

**FERNANDO REIS DA SILVA**

**ANÁLISE NO TEMPO E NO ESPAÇO EM MODELOS DE DINÂMICA  
DE POPULAÇÕES**

Monografia de graduação  
apresentada ao Departamento  
de Ciência da Computação da  
Universidade Federal de Lavras  
como parte das exigências do  
Curso de Ciência da  
Computação para obtenção do  
título de Bacharel em Ciência  
da Computação.

LAVRAS  
MINAS GERAIS – BRASIL  
2009

**FERNANDO REIS DA SILVA**

**ANÁLISE NO TEMPO E NO ESPAÇO EM MODELOS DE DINÂMICA  
DE POPULAÇÕES**

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Orientador

Prof. Dr. Luiz Henrique Andrade Correia

Prof. Dr. João Domingos Scalon

LAVRAS  
MINAS GERAIS – BRASIL  
2009

# ANÁLISE NO TEMPO E NO ESPAÇO EM MODELOS DE DINÂMICA DE POPULAÇÕES

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Aprovada em \_\_\_ de \_\_\_\_\_ de \_\_\_\_\_

---

Prof. Dr. André Luiz Zambalde

---

Prof. Dr. Rêmulo Maia Alves

---

Prof. Dr. Luiz Henrique Andrade Correia  
(Orientador)

---

Prof. Dr. João Domingos Scalon  
(Orientador)

LAVRAS  
MINAS GERAIS – BRASIL  
2009

# DEDICATÓRIA

Dedico este trabalho aos meus pais, Nina e Marcos  
Aos meus irmãos e amigos  
E a todos os meus professores.

## **AGRADECIMENTOS**

Agradeço a David Carslake pela paciência e pela disposição em colaborar para a realização deste trabalho ao tirar minhas dúvidas.  
Também agradeço a FAPEMIG pelo apoio e patrocínio.

## **ANÁLISE NO TEMPO E NO ESPAÇO EM MODELOS DE DINÂMICA DE POPULAÇÕES**

### **Resumo**

Este trabalho propõe analisar processos pontuais descontínuos no espaço e no tempo em modelos de dinâmica de populações. É usado o software Presa-predador para simular e gerar configurações espaciais dos indivíduos ao longo do tempo. Assim, é de especial interesse entender o comportamento das configurações espaciais dos indivíduos no tempo. Através da implementação de técnicas estatísticas baseadas na função K é estimada a interação espaço-tempo. Serão usados o software R e programação no ambiente Scilab para detectar a presença de conglomerados (agrupamentos), bem como regularidade (distribuição homogênea) e aleatoriedade (distribuição heterogênea) desses indivíduos. Conclui-se que o método proposto apresenta resultados satisfatórios para o modelo de dinâmica de populações.

**Palavras-chave:** Presa-predador, Função K

## **ANALYSIS IN TIME AND SPACE IN POPULATION DYNAMICS MODELS**

### **Abstract**

This work proposes to analysis point patterns discontinuous in time and space in models of population dynamics. Presa-predador software will be used to simulate and generate spatial configurations of individuals over time. Therefore, it is of particular interest to understand the behavior of the spatial configurations of individuals over time. Through the implementation of statistic techniques based on the K function to estimate the time-space interaction of individuals. The R software and programming in Scilab environment will be used to detect the presence of clusters (groups), and regular (homogeneous distribution) and random (unequal distribution) of these individuals. The analysis concluded that the proposed method provides satisfactory results for the model of population dynamics.

**Keywords:** Presa-predador, K function

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
1.1	Objetivos	4
1.2	Definição do Problema	4
1.3	Motivação	4
1.4	Organização do trabalho	5
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>7</b>
2.1	Simulação de Sistemas	7
2.1.1	Dinâmica de Populações	7
2.1.2	Dinâmica de Populações no Modelo Tipo Presa-Predador	9
2.2	Estatística Espacial	10
2.2.1	Estimador de Intensidade	11
2.2.2	A Função K	13
2.2.3	Análise da Configuração Espacial	15
2.2.4	Análise Espaço-temporal	18
2.3	Tecnologias Computacionais	20
2.3.1	O Programa R	21
2.3.2	Spatstat	22
2.3.3	Scilab	22
<b>3</b>	<b>METODOLOGIA</b>	<b>25</b>
3.1	Procedimentos Metodológicos	25
3.1.1	Análise Individual dos Padrões de Pontos	26
3.1.2	Análise Espaço-temporal	28
3.1.2.1	Coleção de Dados	29
3.1.2.2	A Função de Análise Espaço-temporal	30
3.2	Comparação dos Métodos	31
<b>4</b>	<b>MODELAGEM E DESENVOLVIMENTO DAS FUNÇÕES DE ANÁLISE</b>	<b>33</b>
4.1	Modelo Presa-Predador	33
4.2	Coleção de dados	34
4.2.1	Função de Extração	34

