Finite point processes

To define a finite point process X on a bounded window W, one may specify

- a discrete probability distribution $(p_n)_{n \in \mathbb{N}_0}$ for the total number of points;

- a family of symmetric joint probability densities

$$j_n(x_1, \ldots, x_n),$$

  $n \in \mathbb{N}$, on $(\mathbb{R}^2)^n$ for the locations of the points given that there are n of them.

Density function

The p_n and j_n may be combined in a single function

$$f(\{x_1, \dots, x_n\}) = e^{|W|} n! p_n j_n(x_1, \dots, x_n),$$

the **density function** of X.

The factor $n!$ in the right hand side occurs because f is a function of **unordered sets**, whereas j_n has **ordered vectors** as its argument.

The constant $e^{|W|}$ is a normalisation.

Example – Poisson process

For a Poisson process with intensity function $\lambda : W \to [0,\infty)$,

$$p_n = e^{-\Lambda(W)}\Lambda(W)^n/n!,$$

and

$$j_n(x_1,\ldots,x_n) = \prod_{i=1}^{n} \frac{\lambda(x_i)}{\Lambda(W)}$$

where $\Lambda(W) = \int_W \lambda(w)dw$.

Hence

$$f(\{x_1,\ldots,x_n\}) = \exp\left[\int_W (1-\lambda(w))dw\right] \prod_{i=1}^{n} \lambda(x_i).$$

Note: If $\lambda \equiv 1$ then also $f \equiv 1$.

Recovering p_n and j_n

Density function f is defined uniquely in terms of p_n and j_n. The reverse is also true. Indeed,

$$p_0 = e^{-|W|} f(\emptyset).$$

For $n \in \mathbb{N}$,

$$p_n = \frac{e^{-|W|}}{n!} \int_W \cdots \int_W f(\{u_1, \ldots, u_n\}) du_1 \cdots du_n$$

and

$$j_n(x_1, \ldots, x_n) = \frac{f(\{x_1, \ldots, x_n\})}{\int_W \cdots \int_W f(\{u_1, \ldots, u_n\}) du_1 \cdots du_n}.$$

Conditional specification

For models with interaction, it is often more convenient to work with the **conditional intensity function**

$$\lambda(u|\mathbf{x}) = \frac{f(\mathbf{x} \cup \{u\})}{f(\mathbf{x})},$$

the conditional probability of finding a point at $u \notin \mathbf{x}$ given configuration $\mathbf{x}$ elsewhere (with $\lambda(u|\mathbf{x}) = 0$ when $f(\mathbf{x}) = 0$.)

When $f > 0$,

$$f(\{x_1, \ldots, x_n\}) = f(\emptyset) \prod_{i=1}^{n} \lambda(x_i|\{x_1, \ldots, x_{i-1}\}).$$

Example – Poisson process

For a Poisson process with intensity function $\lambda : W \to [0, \infty)$,

$$\begin{aligned} \lambda(u|\mathbf{x}) &= \frac{f(\mathbf{x} \cup \{u\})}{f(\mathbf{x})} \\ &= \frac{e^{|W| - \Lambda(W)} \lambda(u) \prod_{i=1}^{n} \lambda(x_i)}{e^{|W| - \Lambda(W)} \prod_{i=1}^{n} \lambda(x_i)} \\ &= \lambda(u). \end{aligned}$$

Interaction

The presence of a point at location $w \in W$ may influence the likelihood of finding points 'nearby', e.g.

- points v for which $||w - v|| \leq R$ for some $R > 0$;

- points v in a zone $Z(w) \subset W$ around w.

If the zones $Z(\cdot)$ are not balls, the model is **anisotropic**.

Pairwise interaction models

A **pairwise interaction process** X is a point process whose density function is of the form

$$f(\mathbf{x}) \propto \prod_{x \in \mathbf{x}} \beta(x) \prod_{\{u,v\} \subset \mathbf{x}} \gamma(u, v)$$

for some function $\beta : W \to \mathbb{R}^+$ and some symmetric function $\gamma : W \times W \to \mathbb{R}^+$.

The function β governs the **heterogeneity** or **trend**, γ the **interaction**.

Example: Strauss process

$$\gamma(u,v) = \begin{cases} \gamma & \text{if } ||u-v|| < R \\ 1 & \text{if } ||u-v|| \geq R \end{cases}$$

for $\gamma \in [0,1]$.

