

JIHOČESKÁ UNIVERZITA  
V ČESKÝCH BUDĚJOVICÍCH



PEDAGOGICKÁ FAKULTA

Katedra matematiky

Simulace náhodných množin pomocí počítače

Simulation of Random Sets using a computer

DIPLOMOVÁ PRÁCE

**Diplomant: Radka Pantlíková**

**Vedoucí diplomové práce: RNDr. Tomáš Mrkvička, Ph.D.**

České Budějovice 2009

Prohlašuji, že jsem diplomovou práci zpracovala samostatně a použitou literaturu jsem citovala.

Prohlašuji, že v souladu s § 45b. zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své diplomové práce, a to v nezkrácené podobě, elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách.

V Českých Budějovicích 17.4.2009

Radka Pantlíková

## **Poděkování**

Děkuji vedoucímu mé diplomové práce RNDr. Tomášovi Mrkvičkovi, Ph.D. za prvotní motivaci, odborné vedení, cenné rady a připomínky, které mi v průběhu psaní diplomové práce ochotně poskytl. Dále pak děkuji rodině a přátelům za stálý optimismus a podporu po celou dobu mého studia.

## **Anotace**

Cílem této diplomové práce je vytvoření zdrojových kódů v programu Mathematica potřebných pro simulaci některých náhodných množin a jejich charakteristik. Práce bude zaměřena na vybrané bodové procesy homogenní, nehomogenní, shlukové (cluster) a některé jejich varianty. Tato práce by mohla sloužit jako základ pro další výzkumy spojené s oborem prostorové statistiky.

# OBSAH

1. ÚVOD.....	7
2. TEORIE PROSTOROVÝCH BODOVÝCH PROCESŮ .....	8
2.1 PROSTOROVÝ BODOVÝ PROCES.....	9
2.2 ZÁKLADNÍ OPERACE PRO BODOVÉ PROCESY .....	10
2.3 POISSONŮV BODOVÝ PROCES.....	10
2.4 CLUSTER (SHLUKOVÉ) PROCESY .....	11
2.4.1 <i>Matérn cluster proces</i> .....	11
2.4.2 <i>Thomas cluster proces</i> .....	12
3. CHARAKTERISTIKY BODOVÝCH PROCESŮ .....	13
3.1 ZÁKLADNÍ VLASTNOSTI A VLASTNOSTI DRUHÉHO ŘÁDU.....	13
3.1.1 <i>g-funkce</i> .....	13
3.1.2 <i>Redukovaná momentová míra druhého řádu</i> .....	14
3.2 SOUHRNNÉ STATISTIKY .....	14
3.2.1 <i>Souhrnné statistiky druhého řádu (K a L funkce)</i> .....	14
3.2.2 <i>Souhrnné statistiky založené na vzdálenosti mezi body (F, G, J funkce)</i> ..	15
3.3 NEPARAMETRICKÉ ODHADY.....	15
3.3.1 <i>Neparametrické odhady hustoty funkce</i> .....	16
3.3.2 <i>Neparametrické odhady K a L</i> .....	16
3.3.3 <i>Neparametrické odhady funkce g</i> .....	17
3.3.4 <i>Neparametrické odhady funkcí F,G a J</i> .....	17
3.3.5 <i>Obálky pro souhrnné statistiky</i> .....	18
4. SIMULACE .....	19
4.1 BERNOULLIHO PROCES.....	19
4.2 HOMOGENNÍ POISSONŮV PROCES .....	20
4.3 NEHOMOGENNÍ POISSONŮV PROCES.....	21
4.4 MATÉRN CLUSTER PROCES.....	23
4.5 MATÉRN CLUSTER NEHOMOGENNÍ PROCES.....	25
4.6 MATÉRN CLUSTER PROCES S NEHOMOGENNÍMI CENTRY.....	28
4.7 THOMAS CLUSTER PROCES .....	29
4.8 THOMAS CLUSTER NEHOMOGENNÍ PROCES .....	31
4.9 THOMAS CLUSTER PROCES S NEHOMOGENNÍMI CENTRY.....	33
4.10 ODHAD INTENZITY .....	34
4.11 ODHAD K FUNKCE .....	35
4.11 ODHAD L FUNKCE .....	37
4.12 ODHAD F FUNKCE .....	41
4.13 ODHAD G FUNKCE .....	44
4.14 ODHAD J FUNKCE.....	47
5. PŘÍKLAD VYUŽITÍ V PRAXI.....	50
5.1 DATA A JEJICH PRVOTNÍ ZPRACOVÁNÍ.....	50
5.2 ZOBRAZENÍ DAT .....	50

<b>5.3 VYHODNOCOVÁNÍ A SROVNÁVÁNÍ DAT .....</b>	<b>51</b>
<b>6. ZÁVĚR.....</b>	<b>54</b>
<b>7. POUŽITÁ LITERATURA.....</b>	<b>56</b>
<b>PŘÍLOHY .....</b>	<b>57</b>

# 1. Úvod

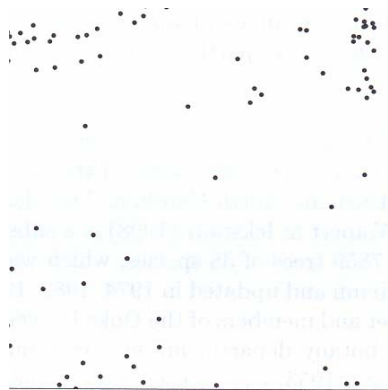
Cílem mojí diplomové práce je, jak už z jejího názvu vyplývá, nasimulovat různé náhodné množiny pomocí počítačového programu. Tímto matematickým softwarem bude program Mathematica, jenž zvládá všechny výpočty i způsoby zobrazení výsledků, které budu potřebovat.

Jakýmsi dílčím úkolem této práce, který jsem si sama stanovila, je dokázat i „nezasvěcenému“ čtenáři, že se nejedná o „matematiku pro matematiku“, ale o praktickou a potřebnou metodu, která se využívá v různých oborech lidské činnosti a s jejichž výsledky se setkáváme i v běžném životě, o čemž se Vás pokusím částečně přesvědčit již v následující kapitole.

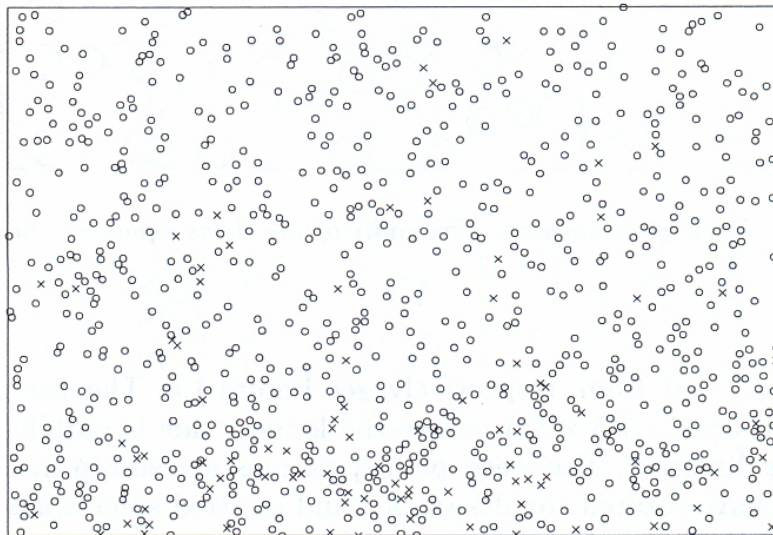
V kapitolách dalších budu muset bohužel začít takřikajíc od nuly, což pro mě znamená zavedení všech pojmů nezbytných pro pochopení problematiky a především naprogramování veškerých zdrojových kódů v programu Mathematica. Tudíž není dost možné obsáhnout v této práci i podrobné zpracování konkrétních příkladů. Doufám, že bude stačit jen hrubé nastínění toho, jak je možné využít tuto teorii pro praktické činnosti a byla bych velmi potěšena, kdyby následující strany někoho zaujaly natolik, že by se rozhodl na moji práci navázat právě ve zkoumání již zmiňovaných praktických úkolů.

## 2. Teorie prostorových bodových procesů

Prostorové bodové procesy se využívají k modelování náhodných událostí v prostoru, kde jsou body umístěny v typických pozicích nebo v centru objektů ve dvojdimenzionální nebo trojdimenzionální oblasti. Okruh jejich aplikace se tudíž neomezuje pouze na matematiku, ale zahrnuje i obory jako je např. astronomie (poloha hvězd nebo galaxií), meteorologie (směry větrů), geografie (rozmístění lidských sídel), lesnictví (ukázkou využití zachycuje obr. 2.1), seismologie (epicentra zemětřesení), biologie (ukázkou využití zachycuje obr. 2.2), zemědělství (rozložení plevele) a prostorová epidemiologie (výskyt infikovaných pacientů).



Obr. 2.1: Pozice stromů bílého ořechu v Dukově lese (převzato z publikace [1], str. 2).



Obr. 2.2: Pozice dvou různých druhů buněk (rozlišeny značkou křížku a kruhu) v místě křížení sliznice a žaludku u potkana (převzato z publikace [1], str. 3).



Problematika bodových procesů je, vzhledem k ostatním matematickým disciplínám, poměrně novým oborem. Rozvoj výzkumů v této oblasti nastal s rozvojem výpočetní techniky. V minulých několika desítkách let zastávaly bodové procesy dokonce nejdůležitější roli v oboru prostorové statistiky. V současné době je jejich úloha stále významnější, neboť jsou novou dostupnou technologií získávání obrovského množství dat.

## 2.1 Prostorový bodový proces

Prostorový bodový proces  $X$  je náhodná spočetná podmnožina prostoru  $S \subseteq \mathbb{R}^d$ , kde  $\mathbb{R}^d$  je  $d$ -dimenzionální Euklidovský prostor. V praxi ovšem nesledujeme všechny body prostoru  $S$ , ale pouze jeho části. Jedná se tedy o body nacházející se v omezeném sledovaném okně  $W \subseteq S$ . Toto okno má většinou tvar pravoúhelníku nebo je tvořeno souborem pravoúhelníků.

Bodový proces definovaný na prostoru  $S$  bere své hodnoty v  $N$ , což je prostor lokálně konečných bodových konfigurací v  $S$ . Prvky z prostoru  $N$  značíme  $x, y, \dots$ , zatímco body z  $S$  označujeme  $\xi, \eta, \dots$

Prostor  $S$  vybavíme borelovskou  $\sigma$ -algebrou označenou  $\mathcal{B}$  (jedná se o  $\sigma$ -algebru vygenerovanou otevřenými intervaly) a  $\mathcal{B}_0$  označíme třídu omezených borelovských množin. Prostor  $N$  vybavíme  $\sigma$ -algebrou  $\mathcal{N} = \sigma(\{x \in N : n(x_B) = m\} : B \in \mathcal{B}_0, m \in \mathbb{N}_0)$ . Pro zjednodušení si můžeme představit  $\mathcal{B}$  a  $\mathcal{N}$  jako velké rodiny podmnožin  $S$  a  $N$ .

Bodový proces  $X$  definovaný na  $S$  je tedy měřitelné zobrazení definované na nějakém pravděpodobnostním prostoru  $(\Omega, \Phi, \Pi)$  a beroucí hodnoty z  $(\mathcal{N}, \mathcal{N})$ .

## 2.2 Základní operace pro bodové procesy

Máme dvě základní operace pro bodové procesy. První z nich je skládání.

Skládáním nazveme disjunktní soubor  $\bigcup_{i=1}^{\infty} X_i$  bodových procesů  $X_1, X_2, \dots$

Druhou základní operací je ředění (thinning), které definujeme následujícím způsobem. Necht'  $p : S \rightarrow [0,1]$  je funkce a  $X$  je bodový proces na  $S$ . Bodový proces  $X_{thin} \subseteq X$  získaný vložením  $\xi \in X$  do  $X_{thin}$  s pravděpodobností  $p(\xi)$ , kde jsou body vkládány nebo odebírány nezávisle na sobě, bude nazýván nezávislé ředění  $X$  se stálou pravděpodobností  $p(\xi)$ ,  $\xi \in S$ . Formálně můžeme zapsat množinu  $X_{thin} = \{\xi \in X : R(\xi) \leq p(\xi)\}$ , kde  $R(\xi) \sim \text{Uniform}[0,1]$ ,  $\xi \in S$  jsou navzájem nezávislé a nezávislé na  $X$ .

## 2.3 Poissonův bodový proces

Poissonův bodový proces je nejjednodušší bodový proces, a proto nám slouží jako výchozí pro konstrukci procesů složitějších. Jedná se o model úplné prostorové nezávislosti bodů v rovině.

Poissonův bodový proces definovaný na prostoru  $S \subseteq \mathbb{R}^d$  je určen hustotou funkce  $\rho : S \rightarrow [0, \infty)$ , která je lokálně integrovatelná (tj.  $\int_B \rho(\xi) d\xi < \infty$ ) pro všechny omezené  $B \subseteq S$ . Je-li  $\rho$  konstantní, mluvíme o homogenním Poissonově bodovém procesu, jinak se jedná o proces nehomogenní. Homogenní proces je stacionární (tzn., že je jeho rozdělení neměnné po translaci) a izotropní (tj. neměnné rozdělení po rotaci). Nehomogenní proces není stacionární, ale může být izotropní.

Další veličinou určující Poissonův proces je míra intenzity  $\mu$  (daná jako  $\mu(B) = \int_B \rho(\xi) d\xi$  pro omezené  $B \subseteq S$ ), která nám určuje střední hodnotu bodů v  $B$  tj.

$$\mathbb{E}N(B) = \mu(B).$$

Bodový proces  $X$  definovaný na  $S$  nazveme Poissonův bodový proces s hustotou funkce  $\rho$  a hustotou míry  $\mu$ , splňuje-li následující vlastnosti:

- 1) Má Poissonovo rozdělení se střední hodnotou  $\mu(B)$  pro nějaké  $B \subseteq S$  s  $\mu(B) < \infty, N(B) \sim \text{po}(\mu(B))$ .
- 2)  $\forall B, C : B \cap C = \emptyset : N(B)$  a  $N(C)$  jsou nezávislé.

## 2.4 Cluster (shlukové) procesy

Jedná se o složitější procesy, kde body tvoří tzv. shluky (clustery). Vytvoření cluster procesu má dvě fáze:

- 1) Vytvoření mateřského (rodičovského) procesu, který odpovídá Poissonovu procesu a představuje jádra (zárodky) shluků.
- 2) Pro každý bod mateřského procesu generujeme dceřinný (dětský) proces.

Mezi nejznámější příklady cluster procesů patří Neyman-Scott procesy, které jsou v prostorové statistice využívány velmi často. Rodičovské bodové formy jsou tvořeny stacionárním Poissonovým procesem s intenzitou  $\mu$ , zatímco dětské body reprezentačního clusteru  $N_\theta$  jsou náhodné body, jejich rozptýlení je nezávislé a se stejnou distribucí kolem jádra clusteru. Ve sledovaném bodovém vzorku se nesmí vyskytovat rodičovský bod a zahrnujeme tudíž jen body dětské. Díky těmto předpokladům je výsledný cluster proces stacionární.

Do skupiny Neyman-Scott procesů patří i dva procesy, kterými se budu ve své diplomové práci podrobněji zabývat, a proto si je v následujících podkapitolách více přiblížíme.

### 2.4.1 Matérn cluster proces

Matérn cluster proces patří mezi tzv. globulární poissonovské procesy, což znamená, že dcery jsou rozmístěny rovnoměrně náhodně v kouli.

Počet rodičovských bodů v reprezentativním clusteru  $N_\theta$  je tedy dán Poissonovým rozdělením s parametrem  $\mu > 0$ . Body z  $N_\theta$  jsou nezávisle rovnoměrně rozptýleny v kouli  $b(\theta, R)$ , kde  $R$  je poloměr koule.

### 2.4.2 Thomas cluster proces

Thomas cluster proces se od předchozího liší ve způsobu vytvoření dceřiných bodů. Vzdálenost bodu od středu clusteru je dána normálním rozdělení a popisuje ji následující vztah  $v \sim N(0, \sigma^2)$ . Směr bodu od středu je učen rovnoměrným rozdělením.

### 3. Charakteristiky bodových procesů

V této kapitole definuji nejčastěji užívané souhrnné statistiky pro bodový proces pomocí kterých lze analyzovat jednotlivé typy bodových procesů. Prvním krokem při určování bodových procesů bývá hledání odlišností od Poissonova modelu, který můžeme nazvat základním srovnávacím procesem.

#### 3.1 Základní vlastnosti a vlastnosti druhého řádu

V této sekci budeme vždy uvažovat bodový proces  $X$  definovaný na  $S = \mathbb{R}^d$ . Základní vlastnosti a vlastnosti druhého řádu jsou popsány intenzitou míry a momentovou mírou druhého řádu.

Můžeme je definovat pomocí střední hodnoty  $\mathbb{E}$  následujícím způsobem. Intenzita míry  $\mu$  na  $\mathbb{R}^d$  je určena jako  $\mu(B) = \mathbb{E}N(B)$ ,  $B \subseteq \mathbb{R}^d$ . Momentová míra druhého řádu  $\alpha^{(2)}$  na  $\mathbb{R}^d \times \mathbb{R}^d$  je dána jako  $\alpha^{(2)}(C) = \mathbb{E} \sum_{\xi, \eta \in X}^{\neq} \mathbf{1}[(\xi, \eta) \in C]$ ,  $C \in \mathbb{R}^d \times \mathbb{R}^d$ , kde  $\mathbf{1}[\cdot]$  značí indikátorovou funkci.

Jinou možností je vymezení pomocí hustoty  $\rho$ , která je nezápornou funkcí. Potom získáme pro intenzitu míry vztah  $\mu(B) = \int_B \rho(\xi) d\xi$ ,  $B \subseteq \mathbb{R}^d$ . Současně platí, že  $\rho(\xi) d\xi$  je pravděpodobnost výskytu bodů v nekonečně malé kouli se středem  $\xi$  a objemem  $d\xi$ . Momentovou mírou druhého řádu můžeme tedy posléze zapsat jako  $\alpha^{(2)}(C) = \iint \mathbf{1}[(\xi, \eta) \in C] \rho^{(2)}(\xi, \eta) d\xi d\eta$ ,  $C \in \mathbb{R}^d \times \mathbb{R}^d$ , kde  $\rho^{(2)}$  je nezáporná funkce a nazýváme jí součinnou hustotu druhého řádu. Analogicky jako u intenzity míry i zde nám  $\rho^{(2)}(\xi, \eta) d\xi d\eta$  udává pravděpodobnost páru bodů z  $X$  vyskytující se v každé ze dvou nekonečně malých koulí se středy  $\xi, \eta$  a objemy  $d\xi, d\eta$ .

