4 - Análise de padrões pontuais espaciais.

Prof. Vinícius D. Mayrink
EST171 - Estatística Espacial

Sala: 4073 Email: vdm@est.ufmg.br

1^o semestre de 2024

Introdução.

A análise de padrões pontuais aparece em muitas áreas de pesquisa.

Exemplo em ecologia: determinar a distribuição espacial (e suas causas) de uma espécie de árvore em uma floresta. Se duas espécies de árvore são consideras, podemos estar interessados em verificar se elas estão igualmente distribuídas na floresta ou se há competição entre elas.

Exemplo em epidemiologia: determinar se os casos de uma doença estão aglomerados. Isto pode ser verificado através da comparação da distribuição espacial dos casos de doença em relação à localização de um conjunto de controles.

Descreveremos aqui aspectos básicos para a análise de padrões pontuais.

Em geral, um processo pontual é um processo estocástico no qual observamos os locais de algum evento de interesse dentro de uma região limitada A.

Diggle (2003) definiu um processo pontual como um mecanismo estocástico que gera um conjunto contável de eventos.

Os locais dos eventos gerados por um processo pontual na área de estudo A formarão o que chamamos de padrão pontual.

Covariáveis podem ser registradas em cada local do evento de interesse.

Pacotes para a análise de padrões pontuais espaciais.

Existem diversos pacotes do R implementando diferentes funções para a análise de padrões pontuais. Nosso foco será: `splancs` e `spatstat`.

Para ilustrar o uso de algumas das técnicas para analisar padrões pontuais, considere os dados a seguir.

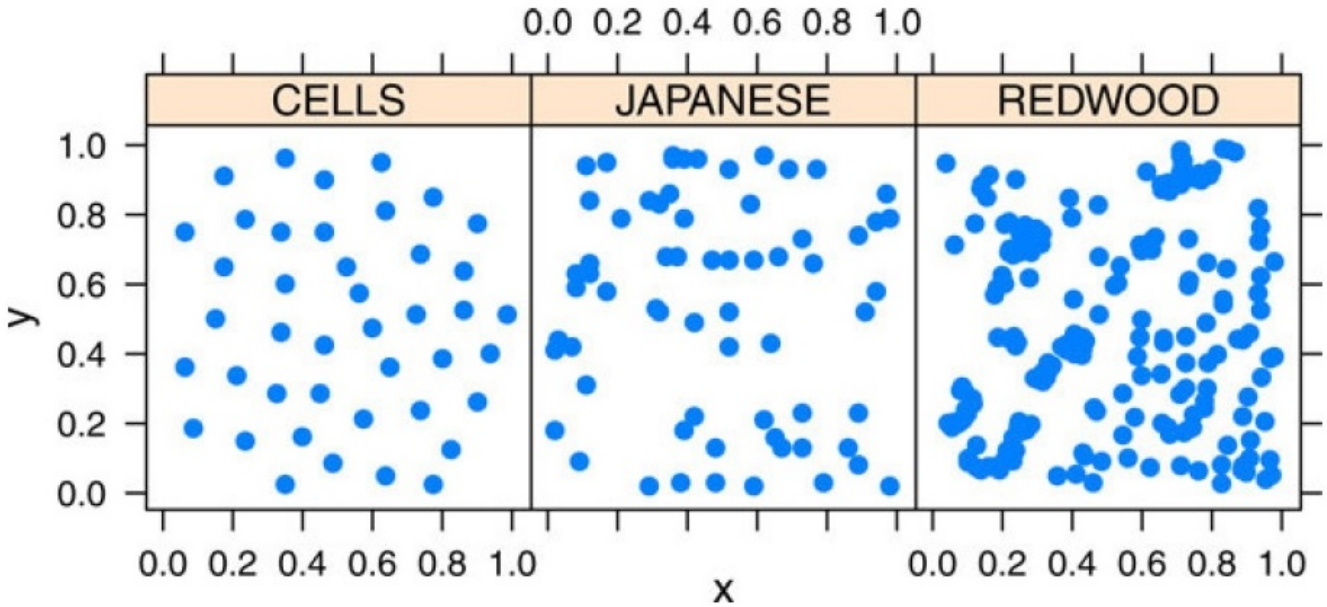

Fig.1: Padrões pontuais de dados reescalados para o quadrado unitário. Esquerda = locais de centros de células, Meio = locais de pinheiros japoneses, Direita = locais de árvores Redwood. Dados disponíveis no pacote `statspat`.

Primeiramente iremos explorar o exemplo dos pinheiros japoneses.

```
> library(spatstat)
> data(japanesepines)
> summary(japanesepines)

Planar point pattern: 65 points
  Average intensity 65 points per square unit (one unit = 5.7 metres)

Coordinates are given to 2 decimal places
  i.e. rounded to the nearest multiple of 0.01 units (one unit = 5.7 metres)

  Window: rectangle = [0, 1] x [0, 1] units
  Window area = 1 square unit
  Unit of length: 5.7 metres.
```

A localização dos pontos foi adaptada ao quadrado unitário. As localizações originais podem ser obtidas considerando a informação "*unit length: 5.7 metres*".

O pacote `spatstat` usa objetos ppp para gravar padrões pontuais.

O pacote `maptools` usa objetos `SpatialPoints` para esse tipo de gravação. Felizmente, o `maptools` possui algumas funções para converter de ppp para `SpatialPoints`.

Quando alteramos objetos ppp com janela retangular para o formato `SpatialPoints`, as coordenadas serão reescaladas para os valores originais.

```
> library(maptools)
> spjpines <- as(japanesepines, "SpatialPoints")
> summary(spjpines)

  Object of class SpatialPoints
  Coordinates:
       min  max
  [1,]   0  5.7
  [2,]   0  5.7

  Is projected: NA
  proj4string : [NA]
  Number of points: 65
```

Podemos retornar ao quadrado unitário usando os comandos:

```
> spjpines1 <- elide(spjpines, scale = TRUE, unitsq = TRUE)
> summary(spjpines1)

Object of class SpatialPoints
Coordinates:
      min max
[1,]    0   1
[2,]    0   1
Is projected: NA
proj4string : [NA]
Number of points: 65
```

Retornando para um objeto ppp:

```
> pppjap <- as(spjpines1, "ppp")
> summary(pppjap)

Planar point pattern: 65 points
Average intensity 65 points per square unit

Coordinates are given to 2 decimal places
i.e. rounded to the nearest multiple of 0.01 units

Window: rectangle = [0, 1] x [0, 1] units
Window area = 1 square unit
```

Estes padrões pontuais foram obtidos por amostragem em diferentes regiões, mas não é difícil encontrar exemplos nos quais diferentes tipos de eventos ocorrem na mesma região.

Em epidemiologia, é comum existir dois tipos de pontos: casos de uma certa doença e controles que refletem a distribuição espacial da população.

Este tipo de padrão pontual é chamado de padrão pontual marcado (*marked*) visto que cada ponto é atribuído a um grupo e indexado de acordo com isso.

Exemplo (dados de asma): resultados de um estudo caso-controle realizado em 1992 sobre a incidência de asma em crianças de *North Derbyshire* (Reino Unido). Será explorada a relação entre asma e a proximidade de rodovias e três fontes de poluição (queima de carvão, fábrica química e centro de tratamento de lixo). No estudo, um número de covariáveis relevantes também foram coletadas por meio de um questionário respondido pelos pais das crianças. Crianças com asma são consideradas casos enquanto que as demais são definidas como controles.