$\gamma = 0$ leads to a **hard core process**: no point is allowed to fall within distance R of another point.

$\gamma = 1$ corresponds to a Poisson process.

For intermediate values of γ, points tend to avoid lying closer than R together, the tendency being stronger for smaller values of γ.

Strauss process – conditional intensity

When $f(\mathbf{x}) > 0$,

$$\lambda(u|\mathbf{x}) = \frac{f(\mathbf{x} \cup \{u\})}{f(\mathbf{x})} = \beta(u)\, \gamma^{S(u;\mathbf{x})}$$

where $S(u;\mathbf{x})$ is the number of points in $\mathbf{x}$ that are closer than R to $u \notin \mathbf{x}$.

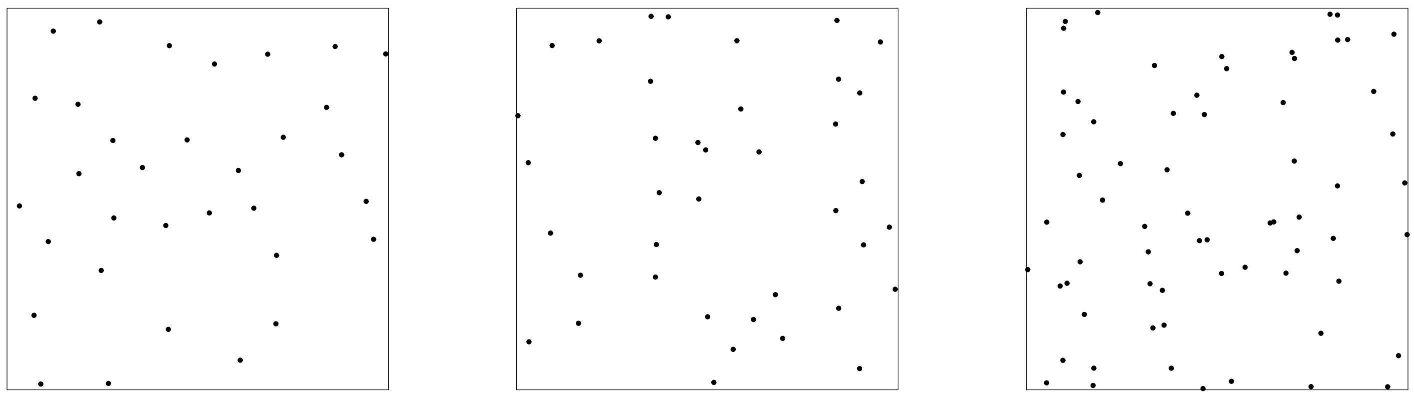
Note that the normalisation constant in f cancels out!

Strauss process – simulation

```
strauss0 <- rStrauss(beta=100, gamma=0.0, R=0.1,
            W=square(1), expand=FALSE, nsim=1)
strauss4 <- rStrauss(beta=100, gamma=0.4, R=0.1,
            W=square(1), expand=FALSE, nsim=1)
strauss8 <- rStrauss(beta=100, gamma=0.8, R=0.1,
            W=square(1), expand=FALSE, nsim=1)
```

If `expand=TRUE`, the simulation is performed on a larger window and clipped. This is appropriate if X is the restriction to W of a point process defined on $\mathbb{R}^2$.

Realisations



Left to right: $\gamma = 0, 0.4$ and 0.8; $R = 0.1$ and $\beta = 100$.

Multi-step process

Piecewise constant pairwise interaction function

$$\gamma(u,v) = \begin{cases} \gamma_j & \text{if } R_{j-1} \leq ||u-v|| < R_j \\ 1 & \text{if } ||u-v|| \geq R_k \end{cases}$$

for $0 = R_0 < R_1 < \cdots < R_k$ and $\gamma_1, \ldots, \gamma_k \in \mathbb{R}$.

