1,\ldots,n\}}{n} \\
&= \frac{|A|}{n}\frac{\#\{d_{ij} \leq d, i,j = 1,\ldots,n\}}{n} \\
&= |A|(\text{proportion of event-to-event distance} \leq d)
\end{aligned}
$$

- In other words, the $\hat{K}(d)$ is the sample cumulative distribution function (CDF) of all $n^2$ event-to-event distances, scaled by $|A|$

# Examples of Event-to-Event Distance Histogram and CDFs



# Examples of K functions




- the sample $K$ function $\hat{K}(d)$ is monotonically increasing and is a scaled (by area $|A|$) version of the CDF of E2E distances

## Spatial point patterns

- set of $n$ point locations with recorded "events"

## Describing the first-order effect

- overal intensity
- local intensity (quadrat count and kernel density estimation)

## Describing the second-order effect

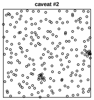
- nearest neighbour distances
  - the G function
- empty space distances
  - the F function
- pair-wise distances
  - the K function

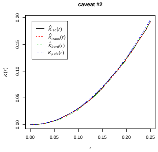
# Caveats

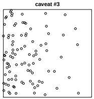
## Caveats:

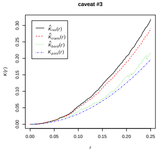
- theoretical G, F, K functions are defined and estimated under the *assumption that the point process is stationary (homogeneous)*
- these summary functions *do not completely characterise the process*
- if the process is not stationary, deviations between the empirical and theoretical functions (e.g. $\hat{K}$ and K) are not necessarily evidence of interpoint interaction, since they may also be attributable to variations in intensity

## Example



## *Descriptive analysis:*

- set of quantitative (and graphical) tools for characterizing spatial point patterns
- different tools are appropriate for investigating first- or second-order effects (e.g., kernel density estimation versus sample G function)
- can shed light onto whether points are clustered or evenly distributed in space

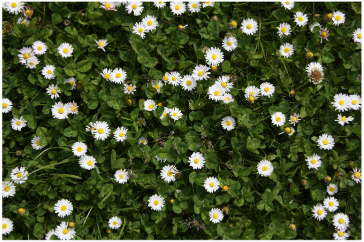
## *Limitation:*

- no assessment of *how* clustered or *how* evenly-spaced is an observed point pattern
- no yardstick against which to compare observed values (or graph) of results

## Statistical analysis:

- assessment of whether an observed point pattern can be regarded as one (out of many) realizations from a particular spatial process
- measures of confidence with which the above assessment can be made (how likely is that the observed pattern is a realization of a particular spatial process)

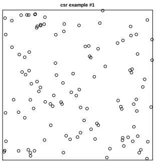
## Are daisies randomly distributed in your garden?

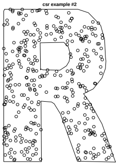


## Complete Spatial Randomness (CSR)

- yardstick, reference model that observed point patterns could be compared with, i.e., null hypothesis
- = *homogeneous (uniform) Poisson point process*
- basic properties:
    - the number of points falling in any region $A$ has a Poisson distribution with mean $\lambda|A|$
    - given that there are $n$ points inside region $A$, the locations of these points are i.i.d. and uniformly distributed inside $A$
    - the contents of two disjoint regions $A$ and $B$ are independent

### Example:



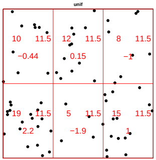
## Quadrat counting test

- partition study area $A$ into $L$ sub-regions (quadrats), $A_1, \ldots, A_L$
- count number of events $n(A_l)$ in each sub-region $A_l$
- Under the null hypothesis of CSR, the $n(A_l)$ are i.i.d. Poisson random variables with the same expected value
- The Pearson $\chi^2$ goodness-of-fit test can be used
  - test statistics: Pearson residual $\sum_l \epsilon(A_l)^2$

    $$\epsilon(A_l) = \frac{n(A_l) - \mu(A_l)}{\sqrt{\mu(A_l)}},$$

    where $\mu(A_l)$ indicates the expected number of events in $A_l$
  - $\sum_{l=1} \epsilon(A_l)^2$ is assumed to follow $\chi^2$ distribution

*Example*




- three values indicate the number of observations, CSR-expected number of observations, and the Pearson residuals
- $p$-value = 0.617

## Nearest neighbour index

- Compares the mean of the distance observed between each point and its nearest neighbor ($\bar{d}_{min}$) and the expected mean distance under CSR $E(d_{min})$

$$NNI = \frac{\bar{d}_{min}}{E(d_{min})}$$

- Under CSR, we have:

$$E(d_{min}) = \frac{1}{2\sqrt{\lambda}}$$

$$\sigma(d_{min}) = \frac{0.26136}{\sqrt{n^2/A}}$$

## *Nearest neighbour index test*

- Test statistics:

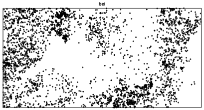
$$z = \frac{\bar{d}_{min} - E(d_{min})}{\sigma(d_{min})},$$

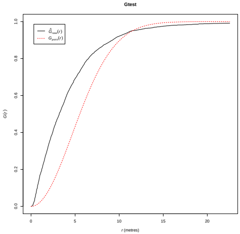
- $z$ is assumed to follow Gaussian distribution, thus, if $z < -1,96$ or $z > 1.96$, we are 95% confident that the distribution is not randomly distributed

- The G function is a function of nearest-neighbour distances
- For a homogeneous Poisson point process of intensity $\lambda$, the nearest-neighhour distance distribution (the G function) is known to be:

$$G(d) = 1 - \exp\{-\lambda \pi d^2\}$$

## Example



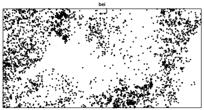
# The F Function under CSR

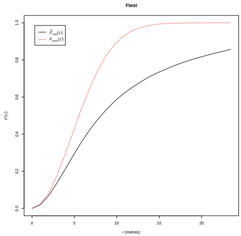
- The F function is a function of empty space distances
- For a homogeneous Poisson point process of intensity $\lambda$, the empty space distance distribution (the F function)is known to be:

$$F(d) = 1 - \exp\{-\lambda \pi d^2\}$$

- Equivalent to the G function
- Intuitively, because points (events) of the Poisson process are independent of each other, the knowledge that a random point is a event of a point pattern does not affect any other event of the process

## Example



# The K Function under CSR

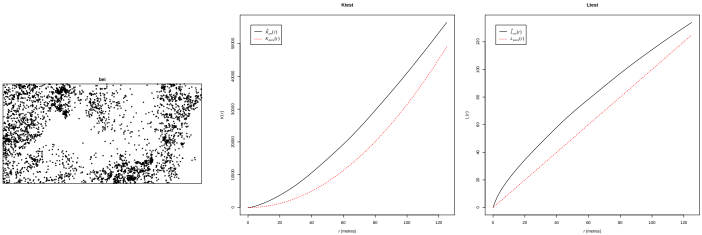
- The K function is a function of pair-wise distances
- For a homogeneous Poisson point process of intensity $\lambda$, the pair-wise distance distribution (the K function) is known to be:

$$K(d) = \pi d^2$$

- A commonly-used transformation of K is the L-function:

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

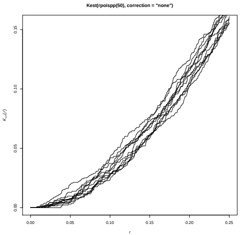
## Example



# Monte Carlo test

- because of random variability, we will never obtain perfect agreement between sample functions (say the K function) with theoretical functions (the theoretical K functions), even with a completely random pattern

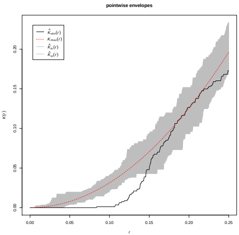
## Example



# *Monte Carlo test*

- A *Monte Carlo* test is a test based on simulations from the null hypothesis
- Basic procedures:
  - generate $M$ independent simulations of CSR inside the study region $A$
  - compute the estimated K functions for each of these realisations, say $\hat{K}^{(j)}(r)$ for $j = 1, \ldots, M$
  - obtain the pointwise upper and lower envelopes of these simulated curves
  - <u>not</u> a confidence interval

## *Example*



## Statitsical analysis of spatial point patterns:

- allows to quantify departure of results obtained via exploratory tools, e.g., $\hat{G}(d)$, from expected such results derived under specific null hypotheses, here CSR hypothesis
- can be used to assess to what extent observed point patterns can be regarded as realizations from a particular spatial process (here CSR)
- Same concepts can be applied for hypothesis of other types of point processes (e.g., Poisson cluster process, Cox process)

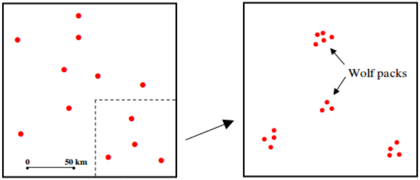
## Sampling distribution of a test statistics

- lies at the heart of any statistical hypothesis testing procedure, and is tied to a particular null hypothesis
- simulation and analytical derivations are two alternative ways of computing such sampling distributions (the latter being increasingly replaced by the former)

## Edge Effects

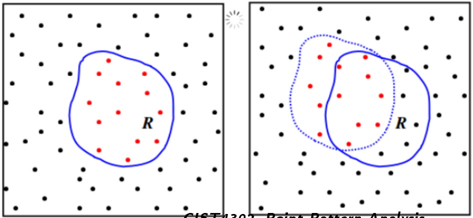
## Scale effects

- Wolf pack example




- Nearest neighour distance (NN distance, G,F functions) vs K function

## Edge effects



# Recap

## Extended into line processes

- Line density