##### 3.1.1 g-funkce

Funkce  $g$ , kterou nazýváme párová korelace, se využívá nejvíce v astronomii a astrofyzice. Existují-li  $\rho$  a  $\rho^{(2)}$ , je funkce párová korelace definovaná jako

$g(\xi, \eta) = \frac{\rho^{(2)}(\xi, \eta)}{\rho(\xi)\rho(\eta)}$ , kde bereme  $\frac{a}{0} = 0$  pro  $a \geq 0$ . Pro Poissonův proces je  $g(\xi, \eta) = 1$ . Když je např.  $g(\xi, \eta) > 1$ , ukazuje to, že se bude dvojice bodů v lokalitách  $\xi, \eta$  vyskytovat pospolu pravděpodobněji než v Poissonově procesu se stejnou hustotou funkce jako  $X$ .

### 3.1.2 Redukovaná momentová míra druhého řádu

Předpokládejme, že  $X$  má hustotu (intenzitu) funkce  $\rho$  a že míra  $\kappa(B) = \frac{1}{|A|} E \sum_{\xi, \eta \in X} \frac{1[\xi \in A, \eta - \xi \in B]}{\rho(\xi)\rho(\eta)}$ ,  $B \subseteq \mathbb{R}^d$  je nezávislá na volbě  $A \subseteq \mathbb{R}^d$  s  $0 < |A| < \infty$ ,

kde bereme  $\frac{a}{0} = 0$  pro  $a \geq 0$ . Potom se  $X$  nazývá second order intensity reweighted stationary (pojem se do českého jazyka nepřekládá; lze chápat jako intenzita druhého řádu převážená na stacionaritu) a  $\kappa$  se nazývá redukovaná momentová míra druhého řádu (second order reduced moment measure).

Míra  $\kappa$  se využívá při konstrukci souhrnných statistik. Důležitá vlastnost, která pro ni platí (a tím potažmo i pro související souhrnné statistiky) je, že  $\kappa$  je neměnná při nezávislém ředění.

## 3.2 Souhrnné statistiky

V této části se zaměřím na charakteristiku různých souhrnných statistik pro bodový proces  $X$  na  $\mathbb{R}^d$ .

### 3.2.1 Souhrnné statistiky druhého řádu (K a L funkce)

$K$  a  $L$  funkce pro second order reweighted stationary point processes (bodové procesy druhého řádu převážené na stacionaritu) jsou definovány následovně:

$$K(r) = \kappa(b(0, r))$$

$$L(r) = (K(r) / \omega_d)^{1/d}, r > 0.$$

$K$  a  $L$  funkce spolu vzájemně korespondují a  $L$  funkce bývá často používána místo  $K$  funkce. Jedním z důvodů je, že  $L$  funkce je pro Poissonův proces blízká identitě  $L(r) = r$ . Obvykle (pro malé hodnoty  $r$ )  $L(r) - r > 0$  ukazuje nahromadění nebo shluknutí

na vzdálenosti menší než  $r$  a naopak  $L(r) - r < 0$  signalizuje rovnoměrnost na vzdálenosti menší než  $r$ .

Úzký vztah je i mezi funkcemi  $K$  a  $g$ . Funkci  $g$  nahrazujeme v odhadech často funkcí  $K$  nebo  $L$ , protože je jejich odhad snazší. Naopak pokud chceme sestrojít graf funkce  $K$ , která je rostoucí, nahradíme jej raději grafem funkce  $g$ , jenž lze sestrojít snáze. Podobně jako u  $L$  funkce, značí i zde  $g(r) > 1$  nahromadění nebo shluknutí na vzdálenosti  $r$  a  $g(r) < 1$  rovnoměrnost na téže vzdálenosti.

### 3.2.2 Souhrnné statistiky založené na vzdálenosti mezi body (F, G, J funkce)

Předpokládejme, že  $X$  je stacionární. The empty space function  $F$  je rozdělení funkce na vzdálenosti z počátku (nebo jiného pevného bodu  $\mathbb{R}^d$ ) k nejbližšímu bodu v  $X$ , tj.  $F(r) = P(X \cap b(0, r) \neq \emptyset)$ ,  $r > 0$ .

The nearest-neighbour function  $G$  studuje vzdálenosti mezi nejbližšími body a je definovaná jako  $G(r) = \frac{1}{\rho|A|} \mathbb{E} \sum_{\xi \in X \cap A} \mathbf{1}[(X \setminus \xi) \cap b(\xi, r) \neq \emptyset]$ , kde  $r > 0$ ,  $A \subset \mathbb{R}^d$  je libovolná množina, pro kterou platí  $0 < |A| < \infty$ .

Funkce  $J$  je definovaná jako  $J(r) = \frac{1 - G(r)}{1 - F(r)}$  pro  $F(r) < 1$ .

Pro stacionární Poissonův proces definovaný na  $\mathbb{R}^d$  s hustotou  $\rho < \infty$  platí, že  $F(r) = G(r) = 1 - \exp(-\rho \omega_d r^d)$  a  $J(r) = 1$  pro  $r > 0$ . Rozšíření funkcí  $F$ ,  $G$  a  $J$  pro nestacionární případy zatím není známé stejně jako pro jiné typy modelů. Obvykle (pro malé hodnoty  $r > 0$ ) ukazuje  $F(r) < G(r)$  (nebo  $J(r) < 1$ ) nahromadění nebo shluknutí a  $F(r) > G(r)$  (nebo  $J(r) > 1$ ) rovnoměrnost.

## 3.3 Neparametrické odhady

V této podkapitole bude  $X$  bodový proces na  $\mathbb{R}^d$  s intenzitou funkce  $\rho$ . Bude-li  $X$  stacionární, budeme považovat  $\rho$  za konstantu. Omezíme se na případy, kde jsou body modelu  $X_w = x$  pozorovány v omezeném okně  $W \subset \mathbb{R}^d$  s  $|W| > 0$ .

### 3.3.1 Neparametrické odhady hustoty funkce

V homogenním případě je nezkreslený odhad hustoty (intenzity)  $\hat{\rho} = \frac{n(x)}{|W|}$ , což je v podstatě maximální pravděpodobnost odhadu, je-li  $X$  homogenní Poissonův proces.

V nehomogenním případě se využívá jádrového neparametrického odhadu hustoty funkce definovaného jako  $\hat{\rho}_b(\xi) = \sum_{\eta \in x} \frac{k_b(\xi - \eta)}{c_{W,b}(\eta)}$ ,  $\xi \in W$ , kde  $k_b$  je jádrová

funkce s band width (šířka jádra)  $b > 0$  (tj.  $k_b(\xi) = \frac{k(\xi/b)}{b^d}$ , kde  $k$  je dáno hustotou funkce) a  $c_{W,b}(\eta) = \int_W k_b(\xi - \eta) d\xi$  je edge correction factor (korekce na okrajové jevy).

Odhad v nehomogenním případě bývá citlivý na výběr  $b$ , zatímco výběr  $k$  je méně důležitý.

Pro naše potřeby (kdy  $d = 2$ ) se s oblibou používá  $k(\xi) = e(\xi_1)e(\xi_2)$  pro  $\xi = (\xi_1, \xi_2) \in \mathbb{R}^2$ , kde  $e(u) = \frac{3}{4}(1 - |u|)\mathbf{1}[|u| \leq 1]$ ,  $u \in \mathbb{R}$  se nazývá Epanečnikovo jádro.

Nezkreslený odhad intenzity míry  $\mu$  je dán jako  $\mu(W) = \int_W \hat{\rho}_b(\xi) d\xi$ .

### 3.3.2 Neparametrické odhady K a L

Odhady funkcí  $K$  a  $L$  budeme provádět pomocí odhadu redukované míry druhého momentu.

Předpokládáme, že  $X$  je second order intensity reweighted stationary a že  $|W \cap W_\xi| > 0$  pro všechny  $\xi \in B$  a pro  $W_\xi = \{\eta + \xi : \eta \in W\}$  značící translaci  $W$  pomocí

$\xi \in \mathbb{R}^d$ . Nezkreslený odhad  $\kappa(B)$  potom získáme jako  $\sum_{\xi, \eta \in x} \frac{\mathbf{1}[\eta - \xi \in B]}{\rho(\xi)\rho(\eta)|W \cap W_{\eta - \xi}|}$ . Tento

nezkreslený odhad lze ovšem využít pouze v případě, kdy známe  $\rho$ , což v praxi nebývá. Proto je nutné ve vzorci nahradit  $\rho(\xi)\rho(\eta)$  jeho odhadem, čímž dostaneme kombinovaný odhad  $\hat{\kappa}(B)$ , který je ale již zkreslený.



### 3.3.3 Neparametrické odhady funkce g

Pro odhad párové korelace předpokládáme, že  $g(\xi, \eta) = g(\|\xi - \eta\|)$  je izotropní. Odhad g funkce můžeme potom vyjádřit pomocí následujícího vzorce jako

$$\hat{g}(r) = \frac{1}{\sigma_d r^{d-1} |W|} \sum_{\xi, \eta \in X} \frac{k_b(r - \|\eta - \xi\|)}{\rho(\xi)\rho(\eta) |W \cap W_{\eta-\xi}|}, \text{ kde } k_b(u) = k(u/b)/b, u \in \mathbb{R} \text{ pro jádro } k(\cdot) \text{ a}$$

band width  $b > 0$  a předpokládáme, že  $|W \cap W_\xi| > 0$  pro  $\|\xi\| \leq r$ . Pro dostatečně malé  $b$  a když je odhad  $\rho(\xi)\rho(\eta)$  blízký  $\rho(\xi)\rho(\eta)$ , tak očekáváme i  $\hat{g}(r)$  blízké  $g(r)$ . Odhad je citlivý na výběr  $b$  a je více zkreslený pro malé vzdálenosti  $r$ .

### 3.3.4 Neparametrické odhady funkcí F, G a J

Existuje více typů neparametrických odhadů funkcí  $F$  a  $G$ . Já ve své práci budu používat odhady pomocí tzv. minusových souborů.

Nechť  $d(\xi, B) = \inf\{\|\xi - \eta\| : \eta \in B\}$  je nejkratší vzdálenost z bodu  $\xi \in \mathbb{R}^d$  k množině  $B \subset \mathbb{R}^d$ . Nechť  $I \subset W$  značí konečnou pravidelnou síť bodů, které jsou vybrány nezávisle na  $X$  a necht'  $\#I_r$  značí mohutnost množiny  $I_r = I \cap W_{-r}$  pro  $r > 0$ .

Pak máme nezávislý odhad  $\hat{F}(r) = \sum_{\xi \in I_r} \mathbf{1}[d(\xi, x) \leq r] / \#I_r$  pro  $\#I_r > 0$ .

Reálně nezávislý odhad  $G$  funkce je následující:

$$\hat{G}(r) = \sum_{\xi \in X \cap W_{-r}} \mathbf{1}[d(\xi, x_\xi) \leq r] / (\hat{\rho} |W_{-r}|) \text{ pro } |W_{-r}| > 0.$$

Dalším způsobem odhadu  $F$  a  $G$  funkce je tzv. Kaplan – Meierův odhad. Někteří autoři jej používají raději, neboť se jim jeví jako účelnější. Proto i já uvedu a osvětlím vzorce tohoto druhu odhadů.

Nechť  $I \subset W$  značí konečnou pravidelnou síť bodů (vybraných nezávisle na  $X$ ) a necht'  $\#$  značí mohutnost množiny. Potom je odhad funkce  $F$  zapsán jako

$$\hat{F}(r) = 1 - \prod_{s \leq r} \left( 1 - \frac{\#\{\xi \in I : d(\xi) = s, d(\xi) \leq c(\xi)\}}{\#\{\xi \in I : d(\xi) \geq s, c(\xi) \geq s\}} \right), \text{ kde } c(\xi) = d(\xi, \partial W) \text{ je vzdálenost}$$

$\xi$  od hranice okna.

Odhad funkce  $G$ , kde  $c(\xi) = d(\xi, \partial W)$  je vzdálenost  $\xi$  od hranice okna a  $e(\xi) = d(\xi, X \setminus \xi)$  je vzdálenost  $\xi$  k nejbližšímu sousedovi, definujeme jako

$$\hat{G}(r) = 1 - \prod_{s \leq r} \left( 1 - \frac{\#\{\xi \in x : e(\xi) = s, e(\xi) \leq c(\xi)\}}{\#\{\xi \in x : e(\xi) \geq s, c(\xi) \geq s\}} \right) \text{ pro } r > 0.$$

Pomocí odhadů  $\hat{F}(r)$  a  $\hat{G}(r)$  získáme odhad  $\hat{J}(r) = \frac{1 - \hat{G}(r)}{1 - \hat{F}(r)}$  pro  $\hat{F}(r) < 1$ .

### 3.3.5 Obálky pro souhrnné statistiky

Graf neparametrických odhadů souhrnných statistik může být znázorněn jistým intervalem pro každou hodnotu  $r$ . Předpokládáme jednoduchou hypotézu  $H$ . Jisté intervaly a distribuční charakteristiky asociované s neparametrickým odhadem dané souhrnné statistiky můžeme získat užitím simulace pod  $H$ . Pro danou vzdálenost  $r > 0$  necháme  $T_0(r) = T(X, r)$ , což značí neparametrický odhad získaný z bodového procesu  $X$  sledovaného uvnitř okna  $W$ . Necht'  $T_1(r) = T(X_1, r)$ , ...,  $T_n(r) = T(X_n, r)$  jsou získány z i.i.d. simulací  $X_1$ , ...,  $X_n$  pod  $H$ . Z empirické distribuce  $T_1(r)$ , ...,  $T_n(r)$  můžeme odhadnout některé kvantily pro rozdělení  $T_0(r)$  pod  $H$  a lze to provést s nějakou požadovanou přesností, je-li  $n$  dostatečně velké. Při porovnávání výsledků pro různé hodnoty  $r$ , musíme být opatrní, neboť  $T_1(r)$ , ...,  $T_n(r)$  jsou sice i.i.d., ale náhodný vektor  $(T_1(r), \dots, T_n(r))$  považujeme pro různé hodnoty  $r > 0$  za závislý.

## 4. Simulace

V této kapitole jsou uvedeny a vysvětleny zdrojové kódy simulací bodových procesů (Bernoulliho, Poissonův, Matérn cluster, Thomas cluster včetně několika typů jim příslušných nehomogenních případů) a jejich charakteristik ( $K$ ,  $L$ ,  $F$ ,  $G$  a  $J$  funkce) pro program Mathematica verze 6.

### 4.1 Bernoulliho proces

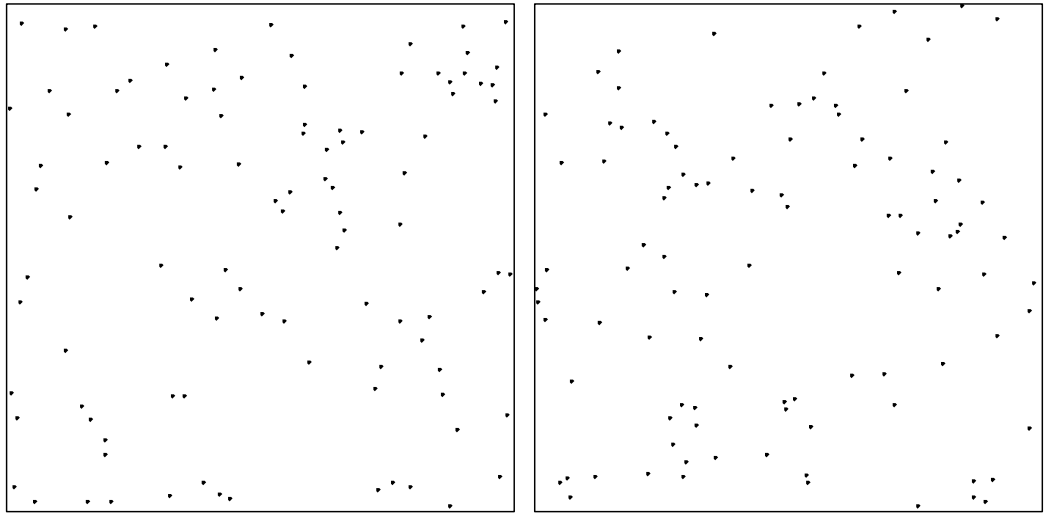
Simulace Bernoulliho procesu se jeví jako nejjednodušší. Jedná se funkci zadanou pomocí tří parametrů. Prvním z nich je počet bodů (ten je v tomto procesu fixní), další dva udávají rozměry sledovaného okna. Pozice jednotlivých bodů je dána rovnoměrným rozdělením.

Tento proces se podobá homogennímu případu Poissonova procesu. Rozdíl je pouze v tom, že zde je dán počet bodů pevně, zatímco u Poissona se bude generovat náhodně.

```
BernoulliProcess = Function[{NumbPoint, A, B},  
  Table[{Random[UniformDistribution[{0, A}]],  
    Random[UniformDistribution[{0, B}]]}, {NumbPoint}];
```

Abychom získali grafický výstup je třeba zadat programu následující kód (tento budeme užívat i pro všechny následující simulace):

```
Vystup[co_, A_, B_] := Module[{},  
  okno = Graphics[ListLinePlot[{{0, 0}, {A, 0}, {A, B}, {0, B}, {0, 0}},  
    Axes -> False, AspectRatio -> 1];  
  Show[okno, ListPlot[co, Axes -> False, AspectRatio -> 1]]];
```



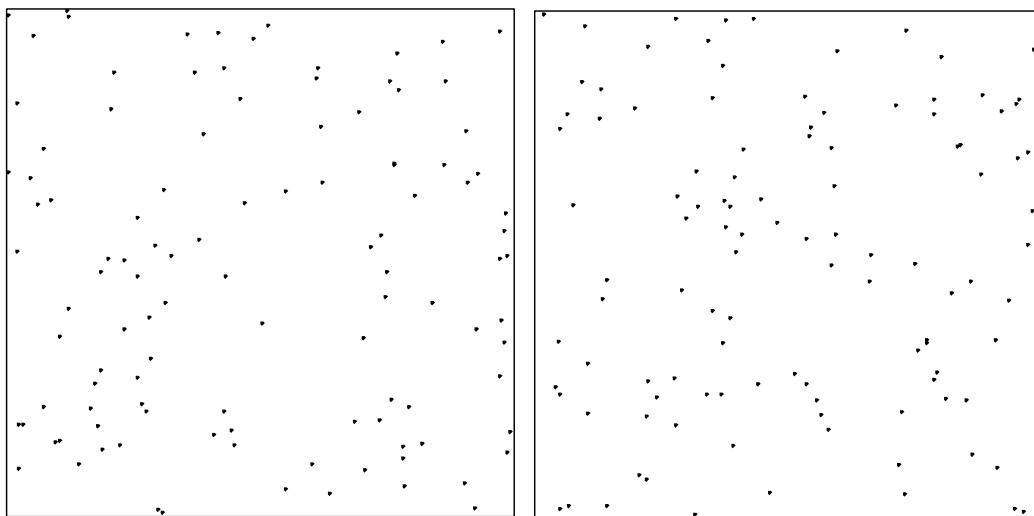
Obr. 4.1: Příklad dvou simulací Bernoulliho procesu pro 100 bodů a rozměry sledovaného okna  $A = 1$ ,  $B = 1$ .