For an inhibition strength that decreases in interpoint distance, take

$$\gamma_1 < \cdots < \gamma_k < 1.$$

For attraction combined with a hard core, take

$$\gamma_1 = 0; \gamma_2, \ldots, \gamma_k > 1.$$

Multi-step process – simulation

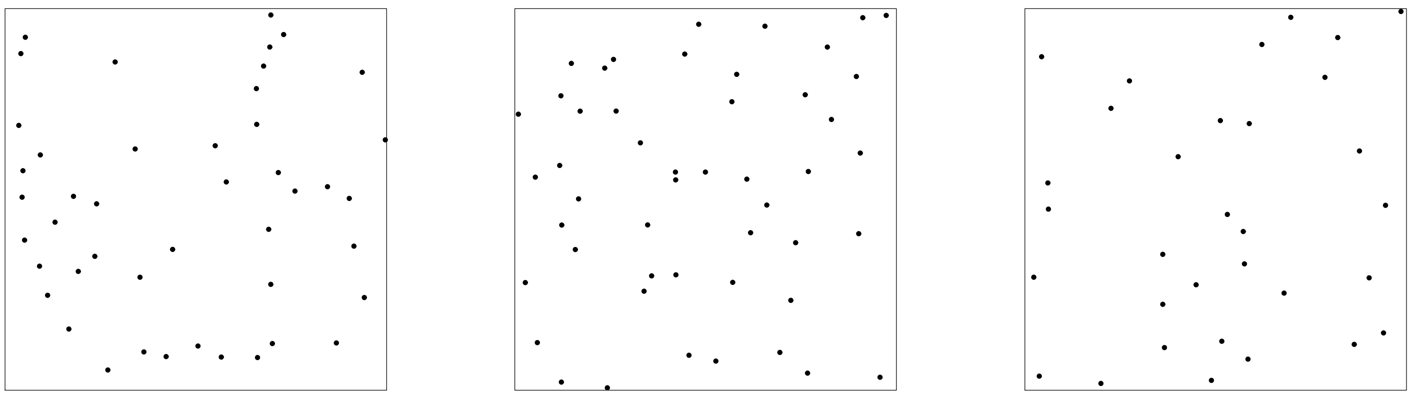
The script

```
r <- seq(0.02, 0.1, by=0.02)
gamma <- c(0.0, 0.2, 0.4, 0.6, 0.8)
ms <- list(beta=100, r=r, h=gamma)
mStep <- rmhmodel(cif="lookup", par=ms, w=square(1))

X <- rmh(mStep,
   start=list(n.start=50), control=list(nrep=1e6))
```

generates an **approximate** realisation of the multi-step process by the Metropolis–Hastings method starting from a binomial point process with 50 points and run for 10^6 iterations.

Realisations



Metropolis–Hastings method

Let $\mathbf{x}_0$ be a realisation of a binomial point process with 50 points. Repeat 10^6 times:

- with probability 1/2, propose to add a new point u to the current pattern $\mathbf{x}$ uniformly on W and accept with probability

$$\min\left\{1, \lambda(u|\mathbf{x})\frac{|W|}{(n(\mathbf{x})+1)}\right\};$$

- with probability 1/2, select one of the current points x_i – if any – with equal probability, propose to delete it and accept this proposal with probability

$$\min\left\{1, \frac{n(\mathbf{x})}{\lambda(x_i|\mathbf{x}\setminus\{x_i\})\,|W|}\right\}.$$

Theorem
If the point process density f is locally stable, the Metropolis–Hastings algorithm on the support $D_f = \{\mathbf{x} : f(\mathbf{x}) > 0\}$ is f-irreducible and f defines an invariant measure.

Influence zone based interaction

Define an **influence function** $\kappa : W \times W \to \mathbb{R}^+$ supported on Z, i.e.

$$Z(x) = \{w \in W : \kappa(w, x) > 0\} \subset W,$$

and write

$$c_{\mathbf{x}}(w) = \sum_{i=1}^{n(\mathbf{x})} \kappa(w, x_i).$$

Then a **shot noise weighted point process** on W with potential function $V(\cdot)$ is defined by

$$f(\mathbf{x}) \propto \beta^{n(\mathbf{x})} \exp\left[-\log\gamma \int_W V(c_{\mathbf{x}}(w))dw\right],$$

where $\beta, \gamma > 0$ and $V : \mathbb{R}^+ \to \mathbb{R}$ with $V(0) = 0$.

Example: Area-interaction process

Let $\kappa(w,x) = 1\{w \in Z(x)\}$. Then

$$c_{\mathbf{x}}(w) = \sum_{i=1}^{n(\mathbf{x})} 1\{w \in Z(x_i)\}$$

is the **coverage function** of $\mathbf{x}$. For $V(x) = 1\{x > 0\}$,

$$f(\mathbf{x}) \propto \beta^{n(\mathbf{x})} \exp\left[-\left|\bigcup_{x\in\mathbf{x}} Z(x)\right| \log\gamma\right].$$

For $\gamma > 1$, realisations tend to be **clustered** to cover a minimum of space.