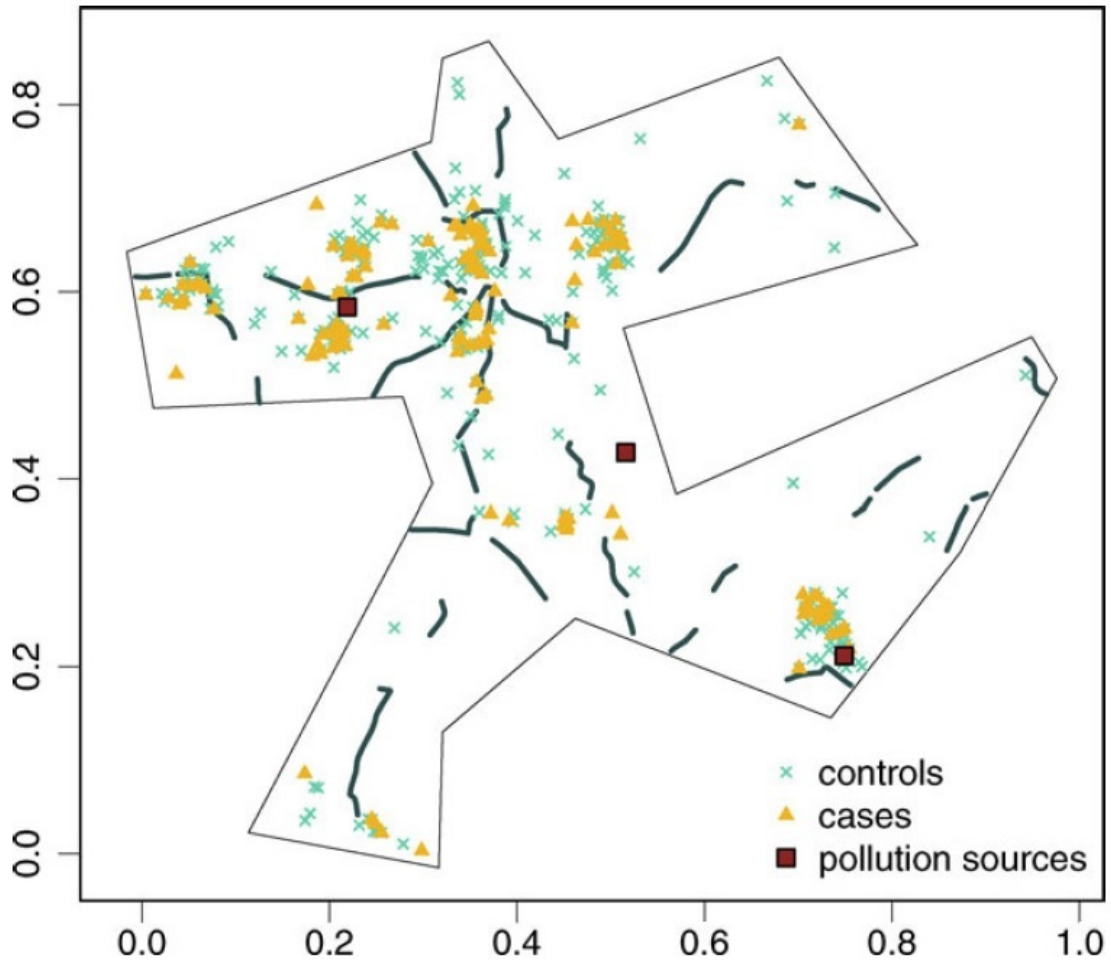

Fig.2: Locais das residências de crianças asmáticas (casos, triângulos laranjas) e não asmáticos (controles, cruzes verdes). O mapa também mostra as fontes de poluição (quadrados marrons) e as rodovias (linhas cinzas).

Análise preliminar.

A análise de padrões pontuais é focada na distribuição espacial dos eventos observados e em realizar inferência sobre o processo subjacente que gerou os pontos.

Existem dois pontos de interesse:

- distribuição de eventos no espaço;
- existência de possíveis interações entre os eventos.

Ferramenta descritiva inicial:
representar os locais do padrão pontual na área de estudo.

Outras ferramentas descritivas também podem ser usadas
conforme discutido a seguir.

Aleatoriedade Espacial Completa (AEC).

Processo de Poisson Homogêneo:
O número de eventos, em qualquer região A, segue a distribuição Poisson(λ).

No estudo de processos pontuais, o teste mais básico que podemos realizar é sobre a existência de AEC.

Sob AEC, se temos n eventos em A, eles serão distribuídos independentemente ao acaso e uniformemente em A. Isto implica que não há regiões em que os eventos são mais (ou menos) prováveis de ocorrer. Além disso, a presença de um evento não modifica a probabilidade de que outros eventos apareçam nas proximidades.

Informalmente, AEC pode ser testada com um gráfico do padrão pontual em que observamos se os pontos tendem a formar clusters ou se eles seguem um padrão regular. Nestes casos, os pontos não são distribuído uniformemente (isso ocorreria se os pontos se espalhassem por todo espaço de estudo).

Em geral, padrões de clusters ocorrem quando há atração (contágio) entre os pontos, enquanto que padrões regulares ocorrem quando há repulsão (competição) entre pontos.

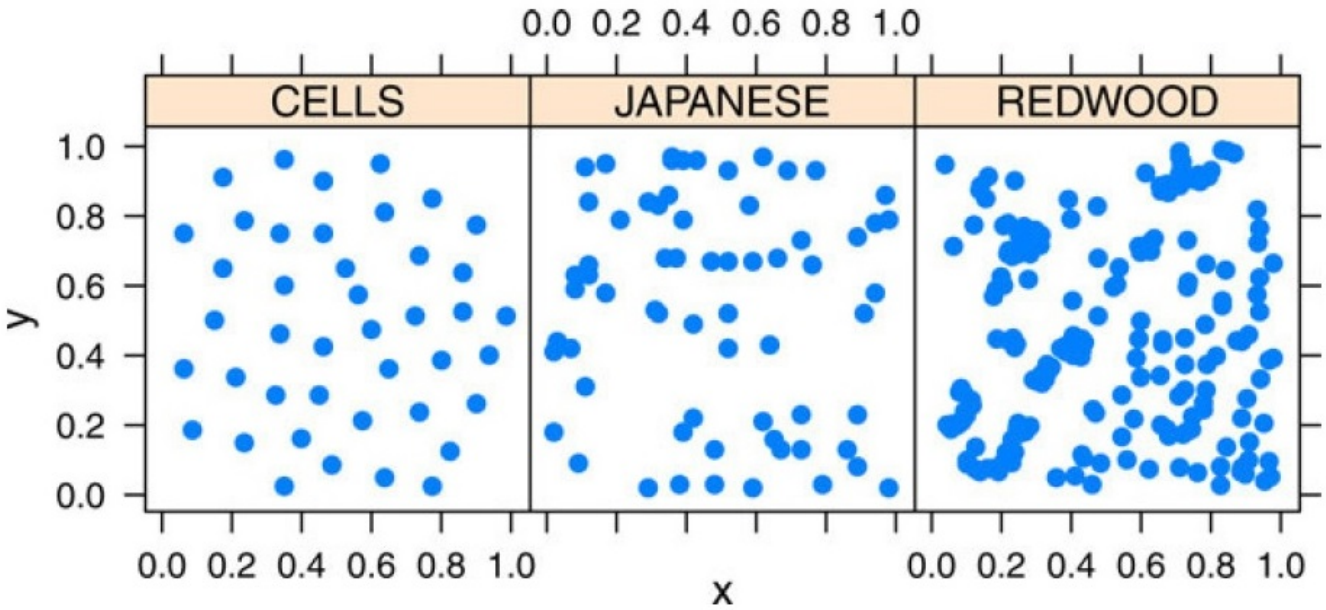

Fig.3: Dados de pinheiros japoneses (sem cluster, sem padrão regular), dados árvores Redwood (cluster), dados de células (padrão regular).

Parece que somente a distribuição dos pontos referentes aos pinheiros japoneses é compatível com a hipótese de AEC. Para medir o grau de aproximação para um AEC, algumas funções podem ser calculadas com base nos dados.

Função G: Distância ao evento mais próximo.

Esta função mede a distribuição das distâncias de um evento arbitrário (local de uma criança com asma) até o evento (asma) mais próximo.

Defina estas distâncias como: $d_i = \min_j\{d_{ij},\ \forall\, j \neq i\}, \quad i = 1, 2, ..., n.$

A Função G é estimada por:

$$\hat{G}(r) = \frac{\#\{d_i : d_i \leq r,\ \forall i\}}{n}$$

Numerador = número de distâncias $\leq r$ no conjunto de distâncias registradas.
Denominador = n é o número de eventos (pontos).

Sob AEC, o valor da Função G será $G(r) = 1 - \exp\{-\lambda\pi r^2\}$, sendo λ o número médio de eventos por unidade de área (intensidade).

Sob AEC (Processo Poisson Homogêneo) o número de pontos (Y) em um círculo de centro u e raio r segue a Poisson(μ) com média $\mu = \lambda(\text{Área})$, ou seja, $\mu = \lambda(\pi r^2)$. Probabilidade de haver pontos nesta região:
$P(Y > 0) = 1 - P(Y = 0) = 1 - \exp\{-\mu\}$.