4.2.2	Função Para Agrupar os Pontos onde Houve Captura pelas Armadilhas .....	36
4.3	A função K no R.....	37
4.4	A Função KF .....	37
4.4.1	Algoritmo da função KF.....	39
<b>5</b>	<b>RESULTADOS E DISCUSSÃO.....</b>	<b>41</b>
5.1	Análise espacial.....	41
5.2	Análise espaço-temporal .....	48
5.3	Comparação dos Métodos .....	50
<b>6</b>	<b>APLICAÇÕES EM REDES SEM FIO.....</b>	<b>53</b>
6.1	Redes de sensores sem fio (RSSF).....	53
6.1.1	Simulação em Uma RSSF .....	54
6.2	Redes de Telefonia Móvel.....	65
6.2.1	Simulação de uma rede de celular .....	66
6.3	Conclusão .....	68
<b>7</b>	<b>CONCLUSÕES .....</b>	<b>69</b>
7.1	Conclusão .....	69
7.1.1	Função K no R.....	69
7.1.2	A função KF .....	70
7.2	Propostas Futuras .....	70
<b>8</b>	<b>REFERENCIAL BIBLIOGRÁFICO .....</b>	<b>73</b>
<b>9</b>	<b>ANEXOS.....</b>	<b>77</b>
9.1	Spatstat .....	77
9.1.1	Instalando o Spatstat.....	77
9.1.2	Carregando o Spatstat.....	79
9.1.3	Carregando os dados .....	80
9.1.3.1	Carregando os dados a partir de um diretório .....	80
9.1.4	Criando uma janela de observação bi-dimensional.....	80
9.1.5	Transformando as coordenadas em padrão de pontos (point patterns).....	81
9.1.6	A função K .....	82



9.2	Implementação da função KF no Scilab.....	83
9.2.1	Rodando a função KF.....	87
9.2.2	.....	89
9.3	Slides .....	94

# LISTA DE FIGURAS

Figura 2 .....	14
Figura 3.1 .....	27
Figura 3.2 .....	28
Figura 3.3 .....	29
Figura 4.1 .....	37
Figura 4.2 .....	40
Figura 5.1 .....	42
Figura 5.2 .....	43
Figura 5.3 .....	44
Figura 5.4 .....	45
Figura 5.5 .....	46
Figura 5.6 .....	47
Figura 5.7 .....	48

Figura 5.8 .....	51
Figura 5.9 .....	52
Figura 6.1 .....	55
Figura 6.2 .....	56
Figura 6.3 .....	57
Figura 6.4 .....	58
Figura 6.5 .....	59
Figura 6.6 .....	60
Figura 6.7 .....	61
Figura 6.8 .....	62
Figura 6.9 .....	63
Figura 6.10 .....	64

Figura 5.9 .....66

Figura 6.11 .....67

# 1 INTRODUÇÃO

As populações biológicas são coleções de indivíduos de uma mesma espécie, que vivem temporariamente juntos em um mesmo hábitat, essa relação é discreta, ou seja não apresenta continuidade. Não é contínua no tempo e no espaço.

O estudo de populações biológicas envolve, frequentemente, o monitoramento do seu tamanho e da distribuição de seus indivíduos em uma determinada área. O tamanho e a distribuição espacial de uma população é um balanço numérico momentâneo entre dois conjuntos dinâmicos de fatores: os ganhos (nascimentos e imigrações) e as perdas (mortes e emigrações) de indivíduos.

Costa (2004) afirma que monitorar o tamanho e a distribuição espacial de populações naturais é um empreendimento com notórias e variadas repercussões práticas, como nos programas de controle de pragas agrícolas e vetores de doenças e, mais recentemente, no manejo de unidades de conservação. Costa (2004) observa ainda, que exceto em casos extremos, como em agroecossistemas super-simplificados, não é possível monitorar as flutuações numéricas e espaciais de todas as populações que convivem em um mesmo habitat. Neste cenário, os usos de simulação computacional, em conjunto com a modelagem matemática e a análise estatística, tornam-se eficazes como forma de representar os fenômenos e as interações entre diferentes indivíduos e sistemas diante das modificações realizadas em suas variáveis e parâmetros. A finalidade principal desse procedimento não é a obtenção exata da dinâmica populacional, mas sim uma aproximação satisfatória que permita a compreensão dessa dinâmica, com uma dada precisão, dentro de alguns limites aceitáveis, previamente estabelecidos. Além disso, através dessa modelagem, os

experimentos podem ser reproduzidos, verificados e comprovados por qualquer pesquisador.