For $\gamma < 1$, **regular** configurations are favoured.

$\gamma = 1$ corresponds to a Poisson process.

Area-interaction process – conditional intensity

When $f(\mathbf{x}) > 0$,

$$\lambda(u|\mathbf{x}) = \frac{f(\mathbf{x} \cup \{u\})}{f(\mathbf{x})} = \beta\,\gamma^{-|Z(u) \setminus \cup_{x \in \mathbf{x}} Z(x)|}$$

depends only on the area of $Z(u)$, $u \notin \mathbf{x}$, that is not yet covered by some $Z(x)$, $x \in \mathbf{x}$.

Interpretation: For $\gamma > 1$, the conditional intensity $\lambda(u|\mathbf{x})$ is high when $|Z(u) \setminus \cup_{x \in \mathbf{x}} Z(x)|$ is small, i.e. when $Z(u)$ is mostly covered by other influence zones (clustering).

Area-interaction process – simulation

```
aiPar <- list(beta=100, eta=1.5, r=0.1)
ai <- rmhmodel(cif="areaint", par=aiPar, w=square(1))

X1 <- rmh(ai,
   start=list(n.start=50), control=list(nrep=1e6))
```

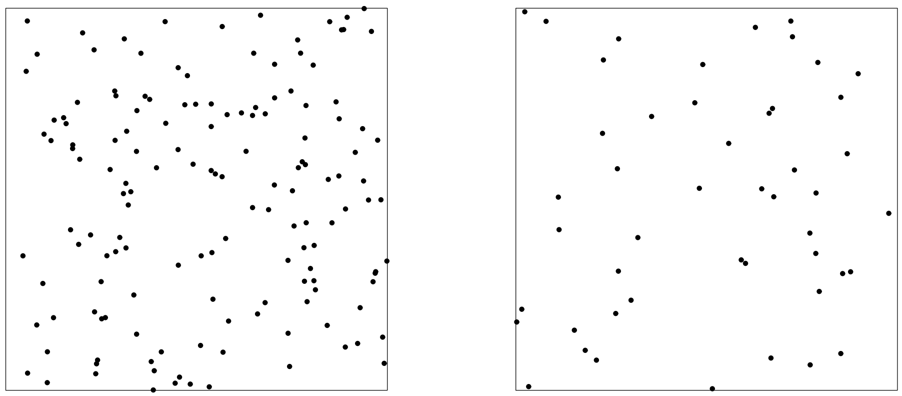
generates an **approximate** realisation of the isotropic model with $Z(w) = B(w, R)$, $R > 0$.

Spatstat uses a parametrisation with

$$\eta = \gamma^{\pi R^2}$$

for numerical stability reasons.

Realisations



Left: $\eta = 1.5$; Right: $\eta = 0.5$. In both cases, $R = 0.1$.

Generalisations

The l_1 function $c_{\mathbf{x}}$ may be replaced by the l_∞ function

$$\tilde{c}_{\mathbf{x}}(w) = \max_{x \in \mathbf{x}} \kappa(w, x).$$

For example, the **multi-step area-interaction model** based on the potential $V(x) = x$ and influence function

$$\kappa(u, v) = \begin{cases} \kappa_j & \text{if } R_{j-1} \leq ||u - v|| < R_j \\ 0 & \text{if } ||u - v|| \geq R_k \end{cases}$$

with $1 = \kappa_1 > \kappa_2 > \cdots > \kappa_k > 0$ is defined by

$$f(\mathbf{x}) \propto \beta^{n(\mathbf{x})} \gamma^{-\int_W \max_{x \in \mathbf{x}} \kappa(w,x) dw}$$

$$= \beta^{n(\mathbf{x})} \gamma^{-\sum_{j=1}^{k} \kappa_j |\{w \in W : d(w,\mathbf{x}) \in [R_{j-1}, R_j)|}.$$

Technical remark

When defining a model by its density function $f(\cdot)$, one needs to make sure that

$$\sum_{n=0}^{\infty} \frac{e^{-|W|}}{n!} \int_W \cdots \int_W f(\{x_1, \ldots, x_n\}) dx_1 \cdots dx_n < \infty.$$

A sufficient condition is that f is **locally stable**: there exists some $\beta > 0$ such that

$$f(\{x_1, \ldots, x_n, x_{n+1}\}) \leq \beta f(\{x_1, \ldots, x_n\})$$

for all $\{x_1, \ldots, x_n\} \subset W$, all $n \in \mathbb{N}$ and all $x_{n+1} \in W$.