A compatibilidade do padrão pontual com a hipótese de AEC pode ser verificada através de um gráfico de $\hat{G}(d)$ vs. a esperança teórica.

Envelopes pontuais sob AEC podem ser calculados com a simulação de processos pontuais AEC assumindo a intensidade estimada $\hat{\lambda}$ na área de estudo. Observaremos se a função empírica está contida dentro dos envelopes.

Envelopes são calculados para cada valor de r (ordenamos os `nsim` valores simulados e tomamos o m-ésimo menor e o m-ésimo maior valores; m = `nrank` no script). Exemplo: Se `nrank` $= 1$, o menor e o maior valor simulado formarão o envelope.

Teste Monte Carlo: H_0 : AEC é válido. Rejeitamos H_0 se o valor observado de $G(r)$ cai fora do envelope construído para r. Nível de significância $\alpha = 2\,\text{nrank}/(\text{nsim} + 1)$

```
> set.seed(120109)
> r <- seq(0, sqrt(2)/6, by = 0.005)

> envjap <- envelope(as(spjpines1, "ppp"), fun = Gest, r = r,
  +                  nrank = 2, nsim = 99)

> envred <- envelope(as(spred, "ppp"), fun = Gest, r = r,
  +                  nrank = 2, nsim = 99)

> envcells <- envelope(as(spcells, "ppp"), fun = Gest, r = r,
  +                    nrank = 2, nsim = 99)

> Gresults <- rbind(envjap, envred, envcells)
> Gresults <- cbind(Gresults, y = rep(c("JAPANESE", "REDWOOD", "CELLS"),
  +                 each = length(r)))
```

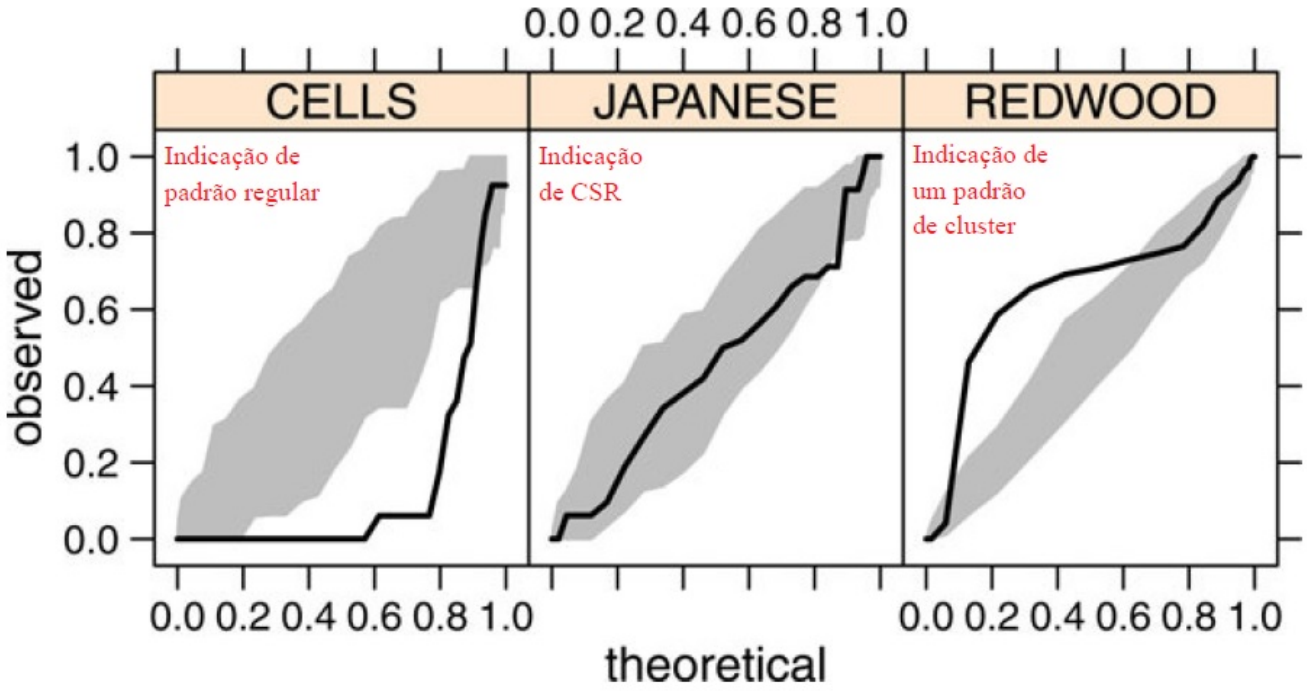

Fig.4: Envelopes e valores observados para a Função G.

A figura acima mostra gráficos de $\hat{G}(r)$ *vs*. $G(r)$ junto com envelopes pontuais de 96% (`nrank` $= 2$). No eixo x temos os valores teóricos de $G(r)$ sob AEC e no eixo y a função empírica $\hat{G}(r)$.

- Dados de pinheiros japoneses parecem ser homogeneamente distribuídos;
- Dados Redwood mostram um padrão de cluster;
  valores de $\hat{G}(r)$ acima dos envelopes;
- Dados sobre locais de células indicam um padrão regular;
  valores de $\hat{G}(r)$ abaixo dos envelopes.

A função `envelope` do R pode ser usada para calcular os envelopes Monte Carlo para um certo tipo de função. Os seguintes passos são seguidos:

1. simular aleatoriamente diversos padrões pontuais;
2. calcular $\hat{G}(r)$ para cada um deles;
3. os resultados são usados para obter envelopes Monte Carlo pontuais (para diferentes distâncias r).

Função F: Distância de um ponto (arbitrário) ao evento mais próximo.

Esta função mede a distribuição de todas as distâncias a partir de um ponto arbitrário (um local qualquer, mesmo sem registro de asma) até o evento mais próximo.

Ela é frequentemente chamada de função do espaço vazio (*empty space function*) visto que é uma medida do espaço médio entre dois eventos.

Sob AEC (Processo Poisson Homogêneo em $\mathbb{R}^2$), o valor esperado da Função F será: $F(r) = 1 - \exp\{-\lambda\pi r^2\}$.

Fórmulas: $\hat{F}(r)$ e $\hat{G}(r)$ são similares. Para $\hat{F}(r)$ temos $n =$ quantidade de pontos arbitrários.

Podemos comparar o valor estimado da Função F com seu valor teórico e calcular envelopes como feito para a Função G.

```
> set.seed(30)
> Fenvjap <- envelope(as(spjpines1, "ppp"), fun = Fest, r = r,
  +                   nrank = 2, nsim = 99)

> Fenvred <- envelope(as(spred, "ppp"), fun = Fest, r = r,
  +                   nrank = 2, nsim = 99)

> Fenvcells <- envelope(as(spcells, "ppp"), fun = Fest,
  +                     r = r, nrank = 2, nsim = 99)

> Fresults <- rbind(Fenvjap, Fenvred, Fenvcells)
> Fresults <- cbind(Fresults, y = rep(c("JAPANESE", "REDWOOD",
+ "CELLS"), each = length(r)))
```