## 4.2 Homogenní Poissonův proces

Homogenní Poissonův proces zavedeme jako funkci tří proměnných:  $\rho$  (průměrný počet bodů v jednotce plochy),  $A$  a  $B$  (rozměry sledovaného okna). V homogenním případě je  $\rho$  konstanta.

Počet bodů je náhodné číslo určené Poissonovým rozdělením s mírou získanou jako součin průměrného počtu bodů v jednotce plochy a celkové plochy okna. Rozmístění jednotlivých bodů získáme, stejně jako u Bernoulliho procesu, pomocí rovnoměrného rozdělení. Pro osu  $x$  bereme hodnoty z intervalu  $\langle 0, A \rangle$ , pro osu  $y$  potom z intervalu  $\langle 0, B \rangle$ . Je tedy zřejmé, že libovolný bod má stejnou pravděpodobnost výskytu v libovolném místě sledovaného okna a jeho umístění nezávisí na poloze jiných bodů.

```
PoissonProcess=Function[{Rho, A, B},
Module[{NumbPoint},
NumbPoint=Random[PoissonDistribution[Rho*A*B]];
Table[{Random[UniformDistribution[{0,A}],Random[UniformDistribution[{0,B}]]],
{NumbPoint}}];
```



Obr. 4.2: Příklad dvou simulací homogenního Poissonova procesu pro hodnoty  $\rho = 100$ ,  $A = 1$ ,  $B = 1$ .

### 4.3 Nehomogenní Poissonův proces

Nehomogenní Poissonův proces budeme simulovat s využitím operace nezávislého ředění následujícím způsobem.

Předpokládáme, že  $X \sim \text{Poisson}(S, \rho)$  podléhá nezávislému ředění se stálou pravděpodobností  $p(\xi)$ ,  $\xi \in S$  a necht'  $\rho_{thin}(\xi) = p(\xi)\rho(\xi)$ ,  $\xi \in S$ . Potom  $X_{thin}$  a  $X \setminus X_{thin}$  jsou nezávislé Poissonovy procesy s příslušnými intenzitami  $\rho_{thin}$  a  $\rho - \rho_{thin}$ .

Z tohoto tvrzení, lze vyvodit, že nehomogenní Poissonův proces může být odvozený ze zředěného procesu homogenního. Platí totiž, je-li  $X \sim \text{Poisson}(\mathbb{R}^d, \rho)$ , kde je  $\rho$  omezená konečnou konstantou  $c$ , je potom  $X$  rozložený jako nezávislé ředění  $\text{Poisson}(\mathbb{R}^d, c)$  se stálou pravděpodobností  $p(\xi) = \rho(\xi) / c$ .

Při vytváření vlastního zdrojového kódu začneme zadáním funkce  $z$ , která nám udává nehomogenitu tohoto procesu. Její první parametr závisí na součtu, druhý na součinu souřadnic.

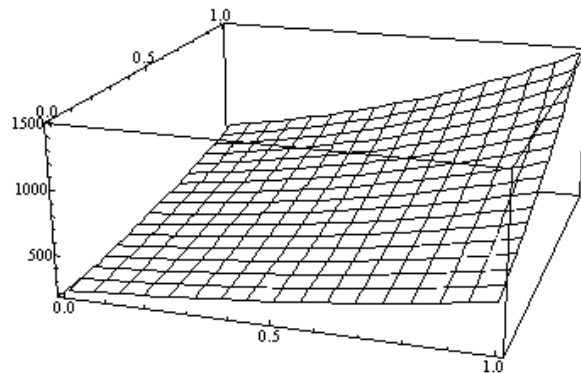
```
z = Function[{u}, {u[[1]] + u[[2]], u[[1]] * u[[2]]}];
```

Další v pořadí je zápis funkce intenzity  $\rho$

```
rho = Function[{alpha, beta, u}, alpha*Exp[z[u].beta]];
```

Pro lepší představu toho, jaká bude intenzita bodů v jednotlivých částech okna, je možné nechat vykreslit 3D graf.

```
Plot3D[rho[alpha, beta, {u, v}], {u, 0, 1}, {v, 0, 1}]
```



Obr. 4.3.1: 3D graf funkce intenzity pro hodnoty  $\alpha = 500$ ;  $\beta = \{1,0\}$ .

Posledním potřebným parametrem, před zadáním samotného procesu, je maximum, kterého dosahuje intenzita. Toto spočítá program ze vzorce

```
M = Maximize[rho[alpha, beta, {u1, u2}], 0 <= u1 <= A && 0 <= u2 <= B, {u1, u2}][[1]]
```

Nyní můžeme přistoupit k vytvoření zdrojového kódu pro Poissonův nehomogenní proces. Je opět zadán jako funkce. Proměnnými tentokrát jsou:  $M$  (maximum intenzity),  $f$  (funkce intenzity zadaná fixními parametry  $\alpha, \beta$  a parametrem nefixním, který zapíšeme jako #1),  $A$  a  $B$  (rozměry sledovaného okna).

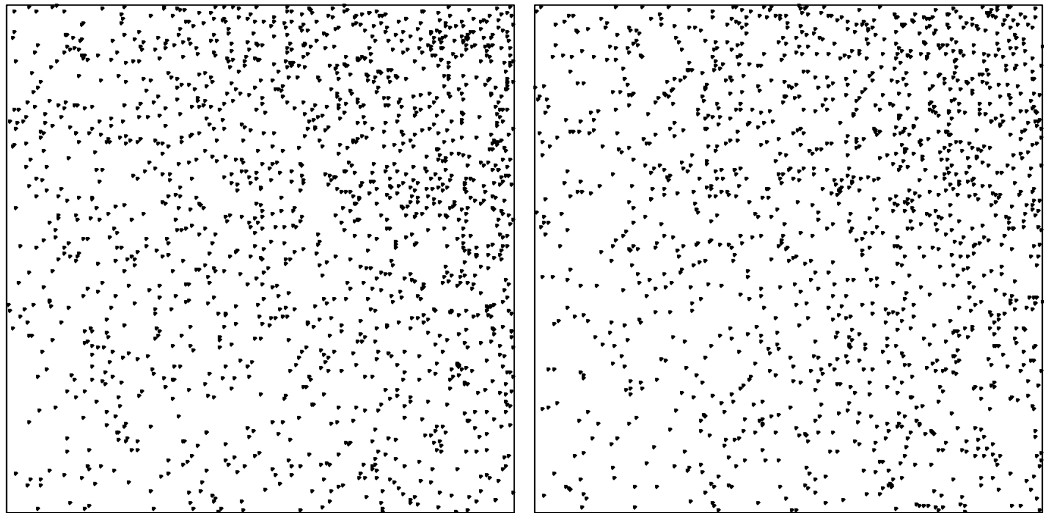
Počet bodů procesu i jejich rozmístění jsou dány v první fázi simulace stejným způsobem jako u procesu homogenního. Poté přistoupíme k ředění tohoto procesu tak, že vygenerujeme ke každému bodu procesu náhodné číslo ( $RN$ ) z intervalu  $\langle 0,1 \rangle$ . Pokud je toto náhodné číslo menší, než skutečná intenzita dělená maximální intenzitou, je i tento bod bodem procesu. V opačném případě bod vymažeme. Tímto způsobem nám vznikne nový, zředěný proces ( $PP$ ).

```
PoissonProcessInhom = Function[{M, f, A, B},
Module[{NumbPoint, CentersUp, RN, PP},
NumbPoint = Random[PoissonDistribution[M*A*B]];
CentersUp = Table[{RandomReal[{0, A}], RandomReal[{0, B}]}, {NumbPoint}];
RN = Table[RandomReal[{0, 1}], {NumbPoint}];
PP = {};
Do[If[RN[[i]] < f[CentersUp[[i]]]/M,
```

```

PP = Append[PP, CentersUp[[i]]]
, {i, 1, NumbPoint}];
PP];

```



Obr. 4.3.2: Simulace nehomogenního Poissonova procesu pro hodnoty parametrů  $\alpha = 500$ ;  $\beta = \{1,0\}$ ;  $A = 1$ ;  $B = 1$ . Rozmístění bodů ve sledovaném okně odpovídá grafu intenzity (větší množství bodů je soustředěno v pravém horním rohu).

#### 4.4 Matérn cluster proces

Matérn cluster proces je daný veličinami  $\rho$  (průměrný počet bodů v jednotce plochy),  $\mu$  (počet bodů v clusteru),  $R$  (poloměr clusteru),  $A$  a  $B$  (rozměry sledovaného okna). Průměrný počet clusterů je označen jako  $l$  a získáme ho ze vztahu  $l = \frac{\rho}{\mu}$ .

Nejdříve je nutné nasimulovat rodičovský proces. Počet rodičů (středů clusteru) je náhodný a je dán Poissonovým rozdělením – jedná se o homogenní Poissonův proces s mírou získanou jako součin průměrného počtu clusterů a plochy okna. Abychom neztratili potomky, kteří leží ve sledovaném okně daném rozměry  $A$  a  $B$ , ale jejichž rodiče se nacházejí mimo tuto oblast, je třeba simulovat rodiče pro větší okno (o rozměrech  $A+2R$ ,  $B+2R$ ). Počet dcer (bodů v každém clusteru; Dougt), které vyprodukuje každý rodič, je opět náhodné číslo určené Poissonovým rozdělením s mírou  $\mu$ .

Nyní je třeba zadat, které dcery budeme řadit do clusteru a které nikoliv. Podmínkou je, aby ležely v určité vzdálenosti od rodiče. Zadáme tedy vzdálenost dcery od rodiče  $r$  jako náhodné číslo s rovnoměrným rozdělením s intervalem  $\langle 0, R \rangle$  a úhel  $\varphi$ , který svírá dcera, rodič a kolmice na příslušnou stranu sledovaného okna. Tento úhel je také náhodné číslo dané rovnoměrným rozdělením s intervalem  $\langle 0, 2\pi \rangle$ . Každá dcera má tedy souřadnice  $\{r \cos(\varphi), r \sin(\varphi)\}$ . Je-li vzdálenost mezi souřadnicemi dcery a rodiče menší nebo rovna poloměru clusteru, je daná dcera dcerou clusteru (Dougters).

```

MaternClusterProcess = Function[{rho, mu, R, A, B},
Module[{l, r, fi},
l = rho/mu;

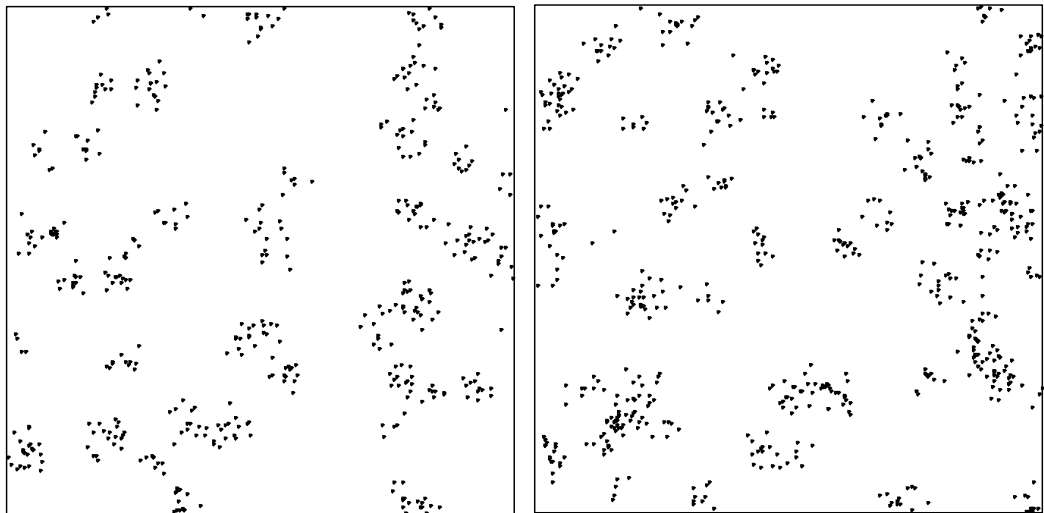
NumParent =
Random[PoissonDistribution[l*(A*B + 2*(A + B)*R + 4*R^2)]];
Parents =
Table[{Random[UniformDistribution[{-R, A + R}]],
Random[UniformDistribution[{-R, B + R}]]}, {NumParent}];

Dougters = {};
Do[NumDoug = Random[PoissonDistribution[mu]];
Doug = {};
For[j = 1, j <= NumDoug, j++,
r = Random[UniformDistribution[{0, R}]];
fi = Random[UniformDistribution[{0, 2*Pi}]];
souradnice = {Parents[[i, 1]] + r*Cos[fi], Parents[[i, 2]] + r*Sin[fi]};
Doug = Join[Doug, {souradnice}];
];
Dougters = Join[Dougters, Doug];
, {i, 1, NumParent}

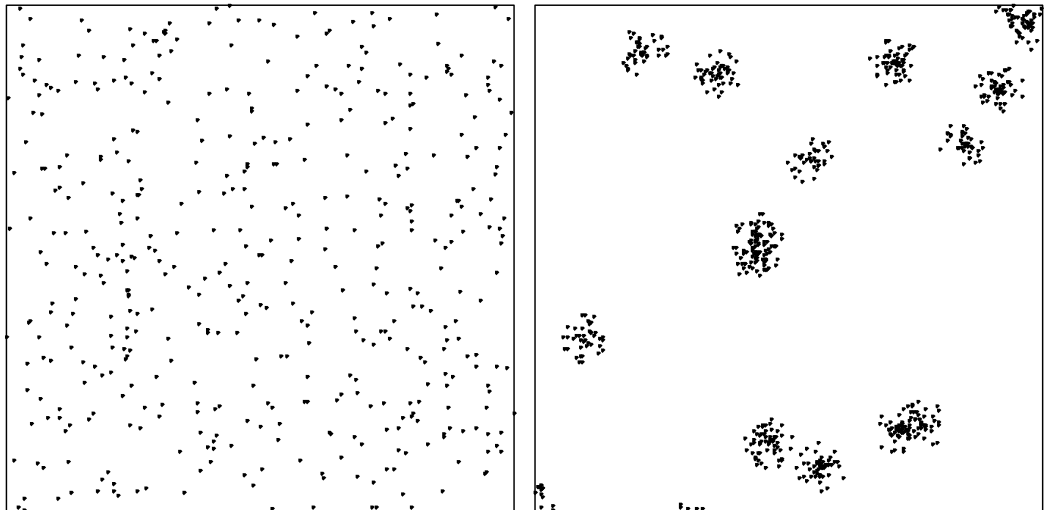
For[i = 1; P = {}, i <= Length[ Dougters], i++,
If[ Dougters[[i, 1]] >= 0 && Dougters[[i, 1]] <= A &&
Dougters[[i, 2]] >= 0 && Dougters[[i, 2]] <= B,
P = Append[P, Dougters[[i]]]];

```





Obr. 4.4.1: Příklady simulací Matérn cluster procesů pro hodnoty parametrů  $\rho = 500$ ;  $\mu = 10$ ;  $R = 0,05$ ;  $A = 1$ ;  $B = 1$ .



Obr. 4.4.2: Příklad simulace Matérn cluster procesu pro hodnoty  $\rho = 500$ ;  $\mu = 100$ ;  $R = 2$ ;  $A = 1$ ;  $B = 1$ . V tomto případě je velmi podobný Poissonovu procesu.

Obr. 4.4.3: Příklad simulace Matérn cluster procesu pro hodnoty  $\rho = 500$ ;  $\mu = 50$ ;  $R = 0,05$ ;  $A = 1$ ;  $B = 1$ . Při takto zadaných parametrech jsou shluky patrné na první pohled.

## 4.5 Matérn cluster nehomogenní proces

Matérn cluster nehomogenní proces je založen na stejném principu jako nehomogenní Poissonův proces. Začneme tedy zadáním funkce  $z$ , která nám udává nehomogenitu tohoto procesu.

```
z = Function[{u}, {u[[1]] + u[[2]], u[[1]] * u[[2]]};
```

Další v pořadí je opět zápis funkce intenzity  $\rho$

$\rho = \text{Function}[\{\alpha, \beta, u\}, \alpha \cdot \text{Exp}[z[u] \cdot \beta]];$

Graf funkce intenzity bude tedy stejný jako u nehomogenního Poissonova procesu, proto jej není nutné znovu zobrazovat.

Nesmíme zapomenout na výpočet maxima, kterého může intenzita dosáhnout.

K tomu použijeme znovu vzorce

$M = \text{Maximize}[\rho[\alpha, \beta, \{u1, u2\}], 0 \leq u1 \leq A \ \&\& \ 0 \leq u2 \leq B, \{u1, u2\}][[1]]$

Nehomogenní Matérn cluster proces je zadán pomocí následujících proměnných:

$M$  (maximum intenzity),  $f$  (funkce intenzity zadaná fixními parametry  $\alpha, \beta$  a parametrem nefixním, který zapíšeme jako #1),  $\mu$  (počet bodů v clusteru),  $R$  (poloměr clusteru),  $A$  a  $B$  (rozměry sledovaného okna). Průměrný počet clusterů je označen jako  $l$

a získáme ho ze vztahu  $l = \frac{M}{\mu}$ . Maximální rozsah clusteru ( $MR$ ) je u Matérn procesu

roven poloměru clusteru.

Prvním krokem je zjištění počtu bodů tvořících středy clusterů ( $PC$ ). Tato hodnota je dána jako náhodné číslo s Poissonovým rozdělením s mírou získanou jako součin průměrného počtu clusterů a plochy okna (opět zvětšené na rozměry  $A+2R$ ,  $B+2R$ ).

Počet dcer, které vyprodukuje každý rodič, je zase náhodné číslo určené Poissonovým rozdělením s mírou  $\mu$ . Vzdálenost dcery od rodiče  $r$  je určena jako náhodné číslo s rovnoměrným rozdělením s intervalem  $\langle -R, R \rangle$  a úhel  $\varphi$  je také náhodné číslo z intervalu  $\langle 0, \pi \rangle$ . Takto vygenerované body mají souřadnice  $\{r \cos(\varphi), r \sin(\varphi)\}$ . Je-li vzdálenost mezi souřadnicemi tohoto bodu a rodiče menší nebo rovna poloměru clusteru, je daný bod dcerou clusteru.