Maximum likelihood estimation

Let $\mathbf{x}$ be a realisation of a Strauss process with parameters $\beta(\cdot) \equiv \beta > 0$ and $\gamma \in [0, 1]$ in window $W \subset \mathbb{R}^2$.

Write

$$S(\mathbf{x}) = \sum_{\{u,v\} \subset \mathbf{x}} 1\{||u - v|| < R\}.$$

Then the log likelihood function becomes

$$L(\beta, \gamma) = n(\mathbf{x}) \log \beta + S(\mathbf{x}) \log \gamma - \log Z(\beta, \gamma)$$

but

$$Z(\beta, \gamma) = \sum_{n=0}^{\infty} \frac{e^{-|W|}}{n!} \int_W \cdots \int_W \beta^n \gamma^{S(\{x_1,\ldots,x_n\})} dx_1 \cdots dx_n$$

depends on the parameters and cannot be evaluated explicitly.

Pseudo-likelihood idea

Let $\mathbf{x}$ be a realisation of a finite point process defined by a density function $f(\mathbf{x};\theta)$ that depends on a parameter θ.

Idea: Approximate the log likelihood by that of a Poisson process with intensity function

$$\lambda_\theta(u|\mathbf{x}) = \frac{f(\mathbf{x} \cup \{u\};\theta)}{f(\mathbf{x};\theta)},$$

the conditional probability of finding a point at $u \notin \mathbf{x}$ given configuration $\mathbf{x}$ elsewhere. Here $\lambda_\theta(u|\mathbf{x}) = 0$ when $f(\mathbf{x};\theta) = 0$.

Maximum pseudo-likelihood estimation

The **log pseudo-likelihood function** is defined as

$$PL(\theta) = \sum_{i=1}^{n} \log \lambda_\theta(x_i | \mathbf{x} \setminus \{x_i\}) - \int_W \lambda_\theta(w|\mathbf{x}) \, dw.$$

Optimise numerically over the parameter θ to obtain the maximum pseudo-likelihood estimate $\widehat{\theta}$.

Advantage: $\lambda_\theta(u|\mathbf{x})$ does not depend on the proportionality constant $Z(\theta)$.

Disadvantage: The approximation may be poor when the interaction is strong.

Exponential family models

The models we presented take the form

$$f(\mathbf{x};\theta) = \frac{1}{Z(\theta)} \exp\left[\sum_{j=1}^{p} \theta_j C_j(\mathbf{x})\right],$$

in other words, form an **exponential family** with **sufficient statistics** C_j and parameters θ_j, $j = 1, \ldots, p$.

Hence, for $u \notin \mathbf{x}$,

$$\log \lambda_\theta(u|\mathbf{x}) = \sum_{j=1}^{p} \theta_j \left[C_j(\mathbf{x} \cup \{u\}) - C_j(\mathbf{x})\right]$$

so $PL(\theta)$ reads

$$\sum_{j=1}^{p} \sum_{i=1}^{n} \theta_j \left[C_j(\mathbf{x}) - C_j(\mathbf{x} \setminus \{x_i\})\right] - \int_W e^{\sum_j \theta_j \{C_j(\mathbf{x} \cup \{w\}) - C_j(\mathbf{x})\}} dw.$$

Maximum pseudo-likelihood estimator

Writing

$$C_j(u; \mathbf{x}) = C_j(\mathbf{x} \cup \{u\}) - C_j(\mathbf{x}),$$

the score equations are

$$\int_W C_j(w; \mathbf{x}) \lambda_\theta(w|\mathbf{x}) dw = \sum_{i=1}^n C_j(x_i; \mathbf{x} \setminus \{x_i\})$$

for $j = 1, \ldots, p$.

The Hessian matrix $H(\theta)$ of second order partial derivatives has entries

$$\frac{\partial^2}{\partial \theta_i \partial \theta_j} PL(\theta) = -\int_W C_i(w; \mathbf{x}) C_j(w; \mathbf{x}) \lambda_\theta(w|\mathbf{x}) dw.$$

Note that $H(\theta)$ **does** depend on $\mathbf{x}$.

Maximum pseudo-likelihood estimator – remarks

- In general, the score equations cannot be solved explicitly.