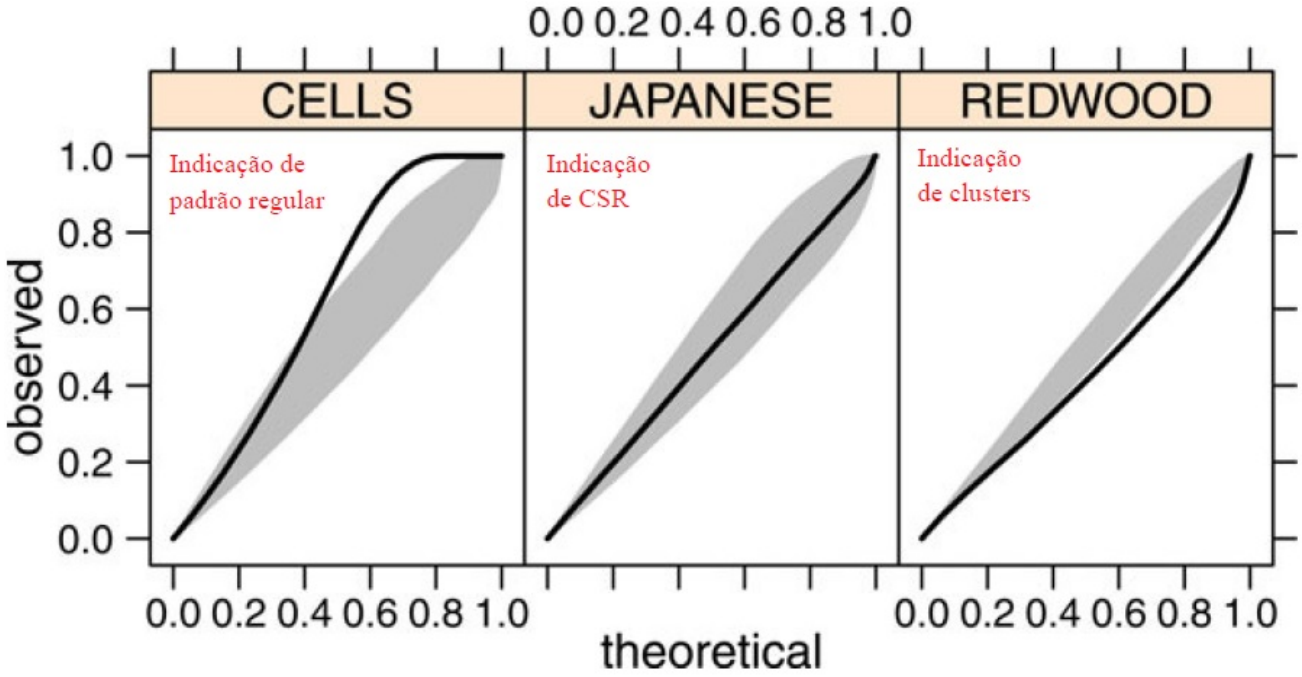

Fig.5: Envelopes e valores observados para a Função F.

A figura mostra gráficos de $\hat{F}(r)$ *vs*. $F(r)$ junto com envelopes pontuais de 96% (`nrank` $= 2$). Os dados dos pinheiros japoneses são compatíveis com a hipótese de AEC. Os dados de células mostram um padrão regular ($\hat{F}(r)$ acima dos envelopes). Os dados Redwood indicam a existência de cluster ($\hat{F}(r)$ abaixo dos envelopes).

Veja que o comportamento da curva obtida via dados reais difere daquele mostrado na figura para a Função G.

Em um padrão regular de eventos, pontos (não eventos) arbitrários podem estar muito próximos de pontos eventos, implicando em $F(r) < \hat{F}(r)$.

Em um padrão "eventos em cluster", pontos (não eventos) arbitrários podem estar em áreas (longe de clusters) sem eventos nas proximidades, implicando em $F(r) > \hat{F}(r)$.

Análise Estatística de Processos Pontuais Espaciais.

Uma primeira descrição do padrão pontual pode ser feita através da estimação da densidade estatística.

Trabalharemos com a intensidade $\lambda(x)$ do processo pontual, a qual é proporcional a sua densidade espacial. A constante de proporcionalidade é o número esperado de eventos do processo pontual na área A.

A intensidade e a densidade espacial são parte de propriedades de primeira ordem que medem a distribuição de eventos na região de estudo. Estes dois elementos não fornecem qualquer informação sobre interação entre dois pontos arbitrários; isto é medido por propriedades de segunda ordem que refletem tendências para que os eventos apareçam aglomerados, independentes ou regularmente espaçados.

A separação entre propriedades de primeira e segunda ordem pode ser difícil. Por exemplo, grupos de eventos aparecem em um local específico pela intensidade alta ou pela tendência de cluster?

Trabalharemos com o Processo Poisson pois ele oferece uma abordagem simples para tratar dados com padrão pontual.

Dois tipos de Processo Poisson:

- Homogêneo (PPH); assume $\lambda(x)$ constante.
- Não-homogêneo (PPN); assume $\lambda(x)$ variando com x.

Cuidado! Note que outros processos espaciais podem ser necessários quando dados mais complexos são analisados. Quando eventos formam cluster, os pontos não ocorrem independentes uns dos outros.

Processo Poisson Homogêneo (PPH).

Processos pontuais nos quais todos os eventos são independentemente e uniformemente distribuídos na região de interesse A.

O local de um ponto não afeta a probabilidade de outros pontos aparecerem na vizinhança. Não há regiões onde eventos são mais prováveis de ocorrerem.

Diggle (2003) descreve o PPH em uma região A como:

1. O número de eventos em A, com área $|A|$, tem distribuição Poisson$(\lambda|A|)$, sendo λ a intensidade constante do processo.
2. Dado n eventos observados na região A, eles são uniformemente distribuídos em A.

O PPH é também estacionário e isotrópico. Portanto, o processo pontual possui intensidade constante e sua intensidade de segunda ordem (descrição mais adiante) depende apenas da distância entre os pontos, sem relação com a posição relativa dos pontos.

Estas restrições implicam que a intensidade do processo pontual é constante, isto é, $\lambda(x) = \lambda > 0$, $\forall\, x \in A$. Os eventos aparecem independentemente uns dos outros. Portanto, o PPH é a definição formal de um processo pontual AEC.

Processo Poisson Não-homogêneo (PPN).

Exemplos: distribuição da população em uma cidade ou localização de árvores em uma floresta. Em ambos os casos, diferentes fatores afetam a distribuição espacial. No caso da cidade, pode ser o tipo de moradia, bairro, etc. No caso da floresta, podem ser fatores ambientais como umidade, qualidade do solo, inclinação do terreno entre outros.

O PPN é uma generalização do PPH. Supomos uma intensidade não constante. O mesmo princípio de independência entre eventos é válido, mas agora a variação espacial pode ser mais diversificada com eventos aparecendo com maior frequência em algumas áreas do que em outras.

A intensidade será uma função genérica $\lambda(x)$ que varia espacialmente.

Estimação da intensidade.

A intensidade de um processo pontual PPH é constante.

O problema se resume em estimar uma função constante λ tal que:

- $\int_A \lambda dx =$ valor esperado do número de eventos na região A.
- A integral acima representa o volume sob a superfície definida pela intensidade em A.

Uma vez observado o processo pontual,
teremos os locais de um conjunto de n pontos.

Um estimador não viciado da intensidade será $n/|A|$, sendo $|A| =$ área de A.

A estimação da intensidade de um processo PPN pode ser feita de formas diferentes.

- Não-parametricamente: por meio de suavização kernel.
- Parametricamente: propondo uma função específica para a intensidade cujos parâmetros serão estimados via máxima verossimilhança.

Se observarmos n pontos $\{x_i\}_{i=1}^n$, a forma do estimador via suavização kernel será a seguinte:

$$\hat{\lambda}(x) = \frac{1}{h^2} \sum_{i=1}^{n} \kappa \left(\frac{||x - x_i||}{h} \right) / q(||x||),$$

sendo $\kappa(u)$ uma função kernel bivariada e simétrica, $q(||x||)$ é uma correção de borda para compensar observações faltantes que ocorrem quando x está perto da borda de A, h é a largura de banda medindo o nível de suavização (valores pequenos produzem estimativas erradicas, valores grandes produzem estimativas suaves).

Silverman (1986) fornece uma descrição detalhada de diferentes funções kernel e suas propriedades.

Uma função kernel bastante utilizada é denominada "Quártica" (também chamada de *biweight*):

$$\kappa(u) = \frac{3}{\pi}(1-u^2)^2 \text{ se } u \in (-1,1), \;\; 0 \text{ caso contrário.}$$

na configuração acima temos $u =$ escalar ($u = ||x - x_i||/h$). Se $|u| > 1$, normalize o vetor original $v = (x - x_i)$ ou escolha outro h. Para normalizar faça: $v^* = v/||v||$.

