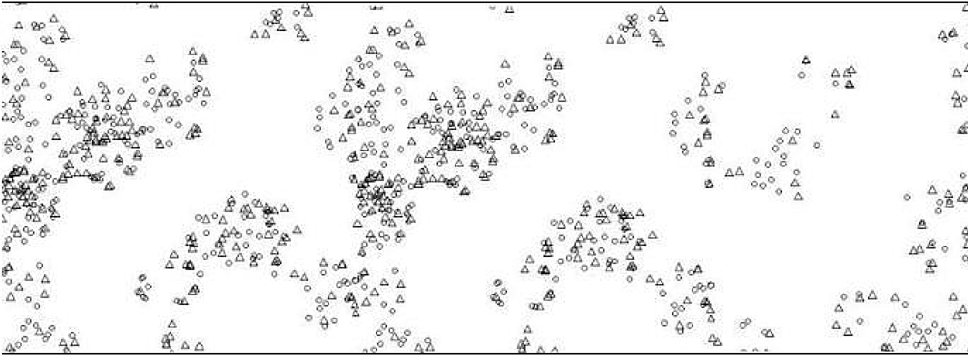


Stochastische Geometrie und ihre Anwendungen
Thema 4: Punktprozessmodelle

Martin Weinberger | 02. November 2009 | Uni Ulm

Überblick

Überblick

Einführung

Φ_b : Basis-Punktprozess (auch Vater-Prozess) in $\mathbb{R}^d$
Φ : Modellierter Punktprozess in $\mathbb{R}^d$
Λ : Intensitätsmaß des Punktprozesses
λ : Intensität (falls Φ stationär)
$\lambda(x)$: Intensitätsfunktion (falls Φ nicht stationär)

Überblick

Einführung

- Ausgangspunkt: Es wurde ein (Basis-)Punktprozess erzeugt
- Ziel: darauf neue Prozesse generieren

Einführung

- Ausgangspunkt: Es wurde ein (Basis-)Punktprozess erzeugt
- Ziel: darauf neue Prozesse generieren

Fundamentale Operationen

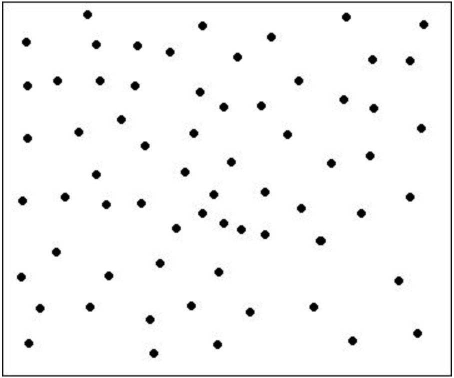
- Ausdünnung (thinning)
- Clustering
- Überlagerung (superposition)
- Komprimieren (pressing)

Fundamentale Operationen

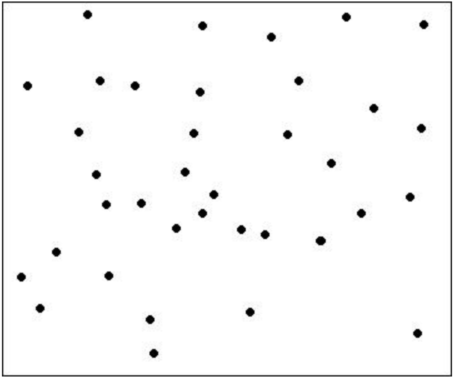
- Ausdünnung (thinning)
- Clustering
- Überlagerung (superposition)
- Komprimieren (pressing)

Ausdünnung

Ausdünnung



Ausdünnung



Ausdünnung

- Punkte des Basisprozesses Φ_b werden nach gewissen Regeln ausgedünnt, so dass gilt: $\Phi \subset \Phi_b$
- einfachste Variante: p-Ausdünnung
  - Jeder Punkt wird mit Wahrscheinlichkeit $1 - p$ (bzw. $1 - p(x)$) 'gelöscht'
  - p kann von x abhängen und auch selbst zufällig sein
  - Unabhängige Ausdünnung (da keine 'Interaktionen' zwischen den Punkten)