Não é por acaso, portanto, que a modelagem matemática da dinâmica populacional é largamente reconhecida como uma ferramenta extremamente útil na investigação de muitas características importantes na organização de indivíduos na natureza (GUTIERREZ, 1996). Nos últimos anos, observa-se um crescimento no número de trabalhos que tentam entender a distribuição espacial e a interação local na caracterização da dinâmica ao longo do tempo de espécies competidoras em sistemas biológicos (Aguilar et al., 2004; King et al., 2003; Sherratt, 1996; Tilman et al., 1997; Wilson et al., 1993). Esses trabalhos tentam tornar mais realísticos os modelos pioneiros do tipo presa-predador propostos por Lotka (1920) e Volterra (1931). Dentro deste contexto, algumas estratégias para análise e simulação desses modelos incluem *Individual Based Models* (IBM) usando autômato celular (Carvalho et al., 2004, 2007; Carneiro et al. 2007; Sherratt, 1996) e IBM não usando autômato celular (Aguilar et al., 2004).

Neste trabalho, é utilizado o modelo IBM com autômato celular (*lattice*<sup>1</sup>) proposto por Carneiro et al. (2007) para gerar configurações espaciais no tempo, pois o mesmo permite a inclusão de diversas variáveis que são observadas na natureza sem, entretanto, aumentar exponencialmente o trabalho computacional. Um dos principais objetivos do modelo proposto por Carneiro et al. (2007) e de outros modelos tais como os propostos por Carvalho et al. (2004, 2007) é entender o comportamento das configurações espaciais dos indivíduos ao longo do tempo. Assim, é de especial interesse detectar a presença de conglomerados (agrupamento) de indivíduos no espaço e no tempo, bem como a regularidade (distribuição homogênea) e a aleatoriedade (distribuição heterogênea) desses indivíduos no espaço e no tempo.

---

<sup>1</sup> Grade cruzada, rede, malha

Apesar de análises espaço-temporais terem sido objeto de intensa pesquisa em estatística nos últimos anos (Bailey et al., 1995), pode-se verificar que a caracterização de conglomerados no espaço e tempo nos modelos IBM ainda é feita visualmente, sem o uso computacional e de estatísticas apropriadas para este fim (CARVALHO et al., 2004, 2007; CARNEIRO et al. 2007). Neste trabalho, estamos interessados na posição (localização) dos indivíduos das espécies em uma determinada região e, portanto, nos concentraremos nos aspectos metodológicos da área de estatística espacial denominada de processos pontuais.

Diggle et al. (1995) propõe utilizar a função K para analisar interação espaço-tempo em processos pontuais. O método permite a estimação da interação espaço-tempo para diferentes escalas. A aplicação direta desse método é inapropriada para a análise de agrupamentos em modelos de dinâmica populacional, uma vez que o método é adequado apenas para dados que são localizados de modo contínuo no espaço e tempo. Indivíduos nos modelos de dinâmica populacional, em geral, são localizados em uma grade descontínua (*lattice*) e são amostrados em tempos discretos. Carslake et al. (2005) propõe uma adaptação do método da função K para atender essas particularidades em um problema específico de risco de infecção em bovinos. Entretanto, esta adaptação não foi testada em diferentes tipos de interação espaço-tempo que ocorrem com frequência nos modelos de dinâmica populacional. Além disso, pode-se pensar em usar outras funções que também são capazes de descrever a configuração espacial dos pontos, tais como as funções F e G descritas em Diggle (2002).

## **1.1 Objetivos**

Este trabalho tem como objetivo geral a análise de processos pontuais descontínuos no espaço e no tempo em modelos de dinâmica de populações. Ou seja, analisar padrões de pontos no espaço com técnicas estatísticas baseadas na adaptação da função K proposta por Carslake et. al. (2005). Estes métodos serão desenvolvidos utilizando o software R (R Development Core Team, 2006) e o ambiente Scilab (Scilab – Digitio).