V této fázi simulace přistoupíme k samotnému ředění. Nejprve přiřadíme ke každému bodu procesu náhodné číslo ( $RN$ ) z intervalu  $\langle 0, 1 \rangle$ . Je-li toto náhodné číslo menší, než skutečná intenzita dělená maximální intenzitou středů, necháme tento bod bodem procesu. V opačném případě bod vymažeme. Tímto způsobem nám vznikne nový, zředěný proces ( $PPT_{thin}$ ).

```

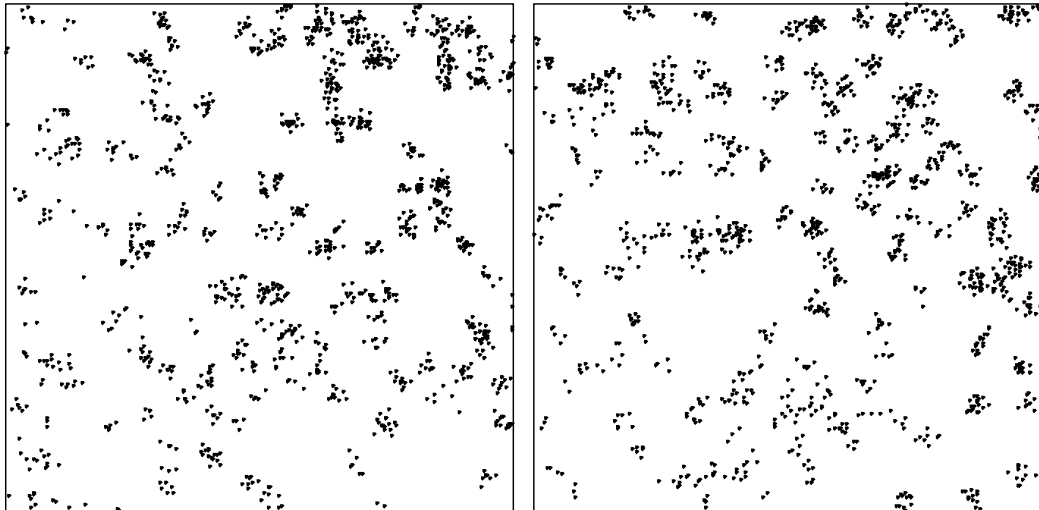
MaternProcessInhom=Function[{M, f, mu, R, A, B},
  Module[{l, NumbPoint, RN, Dougters, PC, NumbDougters, fi, r, Pointik, PPThin,
MR},
  l=M/mu;
  MR=R;

  NumbPoint=Random[PoissonDistribution[l*(A*B+2 (A+B)*MR+4*MR^2)]];
  PC=Table[{RandomReal[{-MR,A+MR}],RandomReal[{-MR,B+MR}]},{NumbPoint}];

  Dougters={};
  Do[NumbDougters=Random[PoissonDistribution[mu]];
  r=Table[Random[UniformDistribution[{-R,R}]],{NumbDougters}];
  fi=Table[RandomReal[{0,Pi}],{NumbDougters}];
  Do[
  Pointik={PC[[i,1]]+Cos[fi[[j]]]*r[[j]],PC[[i,2]]+Sin[fi[[j]]]*r[[j]];
  If[Pointik[[1]]>0&&Pointik[[2]]>0 && Pointik[[1]]<A && Pointik[[2]]<B,
  Dougters=Append[Dougters,Pointik]];
  ,{j,1,NumbDougters}];
  ,{i,1,Length[PC]};

  RN=Table[RandomReal[{0,1}],{Length[Dougters]}];
  PPThin={};
  Do[If[RN[[i]]<f[Dougters[[i]]]/M,PPThin=Append[PPThin,Dougters[[i]]]
  ,{i,1,Length[Dougters]}];
  PPThin]];

```



Obr. 4.5: Ukázka simulací Matérn cluster nehomogenního procesu pro hodnoty  $\alpha = 500$ ;  $\beta = \{1,0\}$ ;  $\mu = 20$ ;  $R = 0,025$ ;  $A = 1$ ;  $B = 1$ . Opět je patrné větší množství bodů v pravé horní části sledovaného okna.

## 4.6 Matérn cluster proces s nehomogenními centry

Rozdíl mezi předchozím nehomogenním Matérn cluster procesem a procesem následujícím spočívá v tom, že nyní se nehomogenita projeví hned při výběru rodičů (středů clusterů).

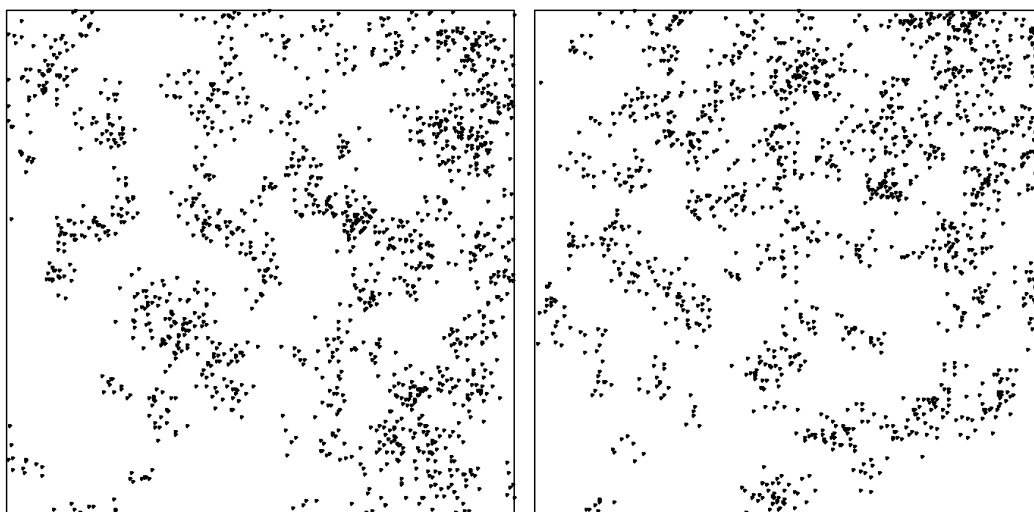
Zadávané proměnné i větší část simulace se shoduje s předešlou simulací. Nejprve určíme pomocí Poissonova rozdělení počet bodů procesu. V dalším kroku rozmístíme středy clusterů (*CentersUp*), každému přiřadíme náhodné číslo (*RN*) a opět budeme zjišťovat, zda splňuje toto číslo podmínku, že je menší než skutečná intenzita dělená maximální intenzitou středů. Není-li tato podmínka splněna, bod nebudeme dále uvažovat.

Počet a rozmístění dceřiných procesů simulujeme stejným způsobem jako u klasického Matérn cluster procesu.

```
MaternProcessInhomCenter = Function[{M, f, mu, R, A, B},
  Module[{NumbPoint, CentersUp, RN, Dougters, PC, NumbDougters, fi, r, Pointik,
    MR},
    MR = R;

    NumbPoint = Random[PoissonDistribution[M*(A*B + 2 (A + B)*MR + 4*MR^2)]];
    CentersUp = Table[{RandomReal[{-MR, A + MR}], RandomReal[{-MR, B + MR}]},
      {NumbPoint}];
    RN = Table[RandomReal[{0, 1}], {NumbPoint}];
    PC = {};
    Do[If[RN[[i]] < f[CentersUp[[i]]]/M,
      PC = Append[PC, CentersUp[[i]]]
    , {i, 1, NumbPoint}];

    Dougters = {};
    Do[NumbDougters = Random[PoissonDistribution[mu]];
      r = Table[Random[UniformDistribution[{-R, R}]], {NumbDougters}];
      fi = Table[RandomReal[{0, Pi}], {NumbDougters}];
      Do[
        Pointik = {PC[[i, 1]] + Cos[fi[[j]]]*r[[j]],
          PC[[i, 2]] + Sin[fi[[j]]]*r[[j]];
        If[Pointik[[1]] > 0 && Pointik[[2]] > 0 && Pointik[[1]] < A &&
          Pointik[[2]] < B,
          Dougters = Append[Dougters, Pointik];
        , {j, 1, NumbDougters}];
      , {i, 1, Length[PC]};
    Dougters];
```



Obr. 5.6: Ukázka simulací Matérn cluster procesu s nehomogenními centry pro hodnoty  $\alpha = 50$ ;  $\beta = \{1,0\}$ ;  $\mu = 10$ ;  $R = 0,05$ ;  $A = 1$ ;  $B = 1$ .

#### 4.7 Thomas cluster proces

Thomas cluster proces se v mnohém podobá Matérn cluster procesu. Hlavní rozdíl mezi nimi spočívá ve způsobu generování dceřinných procesů. I zdrojový kód bude tedy podobný.

Parametry zadávané u Thomas cluster procesu jsou:  $\rho$  (průměrný počet bodů v jednotce plochy),  $\mu$  (počet bodů v clusteru),  $\sigma$  (směrodatná odchylka),  $A$  a  $B$  (rozměry sledovaného okna). Průměrný počet clusterů je označen opět jako  $l$  a získáme ho z již známého vztahu  $l = \frac{\rho}{\mu}$ . Vzhledem k tomu, že normální rozdělení může dosahovat až do nekonečna, musíme zvolit maximální vzdálenost, kterou umožňujeme. Tuto značíme  $R$  a její hodnota je dána jako  $4 * \sigma$ , což by mělo obsáhnout asi 99,9% všech bodů.

Simulace rodičovského procesu probíhá stejně jako u Matérn cluster procesu. Změna nastane právě při generování dceřinných procesů. Počet dcer, které vyprodukuje každý rodič, je náhodné číslo určené Poisonovým rozdělením s mírou  $\mu$ . Jejich vzdálenost od rodiče  $r$  je dána normálním rozdělením s intervalem  $\langle 0, \sigma \rangle$ . Úhel  $\varphi$ , který svírá dcera, rodič a kolmice na příslušnou stranu sledovaného okna je taktéž

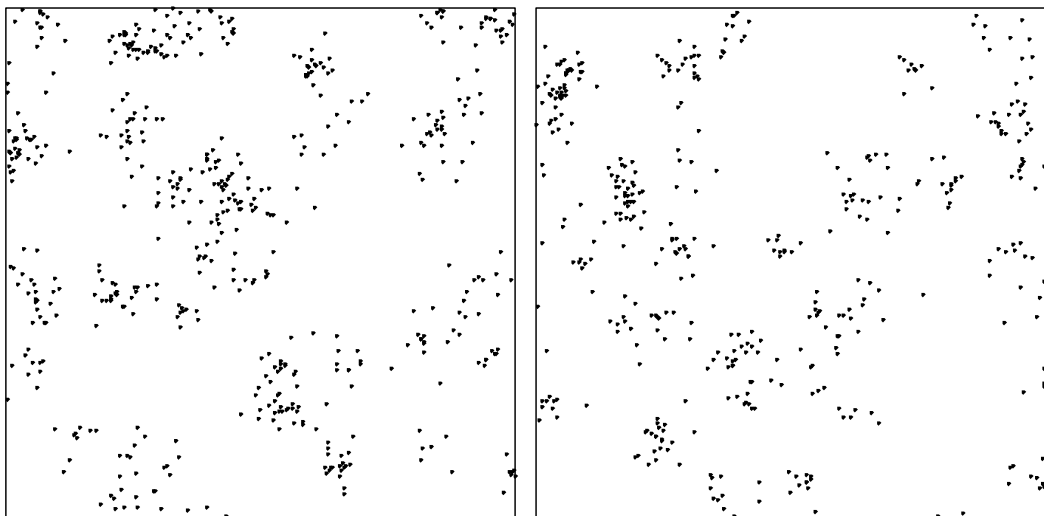
náhodné číslo ale dané rovnoměrným rozdělením s intervalem  $\langle 0, \pi \rangle$  (omezení úhlu do  $\pi$  je zapříčiněno tím, že  $r$  může nabývat záporných hodnot). Souřadnice každé dcery jsou tedy  $\{r \cos(\varphi), r \sin(\varphi)\}$ . Je-li vzdálenost mezi těmito souřadnicemi a souřadnicemi rodiče menší nebo rovna poloměru clusteru, je daná dcera dcerou clusteru.

```

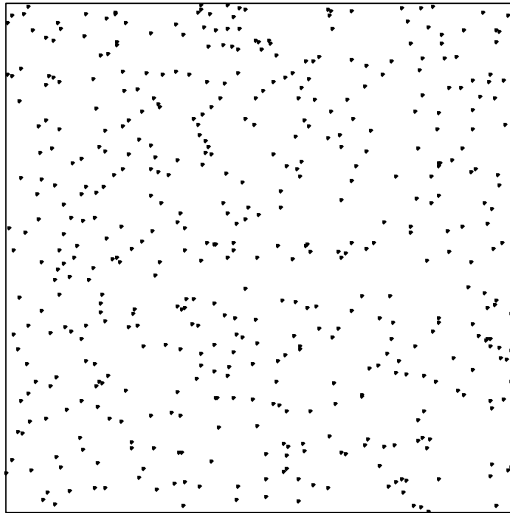
ThomasClusterProcess = Function[{rho, mu, sigma, A, B},
  Module[{l, r, fi, R},
    l = rho/mu;
    R = 4*sigma;
    NumParent = Random[PoissonDistribution[l*(A*B + 2*(A + B)*R + 4*R^2)]];
    Parents = Table[{Random[UniformDistribution[{-R, A + R}]},
      Random[UniformDistribution[{-R, B + R}]]}, {NumParent}];

    Dougters = {};
    Do[NumDought = Random[PoissonDistribution[mu]];
      Dougt = {};
      For[j = 1, j <= NumDought, j++,
        r = Random[NormalDistribution[0, sigma]];
        fi = Random[UniformDistribution[{0, Pi}]];
        souradnice = {Parents[[i, 1]] + r*Cos[fi], Parents[[i, 2]] + r*Sin[fi]};
        Dougt = Join[Dougt, {souradnice}];
      ];
      Dougters = Join[Dougters, Dougt];
    , {i, 1, NumParent}]
    For[i = 1; P = {}, i <= Length[Dougters], i++,
      If[Dougters[[i, 1]] >= 0 && Dougters[[i, 1]] <= A && Dougters[[i, 2]] >= 0 &&
        Dougters[[i, 2]] <= B, P = Append[P, Dougters[[i]]]];
    P];

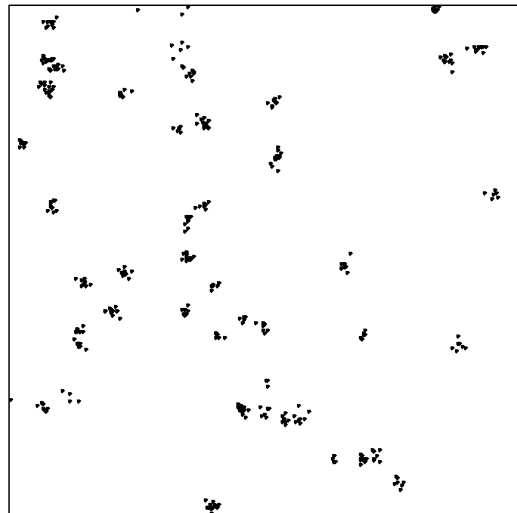
```



Obr. 4.7.1: Příklad simulací Thomas cluster procesu pro hodnoty parametrů:  $\rho = 500$ ;  $\mu = 10$ ;  $\sigma = 0,04$ ;  $A = 1$ ;  $B = 1$ .



Obr. 4.7.2: Simulace Thomas cluster procesu pro hodnoty parametrů:  $\rho = 500$ ;  $\mu = 10$ ;  $\sigma = 1$ ;  $A = 1$ ;  $B = 1$ . Při takovém zadání je problém rozeznat jej od Poissonova procesu.



Obr. 4.7.3: Simulace Thomas cluster procesu pro hodnoty parametrů:  $\rho = 500$ ;  $\mu = 0,01$ ;  $\sigma = 1$ ;  $A = 1$ ;  $B = 1$ . V případě tohoto zadání jsou sluky bodů nejvíce patrné.

## 4.8 Thomas cluster nehomogenní proces

Thomas cluster nehomogenní proces vychází ze stejného základu jako nehomogenní Matérn proces. Parametry jsou  $M$  (maximální intenzita středů);  $f$  (funkce intenzity zadaná fixními parametry  $\alpha, \beta$  a parametrem nefixním, který zapíšeme jako #1);  $\mu$  (počet dcer);  $\sigma$  (směrodatná odchylka);  $MR$  (maximální rozsah clusteru; v našem případě budeme uvažovat hodnotu  $4 * \sigma$ );  $A$  a  $B$  (rozměry sledovaného okna).

Stejně jako u Poissonova nehomogenního procesu musíme začít určením funkce nehomogenity.

```
z = Function[{u}, {u[[1]] + u[[2]], u[[1]]*u[[2]]}]
```

Následuje zadání funkce celkové intenzity.

```
rho = Function[{alpha, beta, u}, alpha*Exp[z[u].beta]]
```

A opět i výpočet maximální intenzity středů.

```
M = Maximize[rho[alpha, beta, {u1, u2}], 0 <= u1 <= A && 0 <= u2 <= B, {u1, u2}][[1]]
```

Intenzitu středů označíme  $l$  a spočítáme ze vztahu  $l = \frac{M}{\mu}$ . Počet bodů představující středy clusterů ( $PC$ ) vygenerujeme jako náhodné číslo s Poissonovým rozdělením a parametrem odpovídajícím intenzitě středů vynásobené velikostí okna, která bere ohled na maximální povolený rozsah clusteru.

Počet bodů dceřiných procesů ( $NumbDoughters$ ) získáme jako náhodné číslo dané Poissonovým rozdělením s parametrem  $\mu$ . Vzdálenosti mezi jednotlivými body jsou určeny normálním rozdělením s intervalem  $\langle 0, \sigma \rangle$ . Směr (úhel) bodu od centra clusteru je náhodné číslo z intervalu  $\langle 0, \pi \rangle$ . Souřadnice každé z dcer jsou tedy  $\{r \cos(\varphi), r \sin(\varphi)\}$ . V této fázi simulace započítáme všechny dceřinné body, které patří do sledovaného okna.