Ausdünnung

- Punkte des Basisprozesses Φ_b werden nach gewissen Regeln ausgedünnt, so dass gilt: $\Phi \subset \Phi_b$
- einfachste Variante: p-Ausdünnung
  - Jeder Punkt wird mit Wahrscheinlichkeit $1 - p$ (bzw. $1 - p(x)$) 'gelöscht'
  - p kann von x abhängen und auch selbst zufällig sein
  - Unabhängige Ausdünnung (da keine 'Interaktionen' zwischen den Punkten)

Ausdünnung

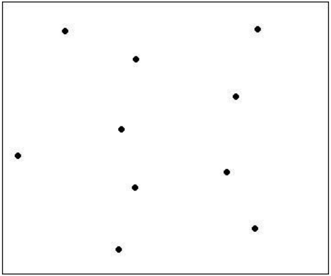
- p(x)-Ausdünnung:
  - Intensitätsmaß: $\Lambda(Y) = \int_Y p(x)\Lambda_b(dx)$
  - erzeugendes Funktional: $G(v) = G_b(v_p)$
    (mit $v_p(x) = v(x)p(x) + 1 - p(x)$)
- Abhängige Ausdünnung
  - z.B. Ausdünnung abhängig von der Distanz zum nächsten Nachbarn $\rightarrow$ Matérn-Hardcore-Prozess

Ausdünnung

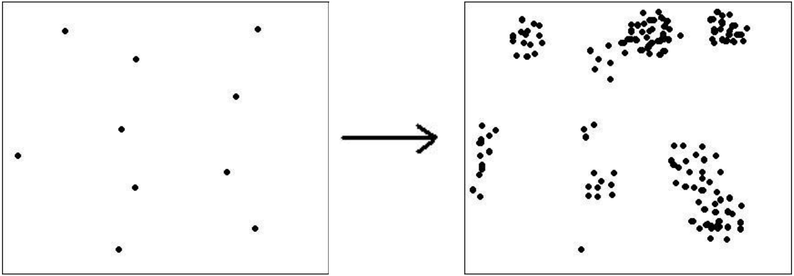
- p(x)-Ausdünnung:
  - Intensitätsmaß: $\Lambda(Y) = \int_Y p(x)\Lambda_b(dx)$
  - erzeugendes Funktional: $G(v) = G_b(v_p)$
    (mit $v_p(x) = v(x)p(x) + 1 - p(x)$)
- Abhängige Ausdünnung
  - z.B. Ausdünnung abhängig von der Distanz zum nächsten Nachbarn $\rightarrow$ Matérn-Hardcore-Prozess

Clustering

Clustering



Clustering



Clustering

- Idee: Auf Mutterprozess Φ_b wird nach einem stoch. Mechanismus einen Nachkommen-Prozess Φ erzeugt
- Jeder ursprüngliche Punkt x wird zu einem 'Cluster' von N^x Punkten
- Es wird angenommen, dass die Punkte verschiedener Cluster disjunkt sind: $N^x \cap N^y = \emptyset$ für $x \neq y$, $x, y \in \Phi_b$

Definition

Der gesamte Prozess kann wie folgt dargestellt werden:
$\Phi = \bigcup_{x \in \Phi_b} N^x$

Clustering

- Idee: Auf Mutterprozess Φ_b wird nach einem stoch. Mechanismus einen Nachkommen-Prozess Φ erzeugt
- Jeder ursprüngliche Punkt x wird zu einem 'Cluster' von N^x Punkten
- Es wird angenommen, dass die Punkte verschiedener Cluster disjunkt sind: $N^x \cap N^y = \emptyset$ für $x \neq y$, $x, y \in \Phi_b$

Definition

Der gesamte Prozess kann wie folgt dargestellt werden:
$\Phi = \bigcup_{x \in \Phi_b} N^x$