- Any $\widehat{\theta}$ that solves the score equations and for which $H(\widehat{\theta})$ is negative definite is a local maximum of the log pseudo-likelihood function $PL(\theta)$.

- $PL(\theta)$ involves an integral that must be approximated.

- Little is known about the small sample properties of $\widehat{\theta}$.

Increasing window asymptotics

When the window W grows to $\mathbb{R}^2$,

- the limit distribution of $X \cap W$ may not exist;
- if it does, it may depend on boundary conditions so not be unique (**phase transition**).

Asymptotic normality of $\widehat{\theta}$ was proved under strong ergodicity conditions.

Approximate covariance matrix

Write $V(\theta) = \mathrm{Var}\, s(\theta; X)$ for the variance of the score equation

$$s(\theta; X) = \sum_{x \in X} C_j(x; \mathbf{x} \setminus \{x\}) - \int_W C_j(w; X) \lambda_\theta(w|X) dw.$$

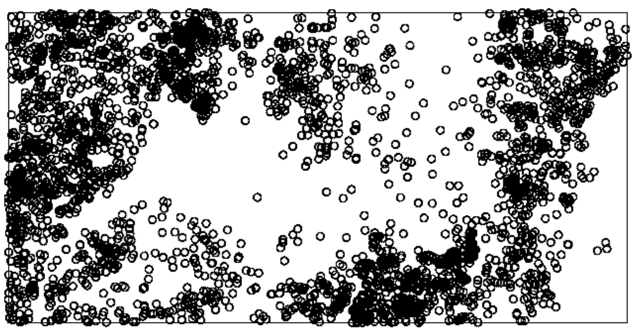
Then, $\widehat{\theta}$ unbiased $\widehat{\theta}$ has approximate (asymptotic) covariance matrix

$$H(\theta)^{-1} V(\theta) H(\theta)^{-1}.$$

This covariance matrix cannot be evaluated explicitly and must be approximated numerically.

Example: Barro Colorado data

`bei` contains the locations of 3604 Beilschmiedia trees in a 1000×500 metre region in the tropical rainforest of Barro Colorado Island (Hubbell and Foster, 1983).



Heterogeneous area-interaction model

A model in which $\log \lambda(x, y)$ is a fourth order polynomial was fitted by

```
fitbeiXY <- ppm(bei ~ polynom(x,y,4))
```

Interaction can be added by by

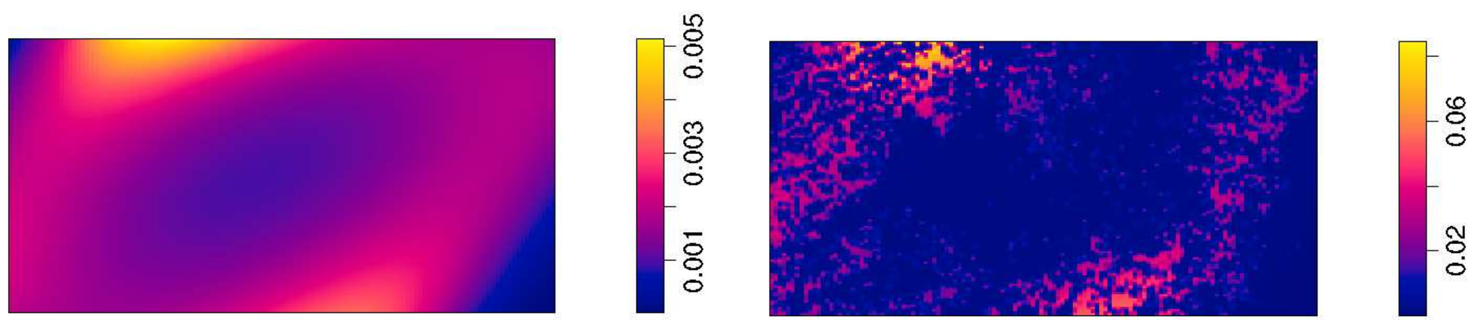
```
fitbeiAI <- update.ppm(fitbeiXY,
   interaction=AreaInter(r=5))
```

which yields

```
Disc radius: 5
Fitted interaction parameter eta:  16.7755
```

Barro Colorado data: Results

```
plot(predict(fitbeiAI, type="trend"))
plot(predict(fitbeiAI, type="cif"))
```



Left: trend (polynomial); Right: cif $\lambda_{\widehat{\theta}}(x|\mathbf{x})$.