Nyní přistoupíme k samotnému zředění procesu a tím k získání procesu nehomogenního. Nejprve vygenerujeme ke každému bodu procesu náhodné číslo z intervalu  $\langle 0, 1 \rangle$ . Je-li toto náhodné číslo menší, než skutečná intenzita dělená maximální intenzitou středů, necháme tento bod bodem procesu. V opačném případě bod vymažeme. Tímto způsobem nám vznikne nový, zředěný proces ( $PPTthin$ ).

```

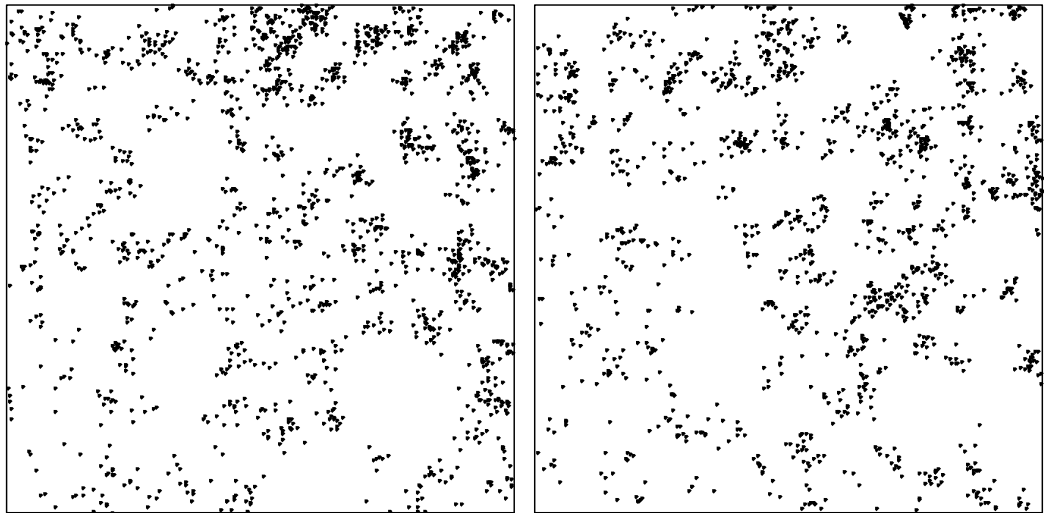
ThomasProcessInhom=Function[{M, f, mu, sigma, MR, A, B},
Module[{l, NumbPoint, RN, Doughters, PC, NumbDoughters, fi, r, Pointik, PPTthin},
  l=M/mu;
  NumbPoint=Random[PoissonDistribution[l*(A*B+2 (A+B)*MR+4*MR^2)]];
  PC=Table[{RandomReal[{-MR,A+MR}],RandomReal[{-MR,B+MR}]},{NumbPoint}];

  Doughters={};
  Do[NumbDoughters=Random[PoissonDistribution[mu]];
  r=Table[Random[NormalDistribution[0,sigma]},{NumbDoughters}];
  fi=Table[RandomReal[{0,Pi}},{NumbDoughters}];
  Do[Pointik={PC[[i,1]]+Cos[fi[[j]]]*r[[j]],PC[[i,2]]+Sin[fi[[j]]]*r[[j]];
  If[Pointik[[1]]>0&&Pointik[[2]]>0 && Pointik[[1]]<A &&
Pointik[[2]]<B,Doughters=Append[Doughters,Pointik]];
  ,{j,1,NumbDoughters}];
  ,{i,1,Length[PC]}}];

  RN=Table[RandomReal[{0,1}},{Length[Doughters}]];
  PPTthin={};
  Do[If[RN[[i]]<f[Doughters[[i]]]/M,PPTthin=Append[PPTthin,Doughters[[i]]];
  ,{i,1,Length[Doughters}]];
PPTthin]];

```





Obr. 4.8: Příklady simulací Thomas cluster nehomogenního procesu pro hodnoty parametrů:  $\alpha = 500$ ;  $\beta = \{1,0\}$ ;  $\mu = 20$ ;  $\sigma = 0,02$ ;  $A = 1$ ;  $B = 1$ .

#### 4.9 Thomas cluster proces s nehomogenními centry

Druhý typ nehomogenního Thomas cluster procesu bude, stejně jako u Matérn cluster procesu, nehomogenní při výběru středů (rodičů) clusteru.

Zadávané proměnné i větší část simulace se shoduje se simulací Thomas cluster nehomogenního procesu. Nejprve určíme pomocí Poissonova rozdělení počet bodů procesu. V dalším kroku rozmístíme středy clusterů (*CentersUp*), každému přiřadíme náhodné číslo (*RN*) a opět budeme zjišťovat, zda splňuje toto číslo podmínku, že je menší než skutečná intenzita dělená maximální intenzitou středů. Není-li tato podmínka splněna, bod nebudeme dále uvažovat.

Počet a rozmístění dceřinných procesů se simuluje stejným způsobem jako u klasického Thomas cluster procesu.

```

ThomasProcessInhomCenter = Function[{M, f, mu, sigma, MR, A, B},
  Module[{NumbPoint, CentersUp, RN, Dougters, PC, NumbDougters, fi, r, Pointik},

NumbPoint = Random[PoissonDistribution[M*(A*B + 2 (A + B)*MR + 4*MR^2)]];
  CentersUp = Table[{RandomReal[{-MR, A + MR}], RandomReal[{-MR, B + MR}]},
{NumbPoint}];
  RN = Table[RandomReal[{0, 1}], {NumbPoint}];
  PC = {};
  Do[If[RN[[i]] < f[CentersUp[[i]]]/M,

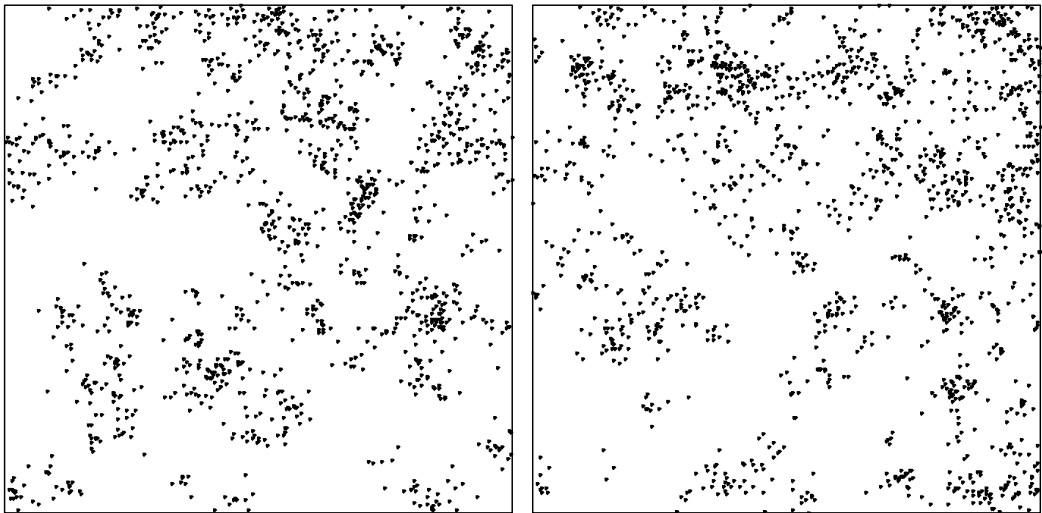
```

```

PC = Append[PC, CentersUp[[i]]]
, {i, 1, NumbPoint}];

Dougters = {};
Do[NumbDougters = Random[PoissonDistribution[mu]];
r = Table[Random[NormalDistribution[0, sigma]], {NumbDougters}];
fi = Table[RandomReal[{0, Pi}], {NumbDougters}];
Do[Pointik = {PC[[i, 1]] + Cos[fi[[j]]]*r[[j]],
PC[[i, 2]] + Sin[fi[[j]]]*r[[j]]};
If[Pointik[[1]] > 0 && Pointik[[2]] > 0 && Pointik[[1]] < A &&
Pointik[[2]] < B, Dougters = Append[Dougters, Pointik]];
, {j, 1, NumbDougters}];
, {i, 1, Length[PC]};
Dougters
];

```



Obr. 4.9: Příklady simulací Thomas cluster procesu s nehomogenními centry pro hodnoty parametrů:  $\alpha = 50$ ;  $\beta = \{1,0\}$ ;  $\mu = 10$ ;  $\sigma = 0,03$ ;  $A = 1$ ;  $B = 1$ .

#### 4.10 Odhad intenzity

První charakteristikou, jejíž odhad provedeme, je intenzita. V podstatě se jedná o zjištění četnosti bodů daného procesu. Jako proměnné tedy slouží název procesu, který nás zajímá a rozměry sledovaného okna. Výstupem je číslo udávající počet bodů.

```

Intenzita = Function[{Process, A, B},
Length[Process]/A/B
];

```

## 4.11 Odhad K funkce

Než se pustíme do samotného odhadu  $K$  funkce, bude nutné zadat do programu obecný vzorec pro výpočet vzdálenosti, neboť jej budeme ve zdrojovém kódu používat.

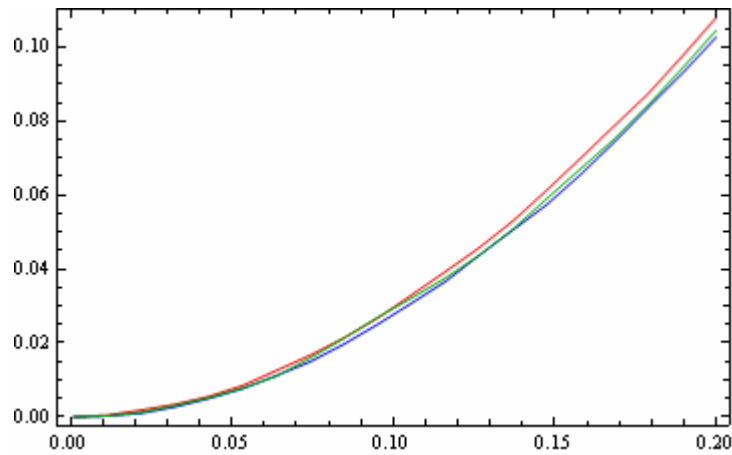
```
Vzd = Function[{x, y},  
  Sqrt[Total[(x - y)^2]]  
];
```

Odhad  $K$  funkce provádíme následujícím způsobem. Definujeme si funkci, jejíž první proměnnou bude proces, u kterého chceme  $K$  funkci sledovat. Dalšími proměnnými budou opět rozměry sledovaného okna – tedy  $A$  a  $B$ . Spodní a horní hranici vzdálenosti  $r$  nám určí  $r_0$  a  $r_1$ . Poslední proměnnou je počet mezikroků při výpočtu, který značíme  $N$ .

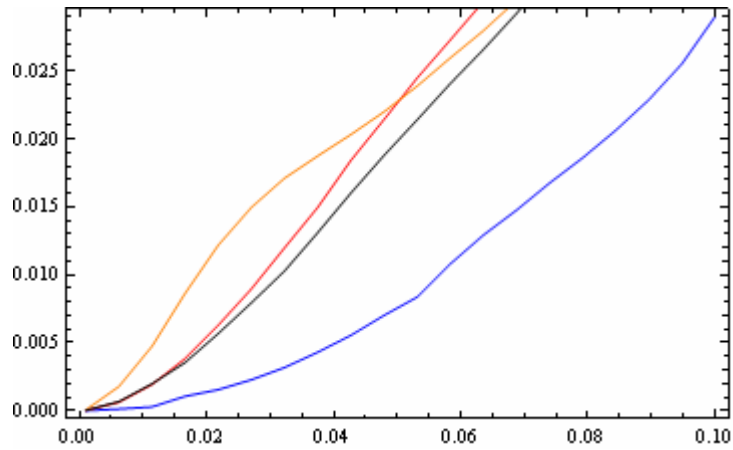
V programu nejprve vytvoříme tabulku čísel, která obsahuje samé nuly. V dalším kroku určíme intenzitu vybraného procesu ( $\rho$ ) a hustotu výskytu dvou bodů ve vzdálenosti  $r$  ( $\rho_2$ ). Následovně program prochází body od  $r_0$  do  $r_1$  a vytváří jejich dvojice (každou dvojici započítá jen jedenkrát). Je-li vzdálenost bodů ve dvojici menší než  $r$ , tak tyto body zařadíme do  $K$ .

Výslednou množinu bodů necháme vykreslit pomocí příkazu *ListPlot* do grafu.  $K$  funkce nám zjišťuje průměrný počet událostí (bodů) v určité vzdálenosti od dané události (bodu). Bohužel ale nevíme, jak vypadá graf pro nulovou prostorovou závislost, a proto nejsme schopni provést interpretaci grafů  $K$  funkcí.

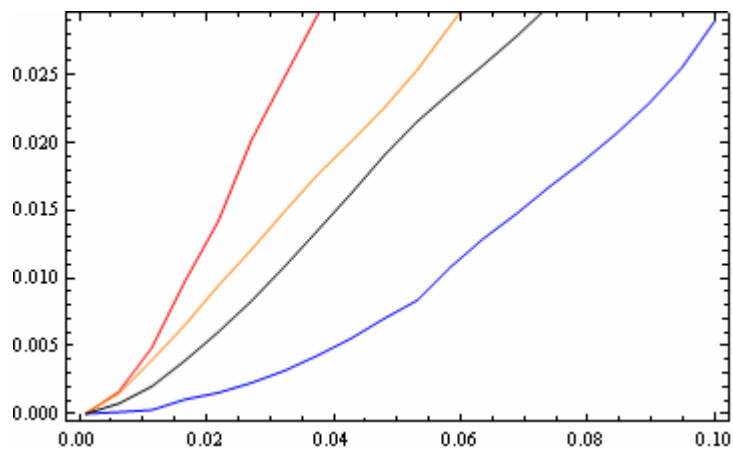
```
OdhadK = Function[{Process, A, B, r0, r1, N},  
  Module[{i, j1, j2, r, rho, rho2, K},  
    K = Table[0, {N}];  
    rho = Intenzita[Process, A, B];  
    rho2 = Length[Process]*(Length[Process] - 1)/A^2/B^2;  
    Do[r = r0 + (r1 - r0)/(N - 1)*(i - 1);  
      K[[i, 1]] = r;  
      Do[  
        Do[If[Vzd[Process[[j1]], Process[[j2]]] <= r,  
          K[[i, 2]] = K[[i, 2]] + 2/rho2/(A - (Process[[j1]] - Process[[j2]])[[1]])/  
            (B - (Process[[j1]] - Process[[j2]])[[2]])],  
          {j2, j1 + 1, Length[Process]}],  
        {j1, 1, Length[Process]}],  
      {i, 1, N}];  
    K];
```



Obr. 4.11.1: Grafy  $K$  funkcí pro Bernoulliho proces (červená barva), Poissonův proces (modrá barva) a Poissonův nehomogenní proces (zelená barva).



Obr. 4.11.2: Grafy  $K$  funkcí pro Matérn cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem  $K$  funkce pro homogenní Poissonův proces (modrá barva).



Obr. 4.11.3: Grafy  $K$  funkcí pro Thomas cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem  $K$  funkce pro homogenní Poissonův proces (modrá barva).

## 4.11 Odhad L funkce

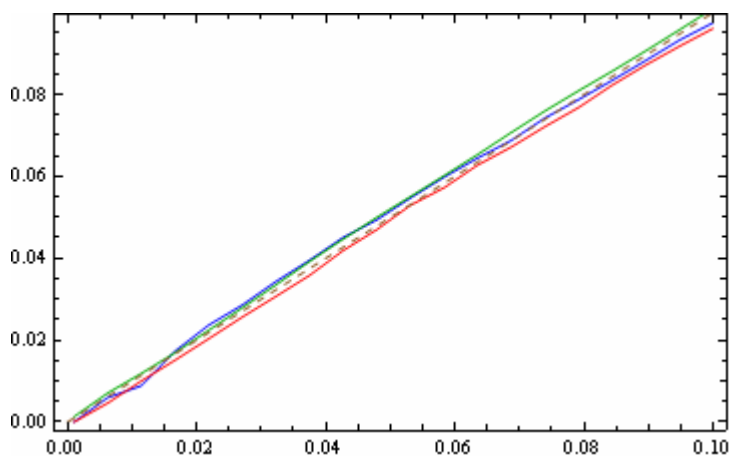
Odhad  $L$  funkce vychází z odhadu  $K$  funkce. Tudiž jsou i proměnné v zadané funkci totožné – tedy vybraný proces, rozměry sledovaného okna, dolní a horní hranice  $r$  a nakonec počet mezikroků výpočtu.

Samotný výpočet probíhá z odmocniny odhadu  $K$ , který dělíme  $\pi$  (což v našem případě odpovídá onomu  $\omega_d$  ve vzorci pro odhad  $L$ ). Tato kalkulace probíhá pro všechny body od 1 do  $N$ .

Výsledek opět zobrazíme přehledně v grafu, který by se měl během celého svého průběhu blížit grafu přímé úměry, neboť velikost  $L$  funkce se zvětšuje úměrně velikosti  $r$ . V místech, kde se graf vyhoupne nad tento graf, rozpoznáme clusterování a naopak v oblastech pod grafem clusterování končí.

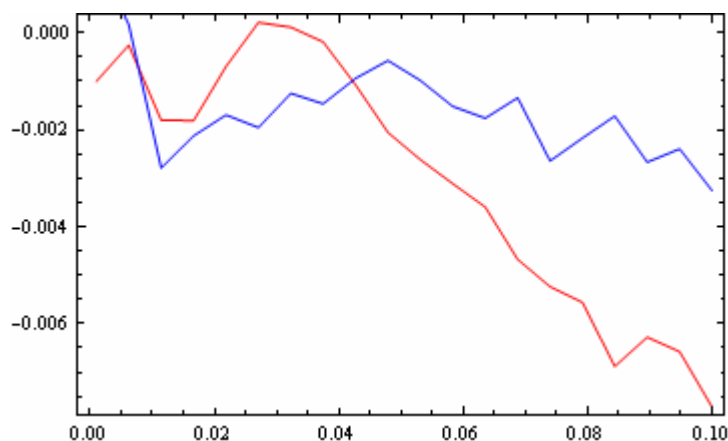
Někdy se zobrazuje raději graf  $L(r) - r$ , čímž se nám výsledné hodnoty posunou k nule a dále pak platí, že graf nad nulou značí shlukování bodů a graf pod nulou odpuzování bodů. Tento způsob se mi jeví jako přesnější, a proto se na něj při komentáři jednotlivých výsledků více zaměřím.

```
OdhadL = Function[{Process, A, B, r0, r1, N},  
  Module[{K, L},  
    K = OdhadK[Process, A, B, r0, r1, N];  
    L = Table[{K[[i, 1]], Sqrt[K[[i, 2]]/Pi]}, {i, 1, N}]  
  ];
```



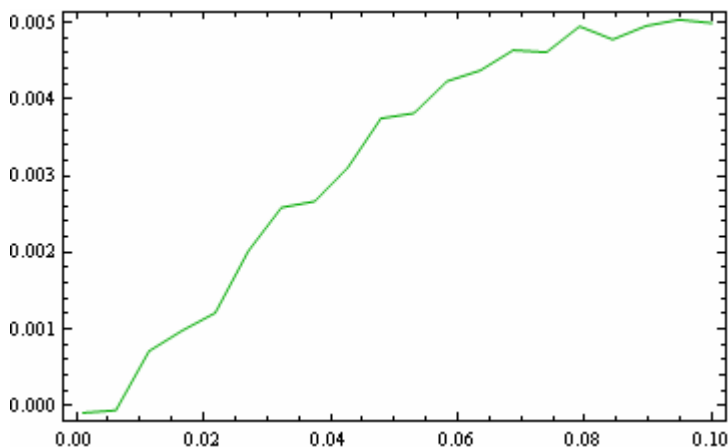
Obr. 4.11.1: Grafy  $L$  funkcí pro Bernoulliho proces (červená barva), Poissonův proces (modrá barva) a Poissonův nehomogenní proces (zelená barva) srovnané s grafem přímé úměry (hnědá barva – čárkovaná čára).