Clustering

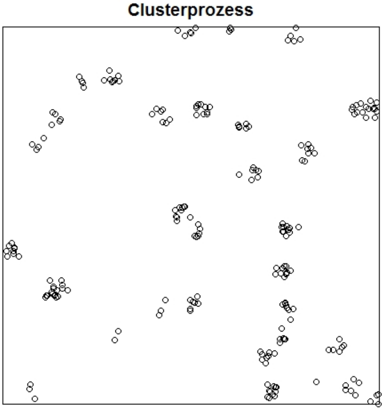
- Jedes Cluster ist für sich selbst ein Punktprozess mit nur endlich vielen Punkten
- Ausdünnung kann als Spezialfall von Clustering gesehen werden
- viele Naturphänomene kann man durch Clustering modellieren
- wichtiger Cluster-Prozess: Neyman-Scott-Prozess

Clustering - Beispiel

- Beispiel: Poisson-Clusterprozess:
  - 1. Mutterprozess ist (inhomogener) Poisson-Prozess mit Intensität $\lambda(x)$
  - 2. Zu jedem Punkt x des Mutterprozesses werden zufällige Anzahl N an Nachkommen erzeugt
  - 3. Die Positionen der Nachkommen relativ zum Mutterpunkt sind gemäß einer eindim. Randdichte $f(x)$ zufällig verteilt
  - 4. Die Lokationen der Nachkommen bilden den finalen PCP

Beispiel: Clusterprozess mit R

Beispiel: Clusterprozess mit R



Überlagerung (superposition)

Definition

Vereinigung zweier Punktprozesse Φ_1 und Φ_2 mit:

$$\Phi = \Phi_1 \cup \Phi_2$$

- Es wird dabei angenommen, dass sich die Punktmengen nicht überschneiden: $\Phi_1 \cap \Phi_2 = \emptyset$
- Intensitätsmaß $\Lambda = \Lambda_1 + \Lambda_2$
- Intensität $\lambda = \lambda_1 + \lambda_2$
- Falls Φ_1 und Φ_2 unabhängig sind: $G(v) = G_1(v) * G_2(v) \ \forall v \in V$

Überlagerung (superposition)

Definition

Vereinigung zweier Punktprozesse Φ_1 und Φ_2 mit:

$$\Phi = \Phi_1 \cup \Phi_2$$

- Es wird dabei angenommen, dass sich die Punktmengen nicht überschneiden: $\Phi_1 \cap \Phi_2 = \emptyset$
- Intensitätsmaß $\Lambda = \Lambda_1 + \Lambda_2$
- Intensität $\lambda = \lambda_1 + \lambda_2$
- Falls Φ_1 und Φ_2 unabhängig sind: $G(v) = G_1(v) * G_2(v) \ \forall v \in V$

Komprimieren (pressing)

Definition

Pressing: $\Phi_b \to \Phi$: $(\xi, \eta) \to (\xi', \eta')$
wobei $\xi' = \xi$, $\eta' = c\eta$; mit $c, \xi, \eta \in \mathbb{R}$ und $0 < c < 1$ fest

- Intensität: $\lambda = \lambda_b / c$
- Falls Φ_b stationär $\Rightarrow$ Φ stationär
- aber: i.A. ist Φ nicht isotrop, falls Φ_b es ist

Komprimieren (pressing)

Definition

$$\text{Pressing: } \Phi_b \to \Phi : \quad (\xi, \eta) \to (\xi', \eta')$$

wobei $\xi' = \xi, \quad \eta' = c\eta$; mit $c, \xi, \eta \in \mathbb{R}$ und $0 < c < 1$ fest

- Intensität: $\lambda = \lambda_b / c$
- Falls Φ_b stationär $\Rightarrow$ Φ stationär
- aber: i.A. ist Φ nicht isotrop, falls Φ_b es ist

Überblick

Cox-Prozesse (Doppelt stochastische Prozesse)

Definition

Ein PP Φ heißt Cox-Prozess, falls die Verteilung P_Φ durch $P_\Phi(Y) = \int P_\Lambda(Y) Q(d\Lambda)$ für $Y \in \mathcal{N}$ gegeben ist.

- Ein CPP ist eine 2-stufige Simulation:
  - 1. Generiere ein Maß Λ auf $\mathbb{R}^d$ (mit Verteilung Q)
  - 2. Generiere einen Poisson-Prozess mit Intensitätsmaß Λ
- Anwendungen v.a. in der Physik;
  erste Anwendungen auch bei Orten zweier verschiedener Baumarten

Cox-Prozesse (Doppelt stochastische Prozesse)

Definition

Ein PP Φ heißt Cox-Prozess, falls die Verteilung P_Φ durch $P_\Phi(Y) = \int P_\Lambda(Y) Q(d\Lambda)$ für $Y \in \mathcal{N}$ gegeben ist.