E como objetivos específicos a implementação dos métodos estatísticos para as análises espaciais e temporais e a comparação entre os métodos implementados com os métodos existentes baseados na função K.

## **1.2 Definição do Problema**

O problema consiste na falta de métodos computacionais para a análise de conglomerados no tempo e espaço. A caracterização de conglomerados no espaço e no tempo nos modelos IBM ainda é feita visualmente, sem o uso computacional e de estatísticas apropriadas para este fim (Carvalho et al. 2004).

## **1.3 Motivação**

Na literatura recente não existem métodos computacionais ou metodologias para análise de conglomerados no tempo e espaço em modelos IBM.

Carslake et al. (2005) propõe uma adaptação da função K para caracterização de conglomerados no espaço e no tempo, porém essa adaptação não foi testada para diferentes escalas de interação. Neste



trabalho propomos a implementação computacional deste método proposto por Carslake et al. (2005) e a extensão do método para diferentes escalas de interação espaço-tempo que ocorrem com frequência nos modelos de dinâmica de populacional.

## **1.4 Organização do trabalho**

No capítulo 2 é apresentado um referencial teórico que descreve a simulação de sistemas utilizada neste trabalho, a área da estatística denominada estatística espacial que é a base matemática para os métodos propostos e as tecnologias computacionais que foram indispensáveis para a implementação.

A metodologia que descreve o tipo de trabalho realizado e os procedimentos metodológicos é apresentada/mostrada no capítulo 3.

O capítulo 4 descreve como foi feita a implementação dos métodos propostos.

Todos resultados gerados pela implementação como também a discussão em cima destes resultados são apresentados no capítulo 5.

Com o capítulo 6 fecha-se o trabalho com as conclusões e as propostas futuras.



## **2 REFERENCIAL TEÓRICO**

### **2.1 Simulação de Sistemas**

Simulação é um processo no qual se cria um modelo de um sistema real e se faz experimentos a fim de entender seu comportamento ou avaliar estratégias de sua operação. Ou seja, é a reprodução do funcionamento de um sistema ou de um processo do mundo real durante determinado período de tempo. A simulação possibilita tirar conclusões sobre novos sistemas sem precisar construí-los, ou fazer alterações em sistemas já existentes (KATO et al. 1997). A crescente complexidade dos problemas que encontramos e a maior disponibilidade de recursos computacionais são dois fatores que têm contribuído para o crescimento do uso de técnicas de simulação. A simulação computacional se torna presente neste trabalho em um modelo proposto por Carneiro et al. (2007) o qual será descrito nas seções seguintes.

#### **2.1.1 Dinâmica de Populações**

Embora este trabalho não esteja diretamente relacionado a populações biológicas, o conceito de dinâmica de populações será usado para a realização do mesmo devido a modificação proposta por Carslake et al. (2005) ter sido implementada especificamente para o caso de estudo da interação espacial em dinâmica de populações.

O estudo de populações biológicas envolve, frequentemente, o monitoramento do seu tamanho e da distribuição de seus indivíduos em uma determinada área. O tamanho e a distribuição espacial de uma população é um

balanço numérico momentâneo entre dois conjuntos dinâmicos de fatores: os ganhos (nascimentos e imigrações) e as perdas (mortes e emigrações) de indivíduos.

Costa (2004) afirma que monitorar o tamanho e a distribuição espacial de populações naturais é um empreendimento com notórias e variadas repercussões práticas, como nos programas de controle de pragas agrícolas e vetores de doenças e, mais recentemente, no manejo de unidades de conservação. Costa (2004) observa ainda, que exceto em casos extremos, como em agroecossistemas super-simplificados, não é possível monitorar as flutuações numéricas e espaciais de todas as populações que convivem em um mesmo habitat. Neste cenário, os usos de simulação computacional, em conjunto com a modelagem matemática e a análise estatística, tornam-se eficazes como forma de representar os fenômenos e as interações entre diferentes indivíduos e sistemas diante das modificações realizadas em suas variáveis e parâmetros.

Não é por acaso, portanto, que a modelagem matemática da dinâmica populacional é largamente reconhecida como uma ferramenta extremamente útil na investigação de muitas características importantes na organização de indivíduos na natureza (Gutierrez, 1996). Nos últimos anos, observa-se um crescimento no número de trabalhos que tentam entender a distribuição espacial e a interação local na caracterização da dinâmica ao longo do tempo de espécies competidoras em sistemas biológicos (Aguiar et al., 2004; King et al., 2003; Sherratt, 1996; Tilman et al., 1997; Wilson et al., 1993).