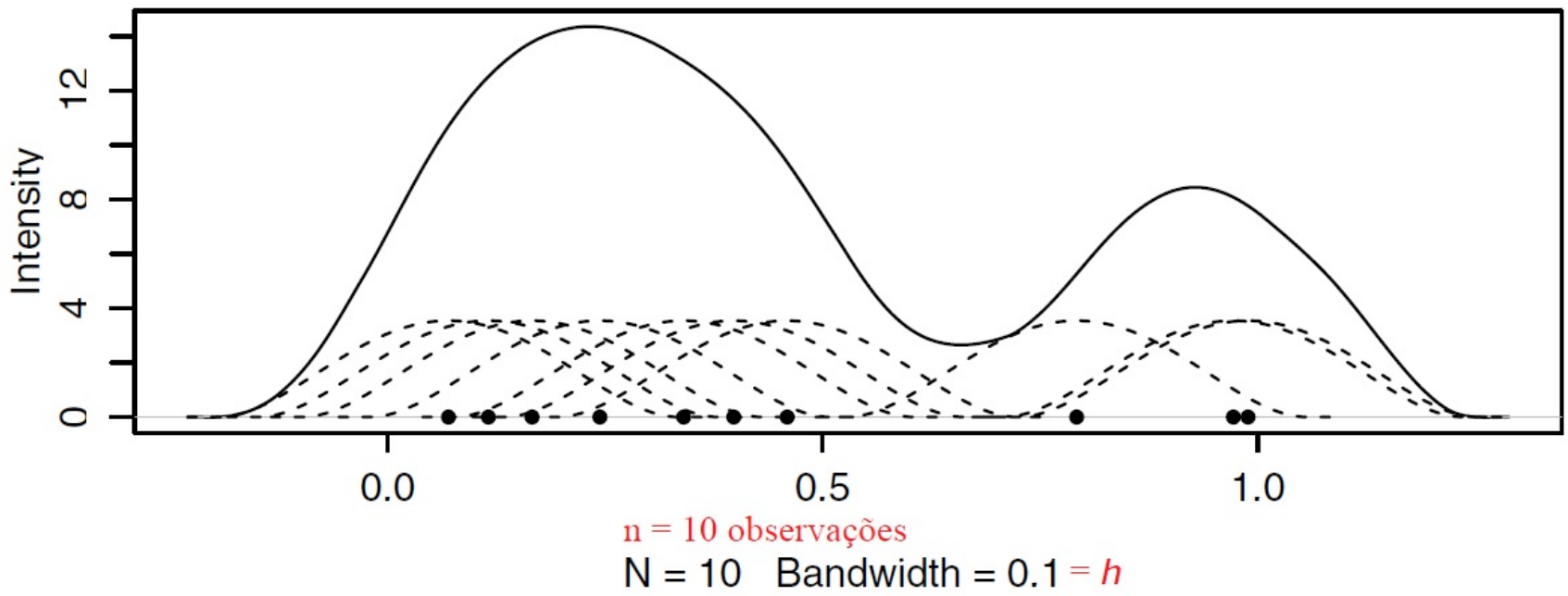

Fig.6: Estimação da intensidade via suavização kernel no caso unidimensional. Tracejado representa a função kernel em torno de cada observação (contribuição dos diferentes pontos para a estimativa da intensidade). Linha sólida representa a estimativa da intensidade.

No contexto da análise espacial, não está claro como escolher um valor ótimo para h. Use vários valores (dependendo do processo sendo considerado) e escolha o valor que pareça mais plausível. Diggle (1985) e Berman e Diggle (1989) propõe um critério baseado na minimização do Erro Quadrático Médio (EQM) do estimador via função kernel.

Aplicaremos a seguir (para os dados de árvores Redwood) a abordagem proposta por Berman e Diggle (1989) que está implementada nas funções `mse2d (splancs)` e `bw.diggle (spatstat)`. Estas funções dependem de `kernel2d` (implementando a kernel Quártica) e `density` (implementando uma outra função kernel chamada de Gaussiana).

```
> library(splancs)
> mserwq <- mse2d(as.points(coordinates(spred)), as.points(list(x =
+                 c(0, 1, 1, 0), y = c(0, 0, 1, 1))), 100, 0.15)

> bwq <- mserwq$h[which.min(mserwq$mse)]
> bwq
[1] 0.039

> mserw <- bw.diggle(as(spred, "ppp"))
> bw <- as.numeric(mserw)
> bw
[1] 0.01977539
```

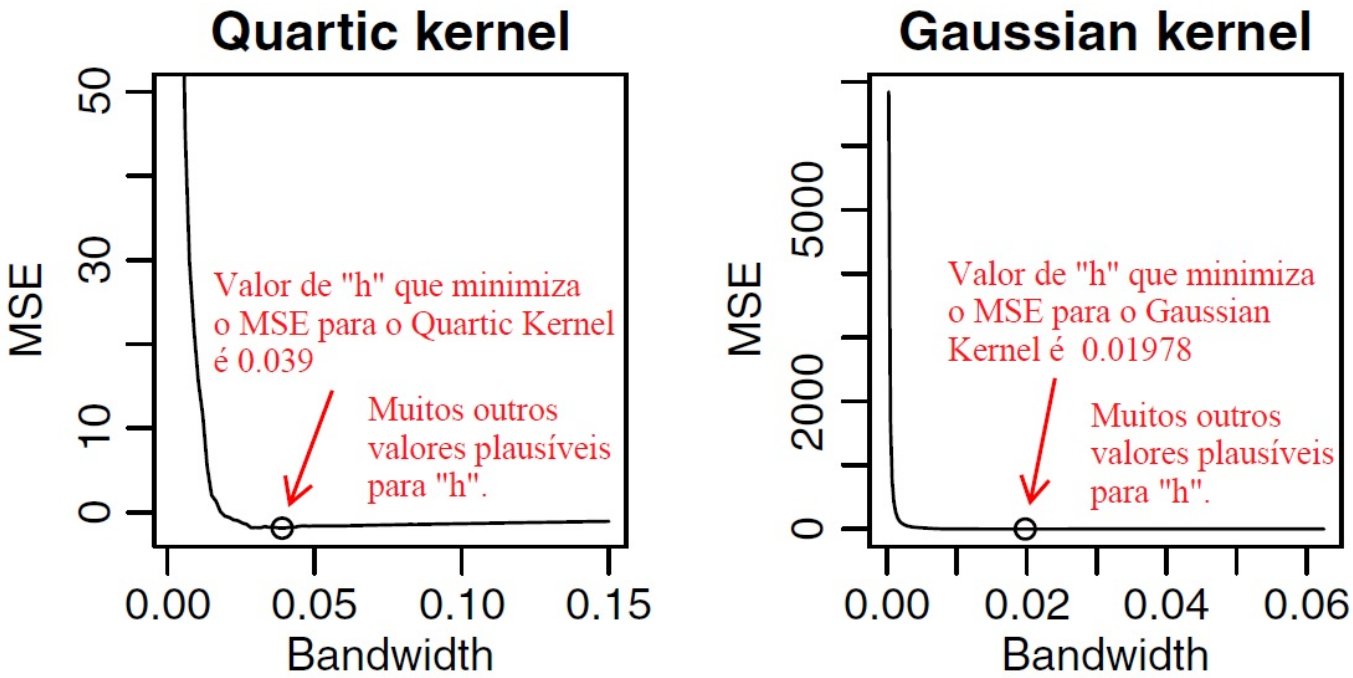

Fig.7: Valores do EQM para diversos valores de h (dados Redwood). Os valores que minimizam o EQM são 0.039 (kernel Quártico) e 0.0198 (Kernel Gaussiano), mas muitos outros valores são plausíveis, dado a estabilização da curva.

Quando estimamos a intensidade via suavização kernel, a escolha principal não é a função kernel, mas sim o valor de h. Funções kernels diferentes podem produzir estimativas similares (com h equivalentes), mas a mesma função kernel (com diferentes h) irá produzir resultados muito diferentes. Isto é mostrado a seguir...