Homogenní Poissonův proces by se měl v tomto grafu shodovat s grafem přímé úměry, což vcelku odpovídá a totéž platí i pro ostatní zobrazované procesy. Je tedy možné učinit závěr, že se jedná o procesy bez shluků a bez odpuzujících se bodů.



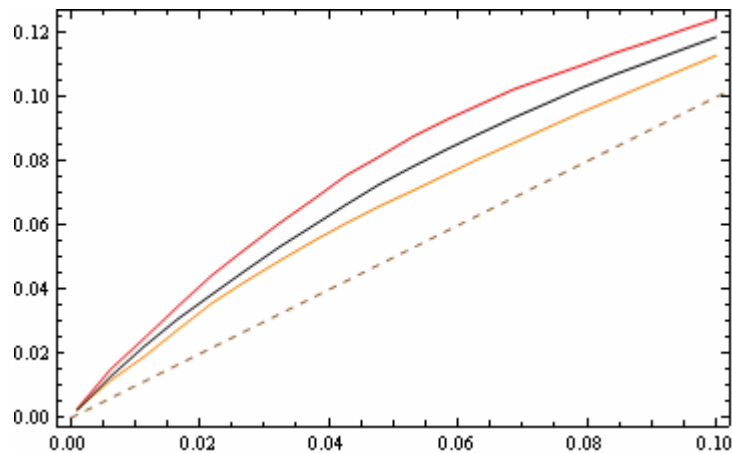
Obr. 4.11.2:  $L(r) - r$  grafy pro Bernoulliho proces (červená barva) a Poissonův homogenní proces (modrá barva).

Oba grafy mají podobný průběh a téměř v celé sledované oblasti se drží mírně pod nulou, což ukazuje rovnoměrnost obou procesů a současně potvrzuje tvrzení pro Poissonův proces, že jeho  $L(r) - r$  graf se má pohybovat v těsné blízkosti nuly.



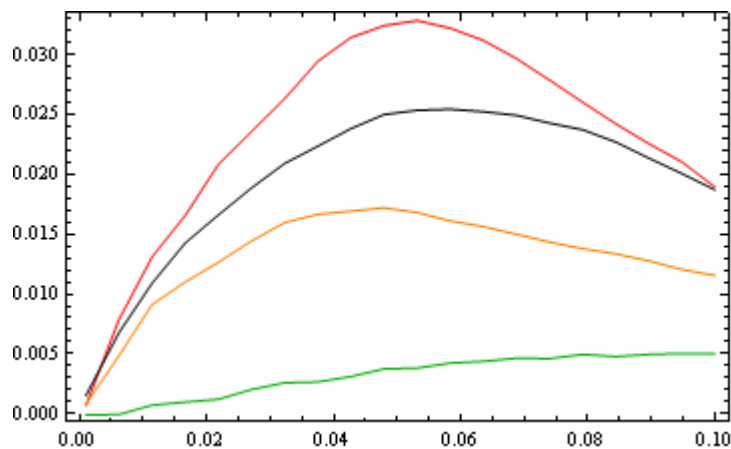
Obr. 4.11.3:  $L(r) - r$  graf pro nehomogenní Poissonův proces.

Umístění celého grafu nad nulou vypovídá o mírné clusterovitosti procesu. Toto je nejpatrnější pro větší hodnoty  $r$ , což jsme vzhledem k charakteru procesu očekávali.



Obr. 4.11.4: Grafy  $L$  funkcí pro Matérn cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem přímé úměry (hnědá barva – čárkovaná čára).

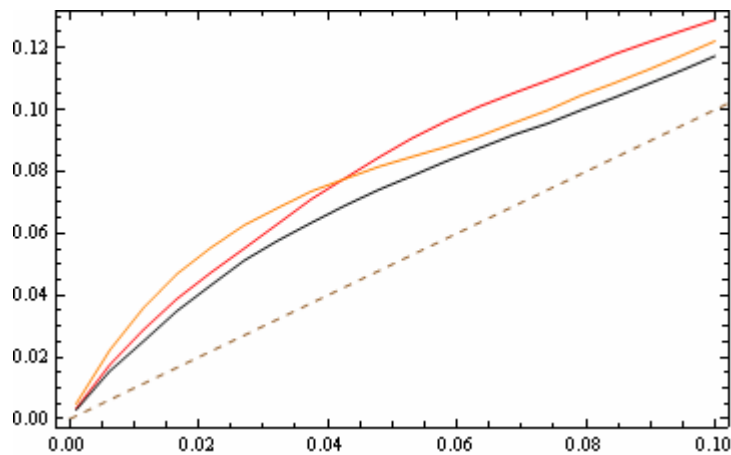
Vidíme, že všechny sledované grafy se po celou dobu svého průběhu drží nad grafem přímé úměry, což vypovídá o jejich clusterovistosti.



Obr. 4.11.5:  $L(r) - r$  grafy pro Matérn cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem funkce pro nehomogenní Poissonův proces (zelená barva).

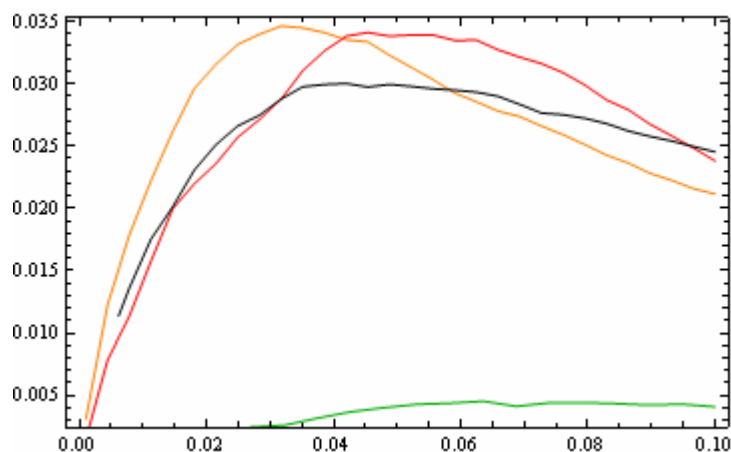
Z obrázku je patrné již na první pohled, že se jedná o cluster procesy, neboť máme veškeré hodnoty vysoko nad nulou. Nejvíce se shluky bodů projevují, podle očekávání, u homogenního Matérn cluster procesu, protože tady nejsou žádné z clusterů vymazány. Nejméně se shluky projevují u celkově nehomogenního procesu, což jsme mohly též předpokládat vzhledem k tomu, že v tomto procesu mažeme náhodné body a nikoliv středy clusterů.

Zajímavé je i srovnání s Poissonovým nehomogenním případem, který se na předchozím obrázku jevil jako proces shlukový. Nyní, při zasazení do kontextu, je jasné, že jeho clusterovitost je opravdu minimální. Nutno ještě připomenout, že homogenní Poissonův proces má graf v blízkosti nuly (v našem případě mírně pod nulou) a cluster procesy tedy můžeme považovat za jeho opak.



Obr. 4.11.6: Grafy  $L$  funkcí pro Thomas cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem přímé úměry (hnědá barva – čárkovaná čára).

I v tomto obrázku je zřejmé, že pozorujeme cluster procesy, neboť se grafy opět po celou dobu svého průběhu drží nad grafem přímé úměry.



Obr. 4.11.7:  $L(r) - r$  grafy pro Thomas cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem funkce pro nehomogenní Poissonův proces (zelená barva).



Výsledné grafy pro Thomas cluster procesy se podobají grafům Matérn procesů. Odlíšnosti jsou zde ale pozorovatelné jsou. Především zde  $L$  funkce dosahuje o něco vyšších hodnot. Dále není tolik patrná clusterovistost procesu s nehomogenními centry (graf se jeví ve své horní části téměř jako lineární). Množství clusterů u klasického Thomase je srovnatelné s množstvím clusterů u jeho nehomogenního případu a oba tyto procesy se k sobě blíží mnohem více, než u grafů Matérn cluster procesu.

## 4.12 Odhad $F$ funkce

Při odhadování  $F$  funkce hledáme pro každý bod minimální vzdálenost k některému zvolenému bodu procesu.

Odhad  $F$  zadáme jako funkci s následujícími proměnnými: proces, který chceme sledovat, rozměry okna ( $A$  a  $B$ ), dolní a horní omezení pro  $r$  ( $r_0$ ,  $r_1$ ), počet mezikroků ( $N$ ) a nakonec počet vybraných bodů ( $NSamplePoints$ ).

Nejprve vytvoříme tabulku ze samých nul. Dále zjistíme vzdálenosti mezi jednotlivými vybranými body (*InterPointDistance*) a počty bodů okna ve směru osy  $x$  ( $NA$ ) a také ve směru osy  $y$  ( $NB$ ).

Nyní přistoupíme k samotnému zjišťování nejkratší vzdálenosti (*NearestDistance*). Prověříme postupně všechny body procesu následujícím způsobem. Porovnáváme vzdálenost bodu procesu a bodu vybraného z tabulky. Pro každé  $r$  spočteme, kolik bodů z tabulky má vzdálenost menší než  $r$ . Takto získáme počet bodů mimo mřížku (*NPointsOut*) a posléze i počet bodů, které budeme procházet (*NIr*). Je-li nejkratší vzdálenost menší nebo rovna  $r$ , bod započteme.

Z výsledného grafu lze vyčíst, jestli mezi body převažují kratší nebo delší vzdálenosti.

```
OdhadF = Function[{Process, A, B, r0, r1, N, NSamplePoints},
  Module[{i, j1, j2, r, F, NA, NB, NPointsOut, InterPointDistance, NearestDistance,
  NIr},
```

```
  F = Table[{0, 0}, {N}];
  InterPointDistance = Sqrt[A*B/NSamplePoints];
  NA = IntegerPart[A/InterPointDistance];
  NB = IntegerPart[B/InterPointDistance];
```

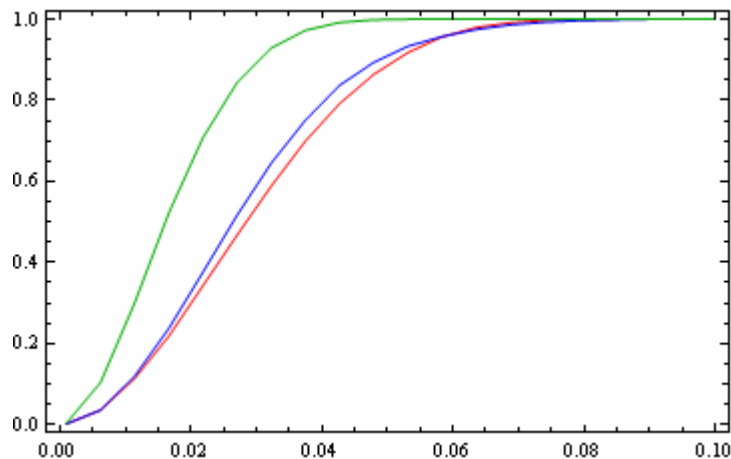
```

NearestDistance = Table[
  Min[
    Table[Vzd[Process[[k]], {i*InterPointDistance - InterPointDistance/2,
      j*InterPointDistance - InterPointDistance/2}], {k, 1, Length[Process]}],
    {i, 1, NA}, {j, 1, NB}];

Do[r = r0 + (r1 - r0)/(N - 1)*(i - 1);
  F[[i, 1]] = r;
  NPointsOut = IntegerPart[(r + InterPointDistance/2)/InterPointDistance];
  Nlr = (NA - 2*NPointsOut)*(NB - 2*NPointsOut);

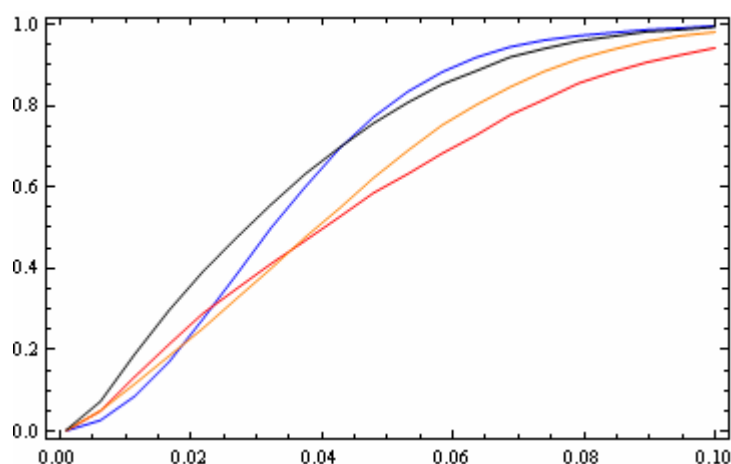
Do[
  Do[
    If[NearestDistance[[j1, j2]] <= r,
      F[[i, 2]] = F[[i, 2]] + 1/Nlr
    , {j2, 1 + NPointsOut, NB - NPointsOut}
    , {j1, 1 + NPointsOut, NA - NPointsOut}
    , {i, 1, N}];
  F
];

```



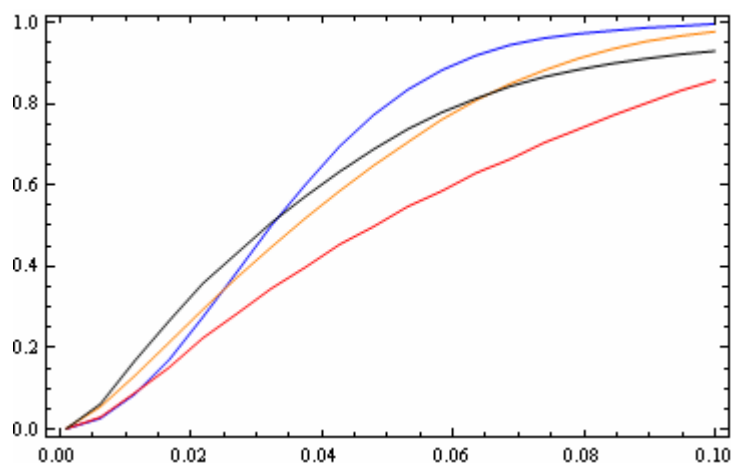
Obr. 4.12.1: Grafy  $F$  funkcí pro Bernoulliho proces (červená barva), Poissonův proces (modrá barva) a Poissonův nehomogenní proces (zelená barva).

Z grafů je zřejmá podobnost mezi Bernoulliho a homogenním Poissonovým procesem. Patrné je i to, že u nehomogenního Poissonova procesu najdeme větší množství kratších vzdáleností mezi body, na což ukazuje rychlejší vzestup jeho grafu.



Obr. 4.12.2: Grafy  $F$  funkcí pro Matérn cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem  $F$  funkce pro homogenní Poissonův proces (modrá barva).

Očekávala jsem, že grafy  $F$  funkcí se budou více podobat grafům  $G$  funkcí a tedy, že grafy shlukových procesů se budou mnohem více lišit od grafu homogenního Poissonova procesu a hlavně, že jejich nárůst bude strmější. V tomto případě to vypadá, jakoby vzdálenosti mezi vybranými body u cluster procesů byly větší, než u Poissonova procesu.



Obr. 4.12.3: Grafy  $F$  funkcí pro Thomas cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem  $F$  funkce pro homogenní Poissonův proces (modrá barva).

Obrázek pro průběh  $F$  funkcí u Thomas cluster procesů je velmi podobný předchozímu Matérn cluster procesům a tudíž i interpretace výsledků by byla dosti obdobná.

### 4.13 Odhad $G$ funkce

U  $G$  funkce budeme zjišťovat, jestli se ke sledovanému bodu do dané vzdálenosti  $r$  vyskytuje nějaký sousední bod. Jedná se tedy o zjištění vzdálenost k nejbližšímu sousednímu bodu (na rozdíl od  $F$  funkce, kde šlo o nejbližší vzdálenost k libovolnému zvolenému bodu procesu).

Proměnné jsou opět sledovaný proces, rozměry okna, dolní a horní omezení  $r$  a počet mezikroků.

Začneme, stejně jako u odhadu  $F$ , vytvořením tabulky ze samých nul. V dalším kroku necháme spočítat odhad intenzity. Nyní můžeme přistoupit ke zjišťování nejkratší vzdálenosti (*NearestDistance*), kterou budeme počítat pro každý bod procesu. Je zřejmé, že nejbližším sousedem by byl vždy daný bod sám sobě, a proto musíme tuto možnost vyloučit, což provedeme setříděním tabulky vzdáleností a vezmeme až druhou nejmenší vzdálenost. Abychom bod započítali, musí splňovat další dvě podmínky. Jeho nejkratší vzdálenost musí být menší nebo rovna  $r$  a musí být součástí menší vybrané skupiny bodů tvořící soubor  $W_r$ .