## 2.1.2 Dinâmica de Populações no Modelo Tipo Presa-Predador

Os primeiros modelos para o estudo da dinâmica de populações foram formulados em termos de equações diferenciais (Lotka, 1920; Volterra, 1931). Esta abordagem permitiu que vários métodos analíticos desenvolvidos em outras áreas (ex. Física e Engenharia) pudessem ser aplicados em dinâmica de populações. Com o surgimento de computadores velozes e baratos é possível construir modelos mais sofisticados que, normalmente, resultam em equações com soluções custosas (Gutierrez et al., 1996). Algumas estratégias para análise e simulação desses modelos incluem Individual Based Models (IBM) com autômato celular (Carvalho et al., 2004, 2007; Carneiro et al. 2007; Sherratt, 1996), IBM não usando autônomo celular (Aguilar et al., 2004) e métodos de elementos finitos (KING et al., 2001).

Neste projeto, é utilizado o modelo autômato celular IBM (*lattice*) proposto por (Carneiro et al., 2007). Um dos principais objetivos do modelo proposto por Carneiro et al. (2007) é entender o comportamento espacial dos indivíduos ao longo do tempo. Assim, é de especial interesse detectar a presença de agrupamentos espaciais dos indivíduos no espaço e no tempo.

Outros modelos propostos também buscam entender as interações espaço-temporais dos indivíduos (Wilson et al., 1993, Carvalho et al., 2004, 2007), entretanto, todos falham em apresentar uma metodologia estatística capaz de estimar a interação espaço-temporal dos indivíduos.

Neste trabalho, é de interesse a posição (localização) dos indivíduos das espécies em uma determinada região e, portanto, concentra-se nos aspectos metodológicos da área de estatística espacial denominada de processos pontuais.

## 2.2 Estatística Espacial

A análise espacial pode ser definida como uma técnica que busca descrever os padrões existentes nos dados espaciais e estabelecer, preferencialmente, de forma quantitativa, os relacionamentos entre as diversas variáveis geográficas. A taxonomia mais utilizada para descrever o problema da análise espacial considera três tipos de dados conforme pode ser visto em Bailey et al. (1995), Cressie (1991) e Camara et al. (2002): eventos de padrões espaciais, superfícies contínuas e áreas com contagens.

*Eventos de padrões espaciais:* são fenômenos expressos através de ocorrências identificadas como pontos localizados no espaço e denominados processos pontuais. São exemplos desse tipo de dados a localização da ocorrência de casos de doenças e localização de indivíduos de uma determinada espécie.

*Superfícies contínuas (geoestatística):* são fenômenos que se distribuem continuamente em uma região. Usualmente, esse tipo de dados é resultante de levantamento de recursos naturais e que incluem mapas geológicos, topográficos e ecológicos. Um exemplo desse tipo de dados são medidas da concentração de um elemento químico no solo.

*Áreas com contagens:* são fenômenos associados aos dados de levantamentos populacionais, como censos, e que originariamente referem a indivíduos localizados em pontos específicos no espaço. Normalmente, esses dados são agregados em unidades de análises, usualmente delimitadas por polígonos fechados tais como setores censitários, municípios e microrregiões.

Esses tipos de dados espaciais podem ser observados em um período de tempo contínuo ou discreto. Assim, o objetivo do uso da estatística é apresentar

diversas formas de tratamento dos dados observados no espaço e tempo que apresentam estrutura de correlação nessas dimensões. Uma revisão sobre as diversas técnicas de tratamento desse tipo de dados, sob o ponto de vista clássico pode ser vista em Cressie (1991).

## 2.2.1 Estimador de Intensidade

Uma alternativa simples para analisar o comportamento de padrões de pontos é a estimar a intensidade pontual do processo em toda a região de estudo. Para isto, pode-se ajustar uma função bi-dimensional sobre os eventos considerados, compondo uma superfície cujo valor será proporcional à intensidade de amostras por unidade de área. Esta função realiza uma contagem de todos os pontos dentro de uma região de influência, ponderando-os pela distância de cada um à localização de interesse.