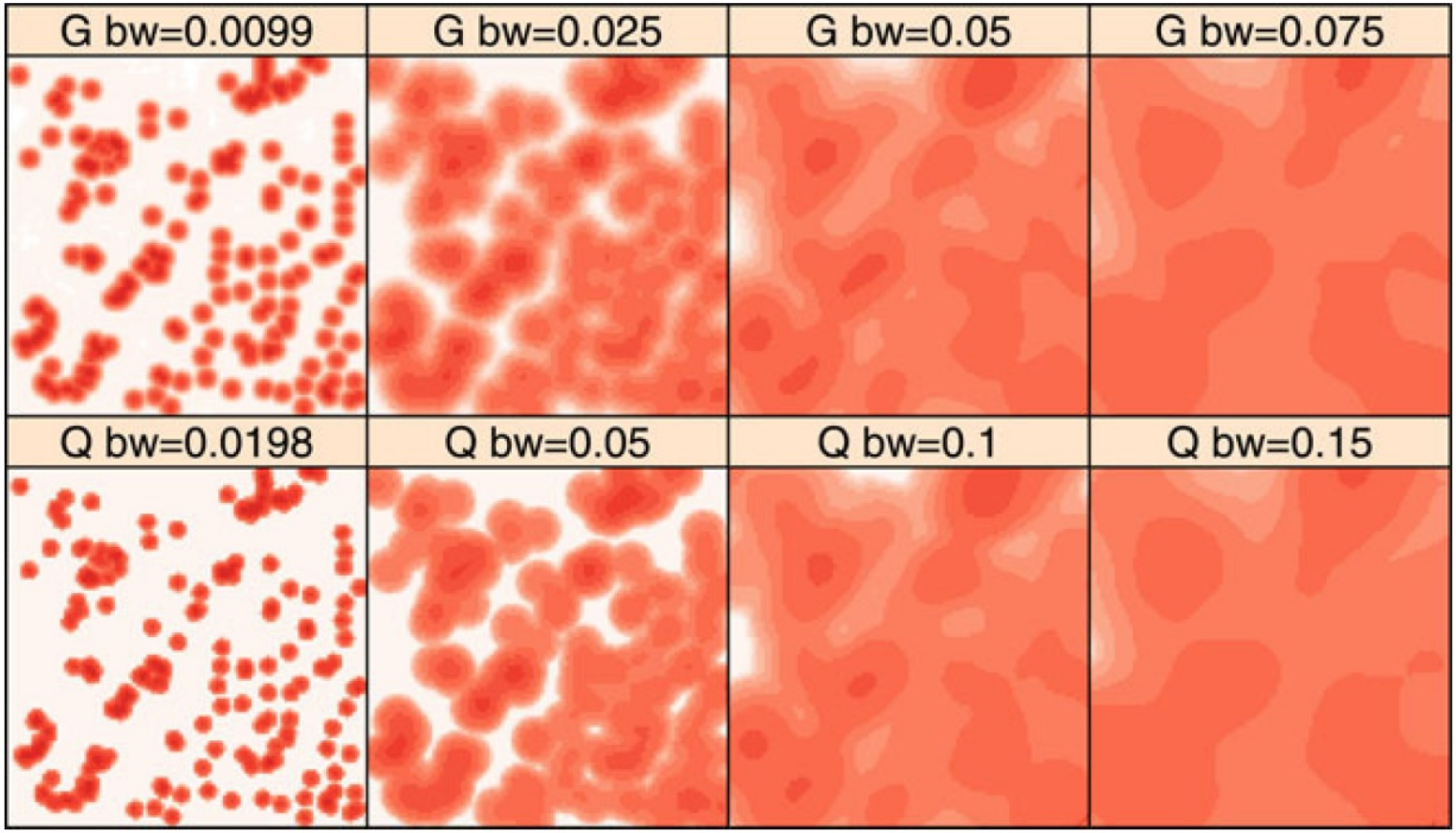

Fig.8: Diferentes estimativas da intensidade (para os dados Redwood) usando os kernels Quártico (Q) e Gaussiano (G) com diferentes valores de h.

Comparando "Q" e "G" temos estimativas similares para escolhas similares de h. Em um kernel, diferentes valores de h produzem estimativas bem distintas.

Suavização via kernel Quártico com a função `spkernel2d` (pacote `splancs`).

```
> library(splancs)
> poly <- as.points(list(x = c(0, 0, 1, 1), y = c(0, 1, 1, 0)))
> sG <- Sobj_SpatialGrid(spred, maxDim = 100)$SG
> grd <- slot(sG, "grid")
> summary(grd)
> k0 <- spkernel2d(spred, poly, h0 = bw, grd)
> k1 <- spkernel2d(spred, poly, h0 = 0.05, grd)
> k2 <- spkernel2d(spred, poly, h0 = 0.1, grd)
> k3 <- spkernel2d(spred, poly, h0 = 0.15, grd)
> df <- data.frame(k0 = k0, k1 = k1, k2 = k2, k3 = k3)
> kernels <- SpatialGridDataFrame(grd, data = df)
> summary(kernels)
```

O pacote `spatstat` fornece funções similares para estimar a intensidade via função kernel Gaussiana isotrópica. Nós ajustamos h empiricamente para tornar as estimativas comparáveis. Ao chamar `density` para um objeto ppp, usamos os argumentos adicionais `dimxy` e `xy` para que o grid usado na estimação seja compatível com aquele definido em `kernels`. Finalmente a estimativa kernel é retornada em uma classe `im` que será convertida em `SpatialGridDataFrame` e incorporada em `kernels`

```
> cc <- coordinates(kernels)
> xy <- list(x = cc[, 1], y = cc[, 2])
> k4 <- density(as(spred, "ppp"), 0.5*bw, dimyx=c(100,100), xy=xy)
> kernels$k4 <- as(k4, "SpatialGridDataFrame")$v
> k5 <- density(as(spred, "ppp"), 0.5*0.05, dimyx=c(100,100), xy=xy)
> kernels$k5 <- as(k5, "SpatialGridDataFrame")$v
> k6 <- density(as(spred, "ppp"), 0.5*0.1, dimyx=c(100,100), xy=xy)
> kernels$k6 <- as(k6, "SpatialGridDataFrame")$v
> k7 <- density(as(spred, "ppp"), 0.5*0.15, dimyx=c(100,100), xy=xy)
> kernels$k7 <- as(k7,"SpatialGridDataFrame")$v
> summary(kernels)
```

Verossimilhança em um PPN.

O procedimento anterior (suavização kernel) para estimar a intensidade é essencialmente não paramétrico.

Alternativamente, uma forma paramétrica ou semi-paramétrica da intensidade pode ser de interesse (ex. incluindo covariáveis).

Técnicas estatísticas como a maximização da verossimilhança podem ser usadas para estimar os parâmetros que aparecem na expressão da densidade.

A expressão da verossimilhança pode ser complicada para muitos processos pontuais, entretanto, no PPN (e no PPH) ela terá uma expressão simples. A log-verossimilhança de uma realização de n eventos independentes de um PPN com intensidade $\lambda(x)$ será:

$$L(\lambda) = \sum_{i=1}^{n} \log[\lambda(x_i)] - \int_A \lambda(x)\, dx,$$

sendo $\int_A \lambda(x)\, dx = n^o$ esperado de casos do PPN com intensidade $\lambda(x)$ em A.

Seja x_i = um ponto de ocorrência do processo PPN.

$x = \{x_1, \ldots, x_n\}$ é um conjunto de n realizações independentes do processo em uma região finita A.

$\mu = \int_A \lambda(x)\, dx$ = número esperado de eventos do PPN em A.

Considere um grid particionando a região A em subregiões muito pequenas A_j para $j = 1, 2, \ldots, m$, sendo m grande.

Assuma que N_{A_j} = número de ocorrências do PPN em A_j.

Considere:

$$
\begin{aligned}
P(N_{A_j} = 1) &= \frac{\lambda(A_j)^1\, e^{-\lambda(A_j)}}{1!} = \lambda(A_j) e^{-\lambda(A_j)} \\
P(N_{A_j} = 0) &= \frac{\lambda(A_j)^0\, e^{-\lambda(A_j)}}{0!} = e^{-\lambda(A_j)} \\
P(N_{A_j} = k) &\approx 0 \text{ para } k \neq 0 \text{ ou } 1.
\end{aligned}
$$

Verossimilhança (região A):

$$
\begin{aligned}
p(N_{A_1}, N_{A_2}, \ldots, N_{A_m}) &= \prod_{j=1}^{m} \left[\lambda(A_j)\, e^{-\lambda(A_j)}\right]^{N_{A_j}} \left[e^{-\lambda(A_j)}\right]^{1-N_{A_j}} \\
&= \prod_{j=1}^{m} [\lambda(A_j)]^{N_{A_j}} \; e^{-\lambda(A_j)} \\
&= \left[\prod_{i=1}^{n} \lambda(x_i)\right] \left[\prod_{j=1}^{m} e^{-\lambda(A_j)}\right] \\
&= \left[\prod_{i=1}^{n} \lambda(x_i)\right] e^{-\sum_{j=1}^{m} \lambda(A_j)} \\
&= \left[\prod_{i=1}^{n} \lambda(x_i)\right] e^{-\mu}
\end{aligned}
$$

$\log[p(N_{A_1}, \ldots, N_{A_m})] = \sum_{i=1}^{n} \log[\lambda(x_i)] - \mu.$

Diggle (2003) sugere o seguinte modelo linear usando covariáveis $z_j(x)$, para $j = 1, \ldots, p$ medida no local x.

$$\log[\lambda(x)] = \sum_{j=1}^{p} \beta_j z_j(x).$$

Estes modelos podem ser ajustados usando técnicas de integração numérica.