Interpretace výsledného grafu je podobná jako u odhadu  $F$  funkce. Opět sledujeme vzdálenosti mezi body. Např. pokud graf funkce na začátku strmě roste a pak se vyrovná, naznačuje to, že je zastoupeno větší množství krátkých vzdáleností a méně vzdáleností dlouhých, z čehož lze vyvodit, že se bude zřejmě jednat o vzorek se shluky bodů. Naopak, roste-li křivka až při konci rozsahu vzdáleností, naznačuje to opakování nebo pravidelnost.

```
OdhadG = Function[{Process, A, B, r0, r1, N},  
  Module[{i, j, r, rho, G, NearestDistance, NPointsWr},  
    G = Table[{0, 0}, {N}];  
    rho = Intenzita[Process, A, B];
```

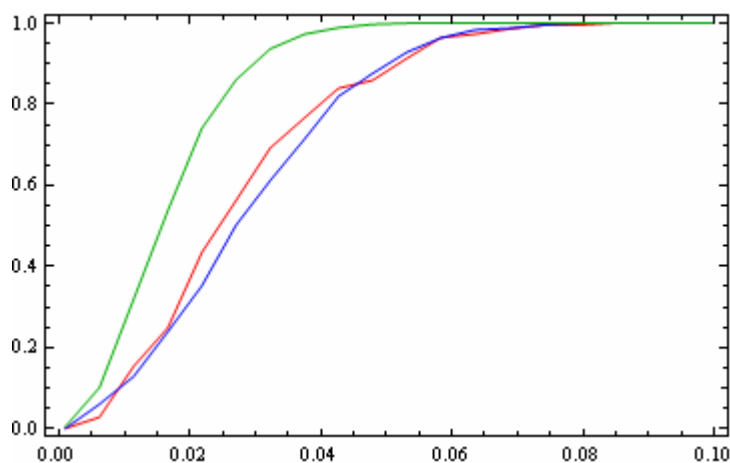
```

NearestDistance = Table[
  Sort[
    Table[Vzd[Process[[k]], Process[[i]], {k, 1,
      Length[Process]}][[2]], {i, 1, Length[Process]}];

Do[r = r0 + (r1 - r0)/(N - 1)*(i - 1);
  G[[i, 1]] = r;
  NPointsWr = 0;

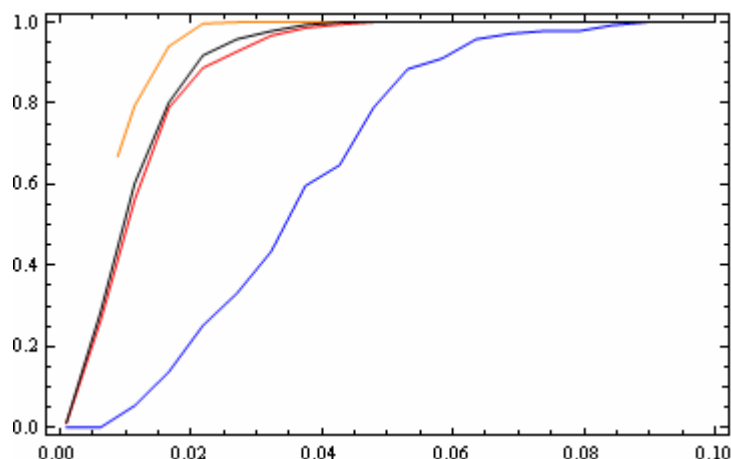
Do[
  If[Process[[j, 1]] > r && Process[[j, 1]] < A - r &&
    Process[[j, 2]] > r && Process[[j, 2]] < B - r, NPointsWr++;
  If[NearestDistance[[j]] <= r, G[[i, 2]]++]
  , {j, 1, Length[Process]}];
G[[i, 2]] = G[[i, 2]]/NPointsWr;
, {i, 1, N}];
G
];

```



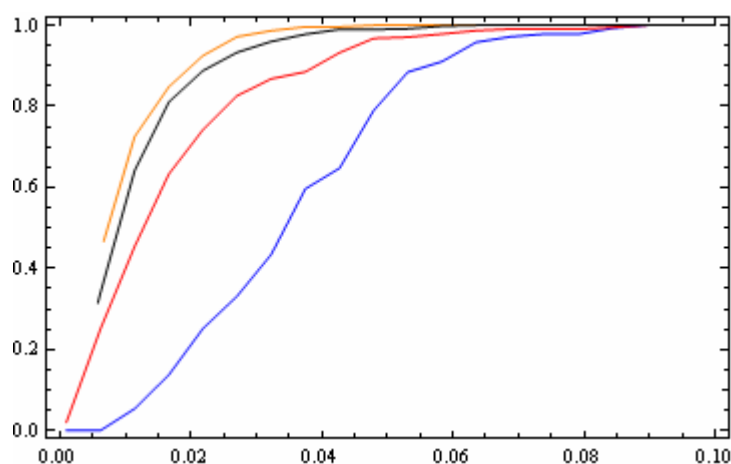
Obr. 4.13.1: Grafy  $G$  funkcí pro Bernoulliho proces (červená barva), Poissonův proces (modrá barva) a Poissonův nehomogenní proces (zelená barva).

Obrázek nám opět potvrzuje značnou podobnost u homogenního Poissonova a Bernoulliho procesu. Z grafu nehomogenního Poissona můžeme díky jeho strmějšímu nárůstu vyvodit, že se jedná o proces s kratšími vzdálenostmi mezi body, než bylo u obou předchozích procesů.



Obr. 4.13.2: Grafy  $G$  funkcí pro Matérn cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem  $G$  funkce pro homogenní Poissonův proces (modrá barva).

Obrázek názorně ukazuje rozdíl mezi clusterovými procesy a homogenním Poissonem. Na první pohled je vidět, že grafy Matérn procesu rostou mnohem strměji, což značí větší množství krátkých vzdáleností mezi body. Nejstrmější nárůst je u nehomogenního případu, který dokonce nezačíná v nule, ale až na hodnotě 0,01, protože pro menší hodnoty  $r$  se v tomto konkrétním případě nevyskytovaly v takové vzdálenosti žádné body. Všechny grafy Matérn procesů se už od hodnoty  $r$  blízko 0,04 mění na lineární funkce v těsné blízkosti u jedničky, zatímco graf Poissonova procesu se k ní pozvolna přiblíží až v hodnotě  $r$  kolem 0,09.



Obr. 4.13.3: Grafy  $G$  funkcí pro Thomas cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem  $G$  funkce pro homogenní Poissonův proces (modrá barva).

Klasický Thomas cluster proces se podobá Matérn cluster procesu, zatímco nehomogenní případy obou procesů se od sebe liší. Patrnější je tento rozdíl u případu s nehomogenními centry. V předchozím obrázku v podstatě kopíroval homogenní cluster proces, ale zde se spíše blíží nehomogennímu cluster procesu. Oba nehomogenní případy začínají až v hodnotě pod 0,01, neboť  $G$  funkce je do té doby záporná. O to strmější nárůst ale poté zaznamenávají a opět nám tak dokazují velké množství bodů v malé blízkosti od sebe – tedy shluky.

#### 4.14 Odhad $J$ funkce

Ke získání odhadu  $J$  funkce jsou třeba odhady předchozích dvou funkcí – tedy  $F$  funkce a  $G$  funkce. Do zadání proto použijeme stejné parametry jako výše.

Celý výpočet spočívá ve vygenerování odhadu  $F$  funkce, odhadu  $G$  funkce a jejich následné dosazení do vzorce pro kalkulaci odhadu  $J$  funkce, což je již zmiňovaný

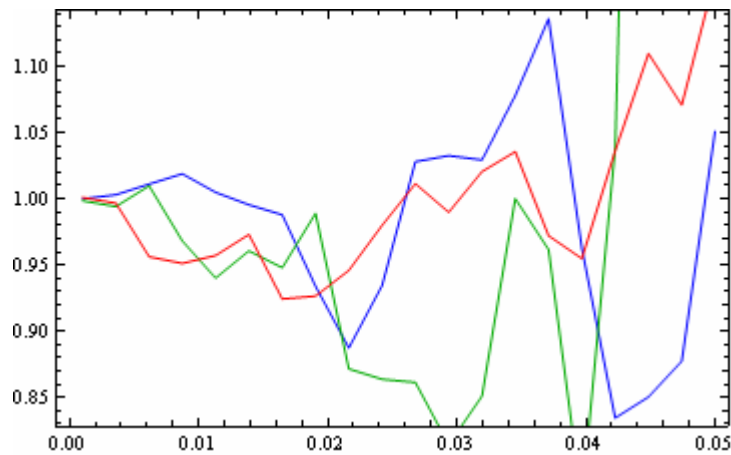
$$\text{vztah } \hat{J}(r) = \frac{1 - \hat{G}(r)}{1 - \hat{F}(r)}.$$

Z grafu je opět možno vyčíst, kdy se jedná o shluky a kdy o rovnoměrnost. Je-li totiž graf nad jedničkou, značí rovnoměrnost procesu. Naopak vyskytuje-li se graf pod jedničkou znamená to, že se jedná o shlukový proces.

```

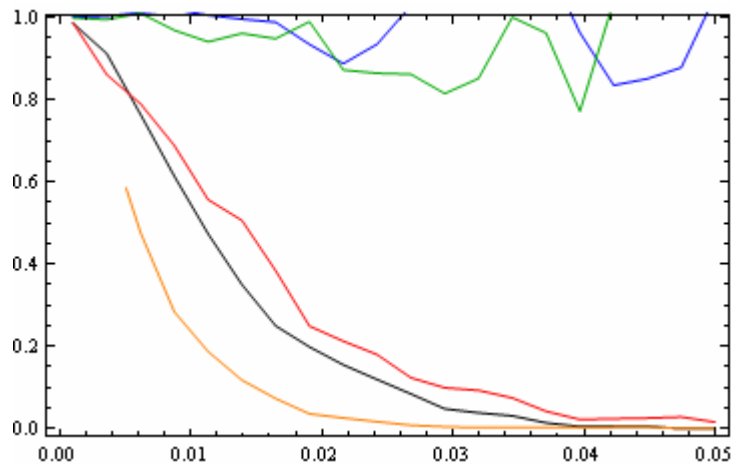
OdhadJ=Function[{Process,A,B,r0,r1,N,NSamplePoints},
Module[{F,G},
F=OdhadF[Process,A,B,r0,r1,N,NSamplePoints];
G=OdhadG[Process,A,B,r0,r1,N];
Table[{F[[i,1]],(1-G[[i,2]])/(1-F[[i,2]])},{i,1,N}
]];

```



Obr. 4.14.1: Grafy  $J$  funkcí pro Bernoulliho proces (červená barva), Poissonův proces (modrá barva) a Poissonův nehomogenní proces (zelená barva).

V případě odhadu  $J$  funkce by se měl homogenní Poissonův proces pohybovat v blízkosti jedničky, což v našem obrázku docela odpovídá, neboť se graf od této hodnoty vychyluje jen v malých odchylkách. Podobně je na tom i graf pro Bernoulliho proces. Největší rozpětí má graf nehomogenního případu, ale ani ten neklesá pod hodnotu 0,8 a nestoupá výše než 1,3.

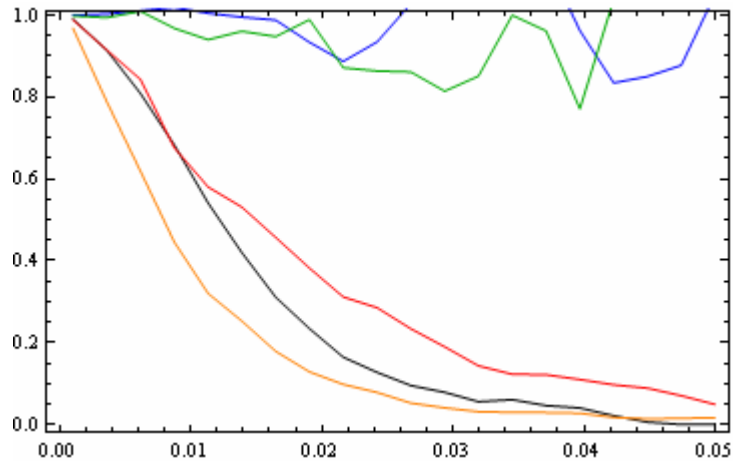


Obr. 4.13.2: Grafy  $J$  funkcí pro Matérn cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem  $J$  funkcí pro homogenní Poissonův proces (modrá barva) a jeho nehomogenní případ (zelená barva).

U všech grafů Matérn cluster procesu je na první pohled jasné, že jsou to procesy shlukové, protože jejich počátek je v jedničce a po celou dobu svého průběhu



monotónně klesají až k nule. Nejstrmější je tento pokles u nehomogenního případu, jehož graf navíc začíná až u hodnoty  $r$  přibližně v 0,005.



Obr. 4.13.3: Grafy  $J$  funkcí pro Thomas cluster proces (červená barva), jeho nehomogenní případ (oranžová barva) a případ s nehomogenními centry (černá barva) srovnané s grafem  $J$  funkcí pro homogenní Poissonův proces (modrá barva) a jeho nehomogenní případ (zelená barva).

Grafy pro všechny typy Thomas cluster procesů se opět podobají grafům pro Matérn cluster procesy. Tentokrát ovšem i nehomogenní případ začíná, stejně jako všechny ostatní grafy, v jedničce.

## 5. Příklad využití v praxi

Nyní se pokusím stručně vysvětlit jak se používá výše popsaná teorie v samotném praktickém zkoumání.

### 5.1 Data a jejich prvotní zpracování

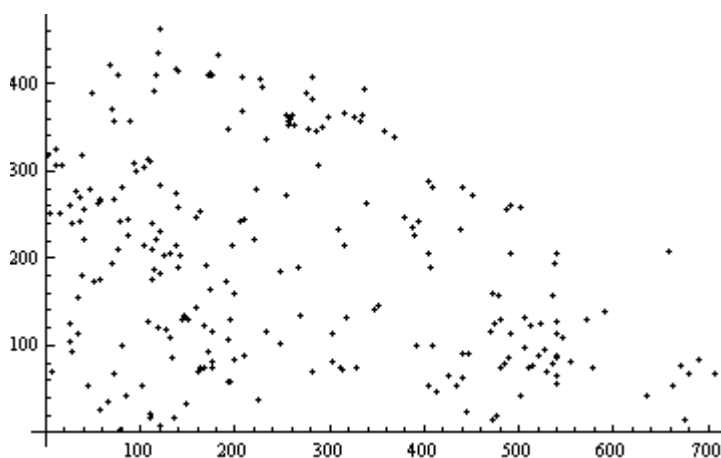
Nejdůležitější v celém procesu je získání potřebných dat. Tímto se ovšem zabývají spíše terénní pracovníci znalí dané problematiky.

Já jsem pro potřeby této práce získala data udávající pozice padlých stromů při vichřici na určitém místě ve Francii. Všechny tyto údaje jsou k nahlédnutí v příloze.

Prvním krokem, který je nutný provést, je převedení dat do řeči, které bude program Mathematica rozumět. To provedeme nejprve nahráním veškerých údajů do programu a příkazem `Take[data, {1, Length[data]}, {1, 2}]` je převedeme do podoby, které bude program rozumět.

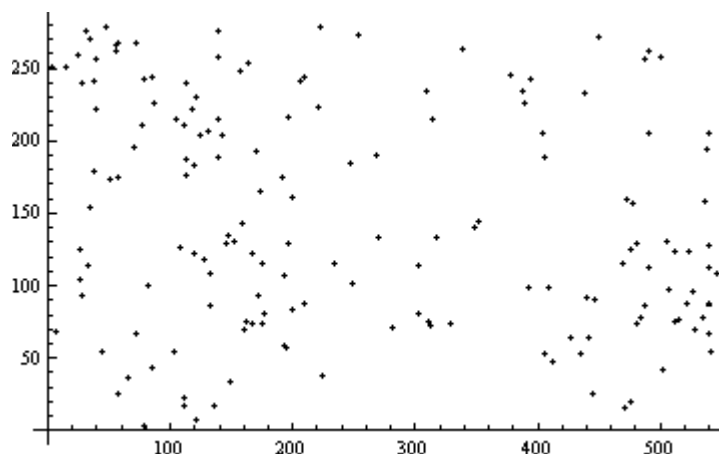
### 5.2 Zobrazení dat

V dalším kroku zadáme programu příkaz `ListPlot`, aby nám data zobrazil jako body do grafu. V našem případě jsme dostali následující obrázek.



Obr. 5.2.1: Pozice padlých stromů při vichřici.

Již na první pohled je zřejmé, že takto zadaná data by se pravděpodobně neshodovala s žádným z uvedených procesů, a proto vybereme jen jejich část (což provedeme vložení podmínky pro horní hranici první i druhé souřadnice) a necháme opět vygenerovat graf.



Obr. 5.2.2: Vybraná oblast z předchozího obrázku. Omezení pro souřadnice osy x je 550, pro osu y 280.

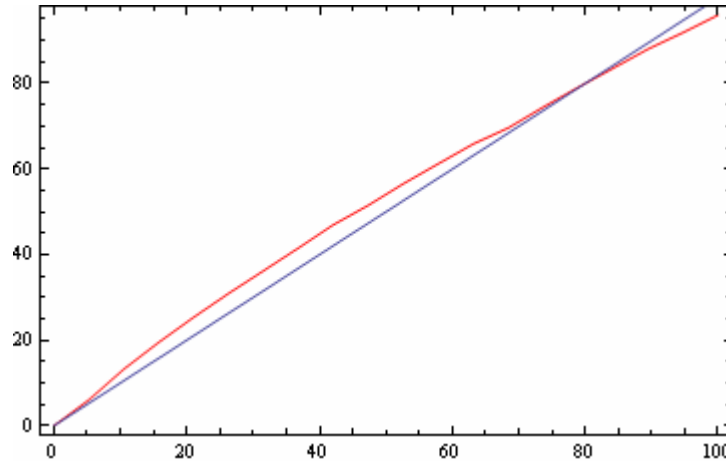
Nyní můžeme přistoupit k samotnému hodnocení zobrazených dat, neboť se zdá, že graf by mohl být podobný některému ze známých bodových procesů. Dejme tomu, že se bude jednat o Poissonův proces.

### 5.3 Vyhodnocování a srovnávání dat

Tato část je nejnáročnější, neboť je třeba provést odhady jednotlivých parametrů, které jsou potřebné při počítání charakteristik daného procesu. Vzhledem k tomu, že nemám dostatečné zkušenosti s touto činností a ani nemám vytvořené žádné pomocné programky, které by mi pomohly hodnotu parametrů určit, budu provádět následující odhady velmi intuitivně. Jedná se mi totiž především o ukázkou postupu a nikoliv o samotný výsledek.

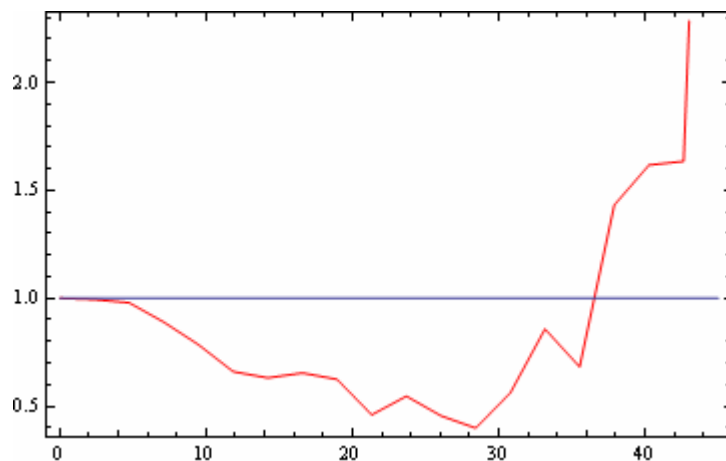
Nejjednodušší bude zjištění počtu bodů procesu, což provedu pomocí odhadu intenzity. V mém případě jsem dostala výsledek  $\frac{163}{154000}$ , z čehož vyčteme, že ve sledované části procesu je 163 bodů a plocha této oblasti je 154 000.

Nyní by bylo vhodné vyjádřit si  $L$  funkci nebo  $J$  funkci, protože u těch je jasné, jak má vypadat graf pro Poissonův proces. Porovnáme tedy graf nám zadaného procesu s grafem Poissonova procesu.