Model validation by residuals

A residual analysis is based on

$$s(x) = h^{-2} \sum_{y \in \mathbf{x}} \kappa\left(\frac{x-y}{h}\right) w_h(x,y)^{-1}$$

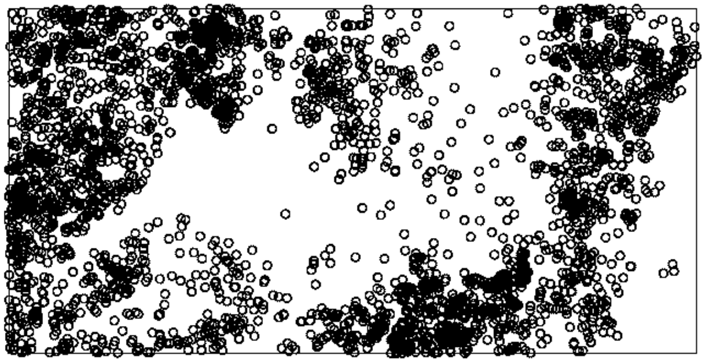
$$-h^{-2} \int_W \kappa\left(\frac{x-w}{h}\right) w_h(x,w)^{-1} \lambda_{\widehat{\theta}}(w|\mathbf{x}) dw,$$

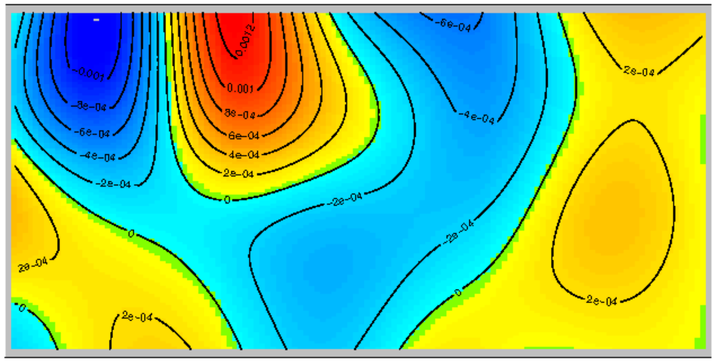
where κ is a probability density function and w_h an edge correction factor.

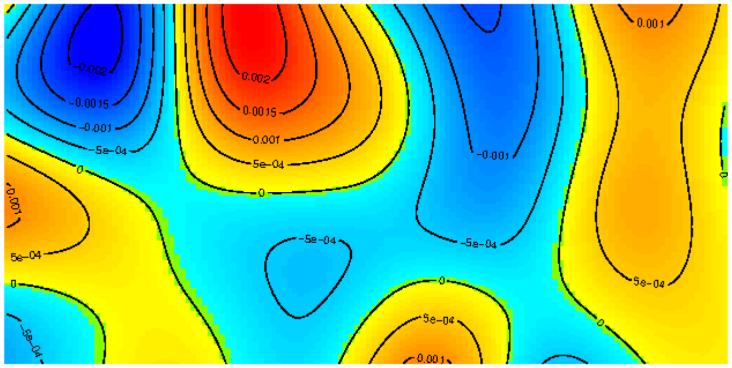
In **spatstat**, use

```
a <- diagnose.ppm(fitbeiAI, which="smooth", sigma=100)
> sum(a$smooth$Z)
[1] -0.2828619
```

Barro Colorado data: Smoothed residuals



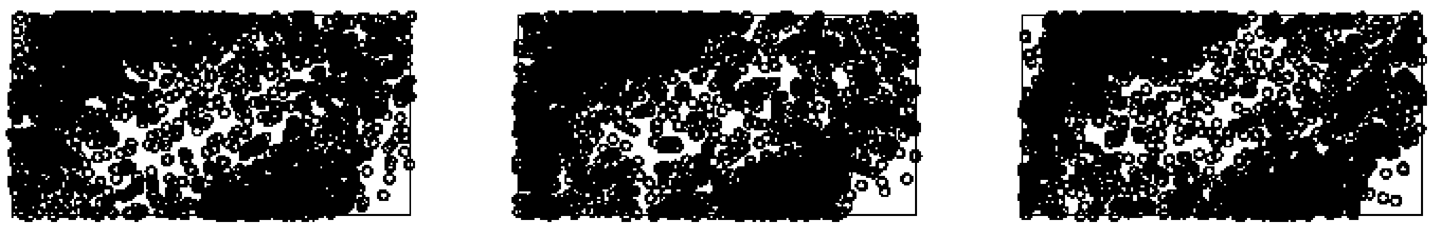




With interaction (top) and without.