O seguinte exemplo define a log-intensidade (`loglambda`) em certo ponto $x = (x_1, x_2)$ usando a especificação paramétrica dada por.

$$\log[\lambda(x)] = \alpha + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1^2 + \beta_4 x_2^2 + \beta_5 x_1 x_2.$$

Esta expressão pode ser usada para construir a verossimilhança (`L`) de um PPN. A função `adaptIntegrate` (pacote `cubature`) é usada para calcular numericamente a integral que aparece na expressão da verossimilhança.

```
> loglambda <- function(x, alpha, beta) {
+               l <- alpha + sum(beta * c(x, x * x, prod(x)))
+               return(l)
+ }

> L <- function(alphabeta, x) {
+       l <- apply(x, 1, loglambda, alpha = alphabeta[1],
+                  beta = alphabeta[-1])
+       l <- sum(l)
+       intL <- adaptIntegrate(lowerLimit = c(0, 0),
+                  upperLimit = c(1,1), fDim = 1, tol = 1e-08,
+                  f = function(x, alpha = alphabeta[1], beta = alphabeta[-1]) {
+                           exp(loglambda(x, alpha, beta))  })
+       l <- l - intL$integral
+       return(l)
+ }
```

O seguinte exemplo usa dados de locais de macieiras (Lansing Woods data set; Gerard, 1969) para mostrar como ajustar uma intensidade paramétrica. Os parâmetros são estimados via máxima verossimilhança (função `optim`)

```
> library(cubature)
> data(lansing)
> x <- as.points(lansing[lansing$marks == "maple", ])
> optbeta <- optim(par = c(log(514), 0, 0, 0, 0, 0), fn = L,
+                  control = list(maxit = 1000, fnscale = -1), x = x)
```

As estimativas dos coeficientes são:
$\hat{\alpha} = 5.56$, $\hat{\beta}_1 = 5.66$, $\hat{\beta}_2 = -0.963$, $\hat{\beta}_3 = -5.14$, $\hat{\beta}_4 = -1.16$, $\hat{\beta}_5 = 0.959$.

O máximo valor da verossimilhança é 2778.3.

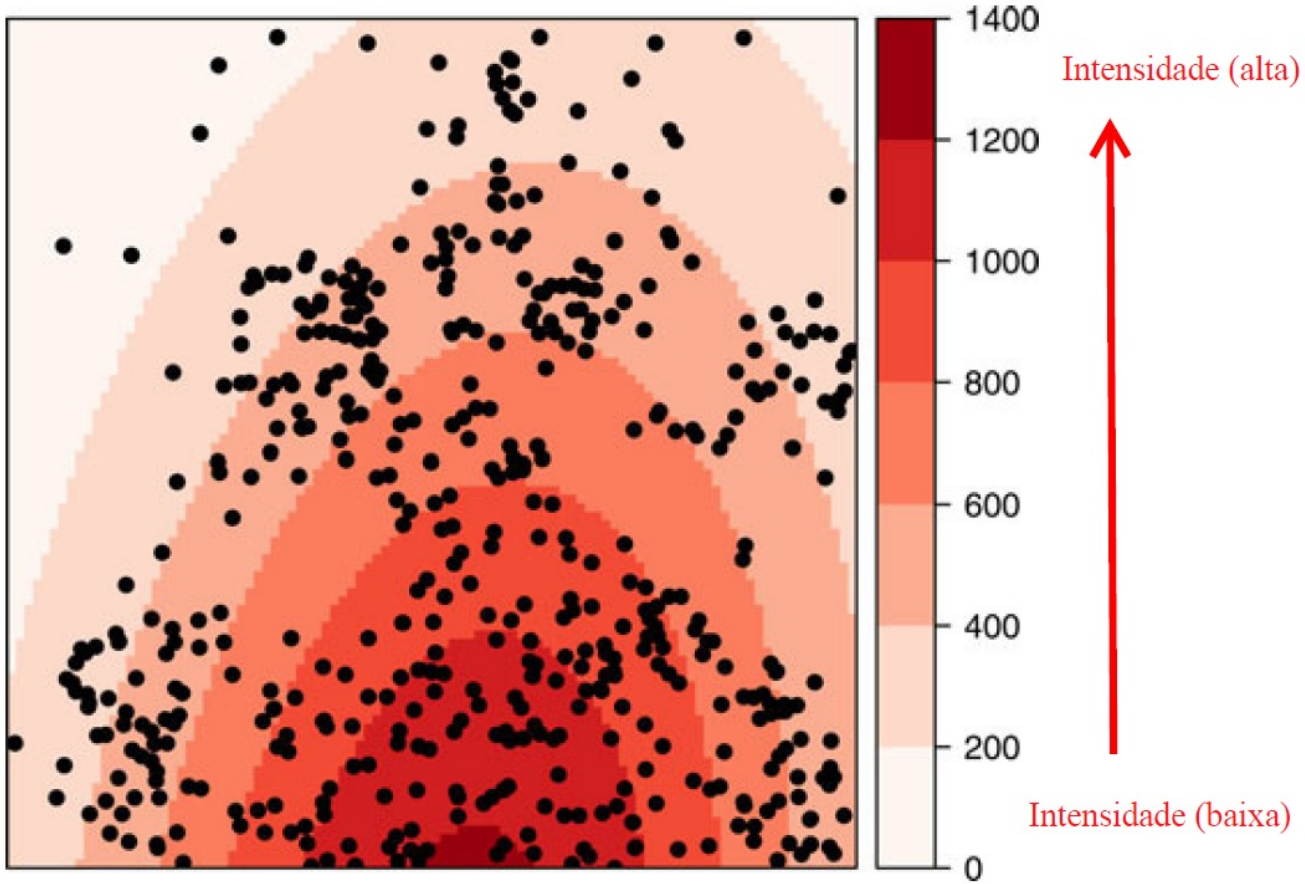

Fig.9: Locais de árvores de maça (Lasing data set) e a intensidade estimada via modelo paramétrico com $\log[\lambda(x)]$ associado a covariáveis.

Este exemplo pode ser avaliado através da função `ppm` do pacote `spatstat` como segue (x e y são coordenadas do processo pontual).

```
> lmaple <- lansing[lansing$marks == "maple", ]
> ppm(Q = lmaple, trend = ~x + y + I(x^2) + I(y^2) + I(x*y))

  Nonstationary multitype Poisson process
  Possible marks:
  blackoak hickory maple misc redoak whiteoak
  Trend formula: ~x + y + I(x^2) + I(y^2) + I(x * y)

  Fitted coefficients for trend formula:
  (Intercept)           x           y      I(x^2)      I(y^2)    I(x * y)
  3.7310742   5.6400643 -0.7663636 -5.0115142 -1.1983209  0.6375824

                Estimate        S.E.  Ztest    CI95.lo      CI95.hi
  (Intercept)  3.7310742  0.2542004     na  3.2328505   4.22929795
            x  5.6400643  0.7990009    ***  4.0740514   7.20607727
            y -0.7663636  0.6990514        -2.1364792   0.60375200
       I(x^2) -5.0115142  0.7011631    *** -6.3857686  -3.63725974
       I(y^2) -1.1983209  0.6428053        -2.4581962   0.06155433
       I(x*y)  0.6375824  0.6989167        -0.7322691   2.00743391
```

Propriedades de segunda ordem.

Estas propriedades medem a força e o tipo de interações entre eventos do processo pontual. Elas são particularmente interessantes para estudarmos a formação de clusters ou a competição entre eventos.

Informalmente, a propriedade de segunda ordem de dois pontos x e y refletem a probabilidade de qualquer par de eventos ocorrendo na vizinhança de x e y.