Obr. 5.3.1: Odhad  $L$  funkce pro padlé stromy (červená barva) srovnaná s homogenním Poissonovým procesem (modrá barva). Zadané hodnoty pro  $L$  funkci byly: náš proces, hranice sledovaného okna ( $A = 550$ ;  $B = 280$ ); hranice pro  $r$  dolní ( $r_0 = 0,001$ ) a horní ( $r_1 = 100$ ) a počet mezikroků výpočtu je 20.

Na první pohled se podle srovnání  $L$  funkcí zdá, že pozice padlých stromů budou velmi blízké Poissonovu procesu. Pro jistotu tuto domněnku ověříme pomocí  $J$  funkce.



Obr. 5.3.2: Odhad  $J$  funkce pro padlé stromy (červená barva) srovnaná s homogenním Poissonovým procesem (modrá barva). Zadané hodnoty pro  $J$  funkci byly: náš proces, hranice sledovaného okna ( $A = 550$ ;  $B = 280$ ); hranice pro  $r$  dolní ( $r_0 = 0,001$ ) a horní ( $r_1 = 45$ ); počet mezikroků výpočtu je 20 a počet vybraných bodů je 1000.

U této charakteristiky ( $J$  funkce) se již oba sledované grafy liší podstatně více – dalo by se říci, že čím větší hodnotu  $r$  dosazujeme, tím jsou rozdíly větší.

Budeme muset tedy přehodnotit svůj prvotní úsudek a zkusit vybrat jiný z procesů. Budou následovat opět stejné kroky, jako při srovnávání s Poissonovým procesem. Pokud by se nám nedařilo pozice padlých stromů ztotožnit s žádným se známých procesů, bylo by možné sledovanou oblast rozdělit na několik menších oken a vyslovit závěry aspoň pro ně. Např. bychom mohli zjistit, že větší část pozic odpovídá Poissonovu procesu, ale okrajové části se shodují spíše s nehomogenním Thomas cluster procesem a tak podobně.

## 6. Závěr

V této, poslední, části méj diplomové práce bych se ráda pokusila o shrnutí toho, co mi její tvorba přinesla, co se mi podařilo, kde zůstaly rezervy a pro koho by mohla být přínosem.

Pravděpodobně největší problém pro mě představoval nedostatek literatury v českém jazyce, neboť v odborné angličtině si nejsem příliš jistá a i kvůli nepřesnostem v překladu jsem se občas potýkala s neporozuměním teorií a z toho plynoucími chybami v programování. Samotné programování skrývalo hlavně zpočátku další komplikace způsobené především mojí povrchní znalostí programu Mathematica a také tím, že jsem během tvorby této práce přešla ze starší verze programu k novější a bylo třeba částečně upravit již hotové zdrojové kódy. Všechny tyto potíže se mi ale postupem času (a s tím souvisejícím nabíráním dalších zkušeností) povedlo snad zdárně překonat.

Celou práci jsem se snažila koncipovat jako základ pro další výzkum v oboru prostorové statistiky. I úplný začátečník v této oblasti matematiky (neboť tím jsem byla na počátku práce i já sama) by zde měl nalézt dostatečně vysvětlenou základní teorii potřebnou pro zvládnutí vytváření navazujících činností. Kromě nezbytné teorie ale tato práce obsahuje i podrobný návod pro tvorbu zdrojových kódů v programu Mathematica a to jak pro simulace samotných bodových procesů, tak i jejich charakteristik společně s pokusem o jejich interpretaci. Věřím, že toto se mi zdařilo a že je vše srozumitelné.

Přiznávám ale, že ne vše dopadlo podle původní plánu. Kvůli nedostatku času, způsobenému již zmíněnými počátečními problémy, jsem se bohužel nedostala k dostatečně hlubokému proniknutí do problematiky práce s konkrétními případy ze života. Při zadávání této diplomové práce jsem sice s touto činností nepočítala, ale při seznamování se s jednotlivými procesy a jejich charakteristikami mi připadalo důležité aspoň naznačit k čemu se veškerá tato teorie používá v praxi a co všechno si pod jednotlivými body v procesech můžeme představit.

Jsou ale i naopak témata, se kterými bylo na začátku počítáno a nakonec se v práci neobjevila. Jedná se o tzv. hard core procesy. Od simulace těchto dalších typů bodových procesů jsem upustila právě kvůli tomu, abych se mohla pokusit aspoň o částečné pochopení již zmiňované praktické činnosti. Věřím, že tato volba byla správná.

Díky této práci jsem si prohloubila a rozšířila znalosti získané na vysoké škole, neboť studium statistiky na takové úrovni nebylo v základu přednášek ani cvičení pro moji aprobaci obsaženo. Dále jsem se zdokonalila v anglickém jazyce a nahlédla jsem do tajů základů programování, se kterým jsem se do té doby nesetkala vůbec.

Na úplný závěr bych si troufla projevít přání, aby byla tato práce přínosem a jakýmsi odrazovým můstkem pro někoho, kdo v tomto oboru začíná a chtěl by se v něm propracovat dál než já.

## **7. Použitá literatura**

[1] MØLLER, J., WAAGEPETERSEN, R.P.: Statistical Inference and Simulation for Spatial Point Processes, Chapman & Hall/CRC, 2004.



## **PŘÍLOHY**

### Pozice padlých stromů při vichřici ve Francii

{{11.29', 324.38', 43.15', 323.41'}}, {2.48', 317.64', -4.65', 289.94'}, {16.24', 07.17', -4.39', 290.44'}, {10.72', 306.67', 23.66', 279.98'}, {31.18', 276, 33.3', 242.32'}, {34.92', 269.76', 34.8', 239.57'}, {24.34', 259.77', 39.26', 226.1'}, {15.02', 250.53', 44.14', 253.8'}, {3.23', 251.52', 38.84', 247.31'}, {39.13', 255.79', 49.05', 226.11'}, {36.81', 241.32', 68.4', 233.12'}, {39.75', 221.86', 72.39', 226.88'}, {33.72', 154.74', 21.32', 128.78'}, {33.31', 113.82', 2.39', 99.07'}, {25.74', 104.08', 50.52', 85.89'}, {27.71', 93.85', 57.77', 79.66'}, {6.79', 68.88', 37.34', 52.94'}, {57.56', 26.01', 26.92', 18}, {65.38', 36.25', 82.12', 15.31'}, {43.62', 53.95', 52.55', 27.51'}, {49.85', 173.72', 66.54', 142.29'}, {57.89', 175.22', 66.56', 146.54'}, {70.01', 194.94', 79.19', 167.01'}, {76.85', 210.67', 109.41', 195.98'}, {78.22', 242.36', 72.84', 212.91'}, {55.71', 262.3', 83.99', 241.86'}, {57.49', 267.54', 70.67', 236.86'}, {71.54', 267.55', 85.24', 239.62'}, {118.12', 436.26', 131.32', 411.57'}, {114.43', 391.1', 126.38', 65.41'}, {109.11', 311.75', 122.56', 286.06'}, {108.36', 312.5', 89.42', 282.54'}, {104.57', 305.51', 111.46', 271.83'}, {94.25', 299.51', 110.2', 270.33'}, {93.03', 308.24', 119.29', 285.31'}, {119.79', 283.81', 135.51', 262.12'}, {85.76', 244.36', 98.95', 216.18'}, {87.44', 225.9', 103.15', 199.21'}, {104.72', 215.43', 113.9', 188}, {111.22', 210.45', 132.7', 183.02'}, {117.04', 221.18', 139.77', 194.51'}, {113.39', 186.75', 139.92', 167.56'}, {113.1', 175.77', 143.65', 160.33'}, {119.65', 183.26', 145.94', 166.57'}, {102.59', 54, 117.8', 29.31'}, {135.32', 17.1', 161.17', 16.63'}, {138.33', 275.1', 147.48', 240.67'}, {181.09', 432.83', 202.45', 440.08'}, {196.06', 215.77', 227.61', 196.09'}, {173.29', 165.35', 182.95', 133.42'}, {195.73', 128.69', 229.08', 121.73'}, {233.07', 116, 232.7', 85.31'}, {221.19', 222.78', 239.42', 199.84'}, {206.2', 241.23', 229.45', 218.04'}, {209.47', 243.72', 237.54', 232.52'}, {277.13', 347.58', 293.33', 320.15'}, {254.25', 272.46', 286.12', 271.24'}, {248.07', 101.04', 264.06', 82.34'}, {280.58', 70.63', 281.23', 45.68'}, {314.76', 214.63', 308.12', 185.68'}, {309.31', 233.84', 309.7', 204.39'}, {404.98', 52.78', 420.16', 20.36'}, {412.23', 48.05', 424.66', 17.37'}, {426.1', 64.78', 436.02', 33.85'}, {434.59', 53.31', 434.49', 27.61'}, {441.4', 63.54', 431.48', 27.61'}, {408.16', 98.69', 413.84', 74.75'}, {392.36', 98.93', 418.74', 108.43'}, {389.33', 225.93', 397.48', 192.5'}, {501.55', 41.64', 510.76', 22.44'}, {488.17', 86.29', 469.77', 64.82'}, {481.34', 74.06', 503.39', 65.35'}, {483.87', 78.55', 489.03', 50.36'}, {511.21', 75.58', 500.85', 55.11'}, {514.48', 77.08', 530.47', 58.14'}, {491.03', 112.99', 477.63', 87.53'}, {505.65', 130.97', 504.06', 108.26'}, {512.15', 123.49', 490.01', 107.75'}, {490.88', 205.06', 507.65', 191.85'}, {535.87', 158.44', 526.74', 133.48'}, {523.2', 124.25', 534.41', 103.55'}, {539.96', 112.54', 558.44', 87.1'}, {546.22', 108.05', 562.71', 88.36'}, {526.6', 95.81', 554.67', 85.85'}, {521.8', 87.57', 541.78', 63.39'}, {539.36', 87.09', 537.25', 59.89'}, {535.56', 78.1', 534.97', 54.9'}, {528.76', 70.36', 549.77', 52.42'}, {539.79', 66.38', 530.89', 37.43'}, {540.75', 55.15', 559.5', 38.95'}, {636.05', 41.27', 634.46', 17.81'}, {675.6', 15.35', 687.32', -3.59'}, {661.95', 54.26', 675.94', 38.56'}, {689.41', 83.23', 690.83', 59.28'}, {47.25', 278.76', 36.33', 246.31'}, {680.57', 67.01', 699.87', 61.29'}, {707.17', 67.78', 720.91', 50.58'}, {577.21', 75.15', 600.03', 70.68'}, {571.65', 129.03', 593.23', 129.05'}, {591.51', 140.03', 613.83', 135.81'}, {540.32', 205.85', 559.12', 198.64'}, {537.77', 194.37', 551.52', 181.66'}, {540.02', 128.01', 562.38', 133.27'}, {501.12', 257.47', 507.99', 280.43'}, {490.6', 261.45', 504.45', 274.69'}, {449.99', 271.39', 469.1', 283.14'}, {439.73', 280.86', 461.27', 270.15'}, {439.05', 233.21', 451.27', 215}, {447.28', 90, 468.33', 83.53'}, {469.21', 115.72', 458.1', 97.74'}, {475.27', 124.95', 468.66', 103.99'}, {445.27', 25.12', 65.3',

12.92`}, {408.11`, 280.83`, 433.17`, 270.13`}, {405.38`, 287.82`, 430.76`, 295.58`},  
{367.44`, 339.43`, 384.2`, 325.97`}, {358.43`, 345.91`, 371.65`, 326.46`}, {393.91`,  
242.15`, 411.95`, 234.18`}, {378.12`, 245.88`, 398.16`, 237.41`}, {387.36`, 234.66`,  
408.14`, 223.2`}, {329.02`, 74.17`, 343.76`, 56.97`}, {316.95`, 132.79`, 333.71`,  
117.84`}, {347.85`, 140.56`, 360.84`, 125.6`}, {351.88`, 144.8`, 369.15`, 133.84`},  
{337.78`, 262.81`, 357.55`, 249.1`}, {331.62`, 357.11`, 360.45`, 350.4`}, {326.61`,  
360.85`, 347.17`, 356.13`}, {334.16`, 364.1`, 355.71`, 356.89`}, {337.03`, 393.55`,  
350.53`, 379.59`}, {316.35`, 366.08`, 336.9`, 360.61`}, {276.03`, 388.75`, 294.3`,  
375.29`}, {281.78`, 381.77`, 304.86`, 380.79`}, {298.51`, 362.57`, 317.28`, 348.87`},  
{291.44`, 350.09`, 311.2`, 332.64`}, {285.91`, 346.84`, 302.64`, 325.15`}, {288.52`,  
307.42`, 301.48`, 284.48`}, {302.07`, 113.32`, 312.79`, 95.61`}, {302.95`, 80.63`,  
318.43`, 61.19`}, {310.46`, 75.65`, 337.53`, 67.69`}, {312.7`, 72.91`, 312.11`, 49.7`},  
{269.02`, 133.75`, 293.62`, 133.27`}, {267.48`, 189.63`, 278.74`, 179.91`}, {257.09`,  
358.04`, 240.72`, 343.31`}, {257.35`, 358.79`, 272.61`, 348.33`}, {256.57`, 352.3`,  
267.8`, 335.6`}, {262.59`, 352.06`, 276.57`, 334.11`}, {254.1`, 363.78`, 254.26`,  
340.08`}, {259.63`, 365.03`, 260.55`, 343.33`}, {207.35`, 408.9`, 225.12`, 395.94`},  
{227.88`, 397.44`, 213.52`, 381.21`}, {206.44`, 368.98`, 223.26`, 371.24`}, {233.42`,  
335.82`, 229.83`, 317.85`}, {199.87`, 160.38`, 194.53`, 141.91`}, {199.07`, 83.78`,  
215.61`, 77.06`}, {194.2`, 57.33`, 202.75`, 62.33`}, {192.45`, 59.07`, 216.59`, 73.32`},  
{176.47`, 81.02`, 181.64`, 54.57`}, {160.12`, 69.77`, 186.47`, 70.05`}, {162.4`, 75.01`,  
184.69`, 64.56`}, {174.69`, 73.78`, 194.3`, 82.53`}, {166.66`, 74.27`, 193.53`, 80.03`},  
{193.13`, 106.73`, 217.21`, 102.26`}, {175.35`, 114.95`, 194.36`, 99`}, {167.6`, 121.68`,  
189.63`, 108.22`}, {158.65`, 143.38`, 163.82`, 119.68`}, {170.13`, 192.54`, 184.38`,  
178.83`}, {172.47`, 410.11`, 185.95`, 390.16`}, {175.98`, 409.12`, 189.21`, 390.67`},  
{173.99`, 412.11`, 196.55`, 406.14`}, {121.23`, 463.46`, 136.73`, 446.26`}, {136.62`,  
417.81`, 162.7`, 414.59`}, {140.12`, 415.57`, 167.71`, 410.61`}, {131.29`, 206.48`,  
151.28`, 184.54`}, {124.75`, 203.73`, 151.55`, 189.53`}, {146.82`, 134.63`, 175.42`,  
133.66`}, {151.32`, 130.15`, 177.46`, 141.9`}, {144.79`, 129.14`, 173.38`, 125.42`},  
{131.66`, 108.17`, 150.17`, 92.72`}, {132.83`, 86.96`, 158.19`, 91.23`}, {121.48`,  
7.11`, 141.24`, -10.33`}, {110.98`, 17.33`, 125.96`, -2.11`}, {111, 22.32`, 125.74`,  
5.87`}, {78.8`, 2.83`, 88.03`, -13.37`}, {71.52`, 66.7`, 88.76`, 45.26`}, {81.44`, 99.64`,  
109.56`, 103.91`}, {79.38`, 282.28`, 103.67`, 268.83`}, {89.71`, 357.64`, 69.58`,  
343.9`}, {66.87`, 420.74`, 77.07`, 400.04`}, {75.86`, 409.28`, 83.06`, 390.07`},  
{48.43`, 390.54`, 63.69`, 376.83`}, {70.44`, 371.84`, 63.34`, 351.38`}, {71.14`,  
357.37`, 54.26`, 340.89`}, {56.23`, 265.79`, 44.66`, 259.29`}, {37.33`, 179.19`, 57.6`,  
165.24`}, {28.28`, 240.31`, 34.69`, 211.63`}, {116.52`, 411.06`, 126.93`, 378.88`},  
{112.09`, 239.89`, 129.3`, 212.21`}, {120.33`, 229.92`, 144.62`, 215.22`}, {148.94`,  
33.33`, 170.18`, 10.9`}, {138.6`, 214.97`, 156.38`, 205.5`}, {141.81`, 203.74`, 143.95`,  
170.81`}, {139.5`, 189.02`, 136.37`, 157.58`}, {472.39`, 160.13`, 480.07`, 134.69`},  
{480.3`, 128.95`, 473.17`, 100.5`}, {506.78`, 97.29`, 512.23`, 78.08`}, {540.37`,  
88.09`, 561.49`, 97.59`}, {554.9`, 80.36`, 552.77`, 50.42`}, {671.07`, 77.23`, 660.7`,  
57.01`}, {658.27`, 207.21`, 677.14`, 217.96`}, {487.07`, 256.21`, 509.43`, 264.96`},  
{477.9`, 156.64`, 498.28`, 169.89`}, {440.51`, 91.49`, 460.52`, 75.29`}, {475.62`,  
20.16`, 452.73`, 7.41`}, {471.83`, 15.42`, 454.95`, -2.55`}, {406, 188.51`, 432.1`,  
189.04`}, {403.55`, 204.98`, 419.54`, 185.78`}, {281.88`, 407.72`, 305.17`, 396.26`},  
{246.89`, 185.12`, 269.26`, 193.63`}, {227.42`, 406.42`, 243.15`, 386.47`}, {222.41`,  
278.67`, 212.31`, 263.44`}, {208.87`, 88.03`, 217.57`, 66.08`}, {223.48`, 37.64`,  
222.37`, 9.95`}, {171, 92.99`, 197.58`, 88.52`}, {190.89`, 174.34`, 210.92`, 163.38`},

{157.79`, 247.42`, 186.64`, 243.45`}, {162.84`, 253.66`, 184.36`, 238.46`}, {191.8`,  
347.01`, 195.47`, 321.56`}, {138.76`, 258.13`, 157.78`, 243.18`}, {127.68`, 118.65`,  
133.31`, 144.85`}, {118.66`, 121.63`, 140.57`, 142.61`}, {108.14`, 126.36`, 136.5`,  
126.39`}, {85.23`, 43.26`, 94.92`, 15.82`}, {38.12`, 318.92`, 66.97`, 318.45`}, {25.58`,  
125.54`, 46.84`, 108.84`}}