- Ein CPP ist eine 2-stufige Simulation:
  - 1. Generiere ein Maß Λ auf $\mathbb{R}^d$ (mit Verteilung Q)
  - 2. Generiere einen Poisson-Prozess mit Intensitätsmaß Λ
- Anwendungen v.a. in der Physik;
  erste Anwendungen auch bei Orten zweier verschiedener Baumarten

Beispiele von Cox-Prozessen

1. gemischter Poisson-Prozess
    - stationärer Poisson-Prozess mit zufälliger Intensität
    - Zufallsmaß: $\Lambda = \chi \nu_d \quad \chi$ nicht-neg. ZV
2. $\pi(x)$-Ausdünnung (für Poisson-Prozesse):
    - Zufallsmaß: $\Lambda(Y) = \int_Y \pi(x)\Lambda_b(dx) \quad Y \in \mathcal{B}$
3. Punktstreuung entlang Fasern
    - Zufallsmaß: $\Lambda(Y) = N_L h_1(\chi \cap Y)$ für $Y \in \mathcal{B}$

Beispiele von Cox-Prozessen

1. gemischter Poisson-Prozess
   - stationärer Poisson-Prozess mit zufälliger Intensität
   - Zufallsmaß: $\Lambda = \chi v_d \quad \chi$ nicht-neg. ZV
2. $\pi(x)$-Ausdünnung (für Poisson-Prozesse):
   - Zufallsmaß: $\Lambda(Y) = \int_Y \pi(x)\Lambda_b(dx) \quad Y \in \mathcal{B}$
3. Punktstreuung entlang Fasern
   - Zufallsmaß: $\Lambda(Y) = N_L h_1(\chi \cap Y)$ für $Y \in \mathcal{B}$

Beispiele von Cox-Prozessen

1. gemischter Poisson-Prozess
   - stationärer Poisson-Prozess mit zufälliger Intensität
   - Zufallsmaß: $\Lambda = \chi v_d \quad \chi$ nicht-neg. ZV
2. $\pi(x)$-Ausdünnung (für Poisson-Prozesse):
   - Zufallsmaß: $\Lambda(Y) = \int_Y \pi(x)\Lambda_b(dx) \quad Y \in \mathcal{B}$
3. Punktstreuung entlang Fasern
   - Zufallsmaß: $\Lambda(Y) = N_L h_1(\chi \cap Y)$ für $Y \in \mathcal{B}$

Cox-Prozesse

- erzeugendes Funktional:
  $G(v) = L_Q(1 - v)$ für $v \in V$

Überblick

Neyman-Scott Prozesse

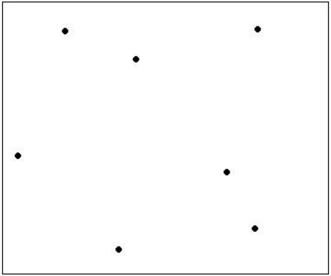
- spezielle Art eines Cox-Prozesses
- entsteht durch homogenes, unabhängiges Clustering auf stationärem Poisson-Prozess
  - parent-Punkte: stationärer Poisson-Prozess
  - offspring-Punkte: bilden zufällige, unabhängige Cluster mit jeweils identischer Verteilung relativ zum Elternpunkt (parent-Punkt)
- Intensität: $\lambda = \lambda_b \overline{c}$
  $\overline{c}$: Mittelwert der Anzahl der Tochterpunkte pro Elternpunkt