No processo PPH, a função K é uma forma alternativa de medir as propriedades de segunda ordem. A função K mede o número de eventos encontrados até uma dada distância de qualquer evento particular. Ela é definida como:

$$K(s) = \lambda^{-1} \, E[N_0(s)],$$

sendo $E[.]$ o valor esperado de $N_0(s)$ o número de eventos até uma distância s em torno de um evento arbitrário.

Ripley (1976) também propôs um estimador não viciado igual a:

$$\hat{K}(s) \;=\; [n(n-1)]^{-1}\,|A| \sum_{i=1}^{n}\sum_{j\neq i} w_{ij}^{-1}|\{x_j : d(x_i,x_j)\leq s\}|,$$

sendo $|A|$ = área de A, $d(x_i, x_j)$ = distância entre x_i e x_j.

- Peso w_{ij}: proporção da área do círculo [centro x_i, raio $d(x_i, x_j)$] dentro de A.
- $w_{ij} = 1$: círculo totalmente contido em A.
- $|\{x_j : d(x_i, x_j) \leq s\}| = 1$ se $d(x_i, x_j) \leq s$ (0 c.c.).
- O segundo somatório conta o número de x_j's dentro do círculo de raio s e centro x_i.
- Correção de borda: $w_{ij} = a_{ij}/[2\pi d(x_i, x_j)]$ com a_{ij} = comprimento de arco do círculo [centro x_i, raio $d(x_i, x_j)$] dentro A.

Para um **PPH**, o valor da **Função K** será

$$K(s) = \frac{1}{\lambda} E[N_0(s)] = \frac{1}{\lambda} \lambda \text{ Área} = \pi s^2 = \text{Área do círculo de raio } s.$$

Comparando $\hat{K}(s)$ ao valor teórico acima, podemos avaliar interações. Estas interações ocorrem em escalas pequenas, então estamos interessados em valores pequenos de s; $\hat{K}(s) > \pi s^2$ indica processos com clusters, $\hat{K}(s) < \pi s^2$ indica competição entre eventos (padrão regular).

```
> set.seed(30)
> Kenvjap <- envelope(as(spjpines1, "ppp"), fun = Kest,
+                     r = r, nrank = 2, nsim = 99)
> Kenvred <- envelope(as(spred, "ppp"), fun = Kest, r = r,
+                     nrank = 2, nsim = 99)
> Kenvcells <- envelope(as(spcells, "ppp"), fun = Kest,
+                       r = r, nrank = 2, nsim = 99)
> Kresults <- rbind(Kenvjap, Kenvred, Kenvcells)
> Kresults <- cbind(Kresults, y = rep(c("JAPANESE", "REDWOOD", "CELLS"),
+                   each = length(r)) )
```

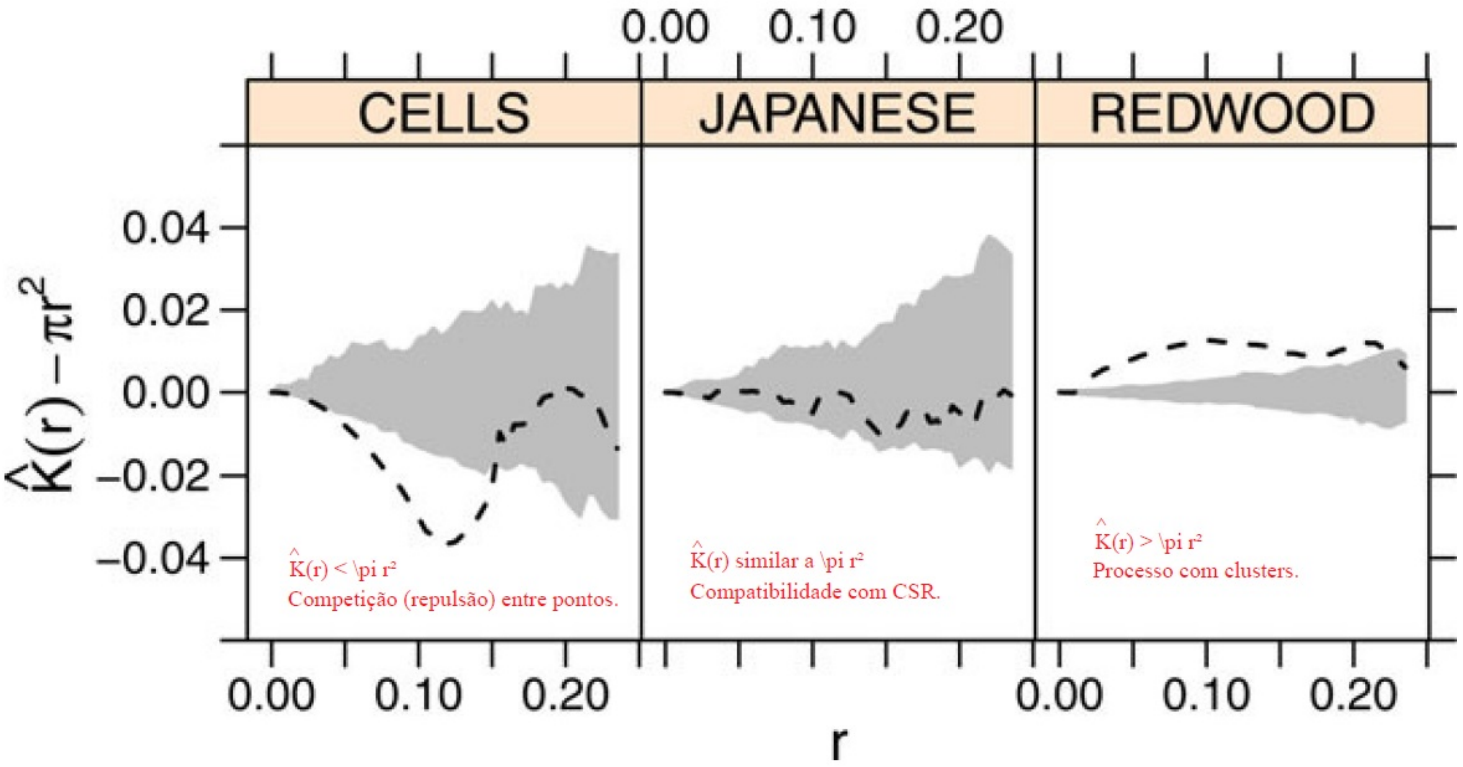

Fig.10: Função K de Ripley estimada menos seus valores teóricos sob AEC (PPH). Três padrões pontuais (locais de células, pinheiros japoneses e árvores Redwood).

A interpretação deve ser cuidadosa, pois os mecanismos geradores do processo são distintos e as escalas de interações (se houver) são provavelmente diferentes. Dados das células indicam padrão regular, Dados dos Pinheiros Japoneses são compatíveis com AEC (Função K dentro dos envelopes), Dados Redwood indicam existência de cluster.

Função K (caso PPN)

Diggle (2007) propôs uma forma similar de avaliar clusters por meio da Função K não-homogênea $K_{I,\lambda}(s)$.

Para um PPN com intensidade $\lambda(x)$ podemos estimar:

$$\hat{K}_{I,\lambda}(s) = |A|^{-1} \sum_{i=1}^{n} \sum_{j \neq i} w_{ij}^{-1} \frac{|\{x_j : d(x_i, x_j) \leq s\}|}{\lambda(x_i)\, \lambda(x_j)}.$$

Note que este é uma generalização do estimador usado na Função K homogênea; ele pode ser reduzido ao caso homogêneo (PPH com $\lambda(x) = \lambda$).

No caso PPN com intensidade $\lambda(x)$, temos também o valor teórico $K_{I,\lambda}(s) = \pi s^2$.

Na prática a intensidade $\lambda(x)$ precisa ser estimada tanto parametricamente ou não parametricamente, então o estimador que usamos será:

$$\hat{K}_{I,\hat{\lambda}}(s) = |A|^{-1} \sum_{i=1}^{n} \sum_{j \neq i} w_{ij}^{-1} \frac{|\{x_j : d(x_i, x_j) \leq s\}|}{\hat{\lambda}(x_i)\, \hat{\lambda}(x_j)}.$$

Veja que usamos, no denominador, estimadores da intensidade estudados anteriormente.

$\hat{K}_{I,\hat{\lambda}}(s) > \pi s^2$ indica maior agregação que a obtida por $\lambda(x)$.
$\hat{K}_{I,\hat{\lambda}}(s) < \pi s^2$ reflete maior homogeneidade relativa.