Barro Colorado data: Simulations

```
simulate(fitbeiAI, nsim=3)
```



Numerical considerations

Baddeley and Turner (1998) proposed to approximate

$$\int_W \lambda_\theta(w|\mathbf{x})dw \approx \sum_{j=1}^{n} \lambda_\theta(u_j|\mathbf{x})\, w_j,$$

where u_j are **dummy points** in W and w_j are **quadrature weights**.

Then

$$PL(\theta) \approx \sum_{x \in \mathbf{x}} \log \lambda_\theta(x|\mathbf{x} \setminus \{x\}) - \sum_{j=1}^{n} \lambda_\theta(u_j|\mathbf{x})\, w_j.$$

Generalised log-linear Poisson regression model

Add the **data points** $x \in \mathbf{x}$ to the set of dummies to form $\{u_j : j = 1, \ldots, n + n(\mathbf{x}) = m\}$. Then

$$PL(\theta) \approx \sum_{j=1}^{m} (y_j \log \lambda_j - \lambda_j)\, w_j,$$

where $\lambda_j = \lambda_\theta(u_j | \mathbf{x} \setminus \{u_j\})$, $y_j = z_j / w_j$ and

$$z_j = \begin{cases} 1, & \text{if } u_j \in \mathbf{x} \text{ is a data point,} \\ 0, & \text{if } u_j \notin \mathbf{x} \text{ is a dummy point.} \end{cases}$$

Quadrature weights – adaptive weights

Baddeley et al. (2014) proposed the following adaptation of Waagepetersen's adaptive scheme,

$$\int_W \lambda_\theta(w|\mathbf{x})dw \approx \sum_{j=1}^{n} \frac{\lambda_\theta(u_j|\mathbf{x} \setminus \{u_j\})}{\lambda_\theta(u_j|\mathbf{x} \setminus \{u_j\}) + n/|W|},$$

based on approximating the area of the Voronoi cell of u_j in $\mathbf{x}$ by

$$\frac{1}{\lambda_\theta(u_j|\mathbf{x} \setminus \{u_j\}) + n/|W|}.$$

Implementation

When there is strong interaction, the approximations may be off and the results may vary!

```
fitbeiPL <- ppm(bei ~ polynom(x,y,4),
  interaction=AreaInter(r=5), method="mpl")
```

```
Fitted interaction parameter eta: 16.7755
```

```
fitbeiLogi <- ppm(bei ~ polynom(x,y,4),
  interaction=AreaInter(r=5), method="logi")
```

```
Fitted interaction parameter eta: 10.2757
```

Profile likelihood

So far, we fixed the interaction radius R.

It can be estimated by maximising the **profile log pseudo-likelihood**

$$PPL(R) = \max_{\theta} PL(\theta, R) = PL(\widehat{\theta}, R).$$

For the **Barro Colorado data**,

```
r <- data.frame(r=seq(1, 10, by=1))
fitbeiProfile <- profilepl(r, AreaInter,
   bei ~ polynom(x,y,4), aic=FALSE)
```

yields an optimal value $\widehat{R} = 3$.

Remarks on model selection

- The likelihood ratio test and AIC rely on the likelihood so do not apply.

- The theory of **composite likelihood** aka **estimating equations** provides alternative tools.

(Outside the scope of this course).

References

1. A. Baddeley and R. Turner. Practical maximum pseudolikelihood for spatial point patterns (with discussion). *Australian and New Zealand Journal of Statistics* 42:283–322, 2000.
2. A. Baddeley, E. Rubak and R. Turner. Spatial point patterns. Methodology and applications with R. CRC Press, 2016.
3. A. Baddeley, J.-F. Cœurjolly, E. Rubak and R. Turner. Logistic regression for spatial Gibbs point processes. *Biometrika* 101:377–392, 2014.

4. M.N.M. van Lieshout. Markov point processes and their applications. Imperial College Press, 2000. 5. R. Waagepetersen. Estimating functions for inhomogeneous spatial point processes with incomplete covariate data. *Biometrika* 95:351–363.

Assessment

The R-package **spatstat** contains the dataset `swedishpines`.

In a previous assessment, the CSR hypothesis was rejected. Formulate a suitable model with interaction for these data, estimate its parameters and validate your results.