Neyman-Scott Prozesse

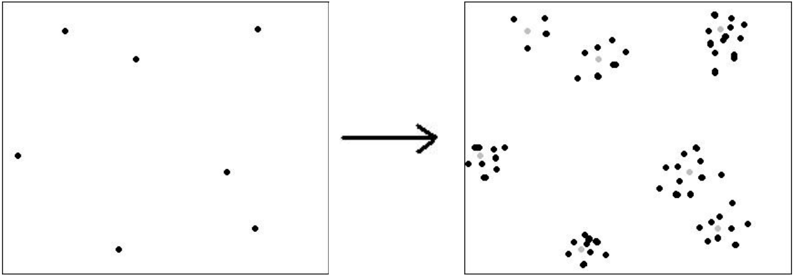
- Modell: (ähnlich Poisson-PP)
  - 1. Mutter-Prozess ist Poisson-Prozess
  - 2. Cluster-Muster der Nachkommen relativ zur Mutter i.i.d. verteilt
  - 3. Die Lokationen der Nachkommen bilden den finalen NSP

Neyman-Scott Prozesse

Neyman-Scott Prozesse



Neyman-Scott Prozesse



Überblick

Hard-Core Prozesse

Definition

Gegeben sei ein markierter PP Φ_b mit Marken $m(x) \;\; \forall x \in \Phi_b$. Dann heißt

$$\Phi = \{x \in \Phi_b : m(x) < m(y) \;\; \forall y \in \Phi_b \cap b(x,h) \setminus \{x\}\}$$

Hardcore-Prozess.

- Punkte haben Mindestabstand zum Nachbarpunkt $\rightarrow$ abhängige Ausdünnung (stationär)
- Intensität: $\lambda = p\lambda_b$; p: "Palm-Eintrittswahrscheinlichkeit"

Hard-Core Prozesse

Definition

Gegeben sei ein markierter PP Φ_b mit Marken $m(x) \ \ \forall x \in \Phi_b$. Dann heißt

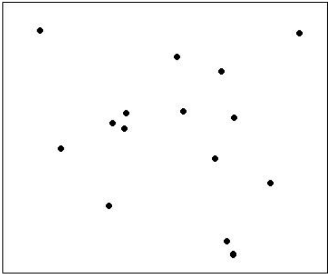
$$\Phi = \{x \in \Phi_b : m(x) < m(y) \ \ \forall y \in \Phi_b \cap b(x,h) \setminus \{x\}\}$$

Hardcore-Prozess.

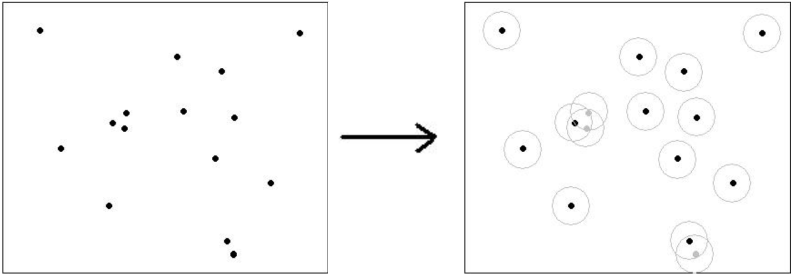
- Punkte haben Mindestabstand zum Nachbarpunkt $\rightarrow$ abhängige Ausdünnung (stationär)
- Intensität: $\lambda = p\lambda_b$; p: "Palm-Eintrittswahrscheinlichkeit"

Hard-Core Prozesse

Hard-Core Prozesse



Hard-Core Prozesse



Matérn-Hardcore Prozess

- Einfache Simulation:
  1. Simulation eines Poisson-Prozesses
  2. Punkte werden unabhängig markiert (mit $m(x) \in (0, 1)$)
  3. Ausdünnung nach Definition des Hardcoreprozesses

Matérn-Hardcore Prozess

- Einfache Simulation:
  1. Simulation eines Poisson-Prozesses
  2. Punkte werden unabhängig markiert (mit $m(x) \in (0, 1)$)
  3. Ausdünnung nach Definition des Hardcoreprozesses

Literaturverzeichnis:

- Stoyan, D., Kendall, W.S., Mecke, J. (1995): Stochastic Geometry and its Applications, Wiley
- Schmidt, V. (2008): Räumliche Statistik (Vorlesungsskript)
- Thiedmann, R. (2003): Einführung in Punktprozesse und Ihre Charakteristiken

Schlusswort

Vielen Dank für Ihre Aufmerksamkeit!