A partir dos conceitos apresentados, suponha que  $u_1, \dots, u_n$  são localizações de  $n$  eventos observados em uma região  $A$  e que  $u$  represente uma localização genérica cujo valor queremos estimar. O estimador de intensidade é computado a partir dos  $m$  eventos  $\{u_i, \dots, u_{i+m-1}\}$  contidos num raio de tamanho  $\tau$  em torno de  $u$  e da distância  $d$  entre a posição e a  $i$ -ésima amostra, a partir de funções cuja forma geral é mostrada na equação (2.1):

$$\hat{\lambda}_\tau(u) = \frac{1}{\tau^2} \sum_{i=1}^n k\left(\frac{d(u_i, u)}{\tau}\right), d(u_i, u) \leq \tau \quad (2.1)$$

Este estimador é chamado kernel estimator e seus parâmetros básicos são: (a) um raio de influência ( $\tau > 0$ ) que define a vizinhança do ponto a ser interpolado e controla o "alisamento" da superfície gerada; (b) uma função de estimação com propriedades de suavização do fenômeno. O raio de influência

define a área centrada no ponto de estimação  $u$  que indica quantos eventos  $u_i$  contribuem para a estimativa da função intensidade  $\lambda$ . Um raio muito pequeno irá gerar uma superfície muito descontínua; se for grande demais, a superfície poderá ficar muito amaciada. No caso da função de interpolação  $k()$ , é comum usar funções de terceira ou quarta ordem, como a mostrada pela equação (2.2), ou o Kernel gaussiano, dado pela equação (2.3):

$$k(h) = \frac{3}{\pi} (1 - h^2) \quad (2.2)$$

$$k(h) = \frac{1}{2\pi\tau} \exp\left(-\frac{h^2}{2\tau^2}\right) \quad (2.3)$$

Nestes estimadores,  $h$  representa a distância entre a localização em que desejamos calcular a função e o evento observado. Com o uso desta função de quarta ordem (equação 2.2), o estimador de intensidade pode ser expresso pela equação (2.4):

$$\hat{\lambda}_\tau(u) = \sum h_i \frac{3}{\pi\tau^2} \left(1 - \frac{h_i^2}{\tau^2}\right)^2 \quad (2.4)$$

O estimador de intensidade é muito útil para nos fornecer uma visão geral da distribuição de primeira ordem dos eventos. Trata-se de um indicador de fácil uso e interpretação. Neste trabalho estamos interessados nos efeitos de segunda ordem (ou segundo momento), em distribuição de pontos, que são caracterizados pela função  $K$ .



## 2.2.2 A Função K

A função K ou "medida reduzida de segundo momento", introduzida originalmente em Ripley (1977), é um método que proporciona uma descrição efetiva da dependência espacial dentro de um padrão pontual em determinada área, ou seja, verifica se os eventos estão distribuídos ao acaso, se eles se encontram em um padrão de regularidade, ou se existem conglomerados específicos, sendo tal análise possível para uma vasta gama de escalas espaciais. A função K é definida como o número esperado de eventos dentro de uma distância t de um evento arbitrário dividido pela intensidade de pontos na região. A sua estimação baseia-se na variação do número médio de outros eventos em torno de um evento qualquer, à medida que variamos o raio de consideração.

A popularidade da função K encontra-se primeiramente na facilidade de sua estimação se comparada com outras medidas de segunda ordem para processos pontuais como a função intensidade de segunda ordem. Ela apresenta uma propriedade muito útil, a invariância sob redução aleatória, o que permite que o tamanho de amostra não interfira na sua estimação. Sob a hipótese de Completa Aleatoriedade Espacial (CAE), que reflete uma disposição casual dos eventos, sem interações entre eles, a função K é expressa pela equação (2.5):

$$k(t) = \pi t^2 \quad (2.5)$$

O teste de Monte Carlo de CAE a partir da função K utiliza-se da geração de diversas simulações sob CAE, estimando a função K para cada realização, Diggle (2003). Como se está interessado em verificar o comportamento da função K através de um intervalo de distâncias, constrói-se

envelopes em volta do valor esperado sob hipótese nula, Diggle (2003) ou Waller and Gotway (2004).

Dado um raio  $t$  a função  $K$  é definida pela equação 2.6:

$$K(t) = \frac{A}{n^2} \sum_i \sum_j \delta_{ij}(d_{ij}) \quad (2.6)$$

Para  $i \neq j$ , sendo  $A$  à área amostrada,  $n$  o total de pontos,  $\delta_{ij}$  uma função indicadora que devolve 1 se a distância  $d_{ij}$  entre os pontos  $i$  e  $j$  é menor que  $t$  e 0 caso contrario.

Em outras palavras, a função  $K$  conta o número de ocorrências do evento  $j$  em um círculo de raio  $t$  para cada centro  $i$  conforme ilustrado na figura 3.

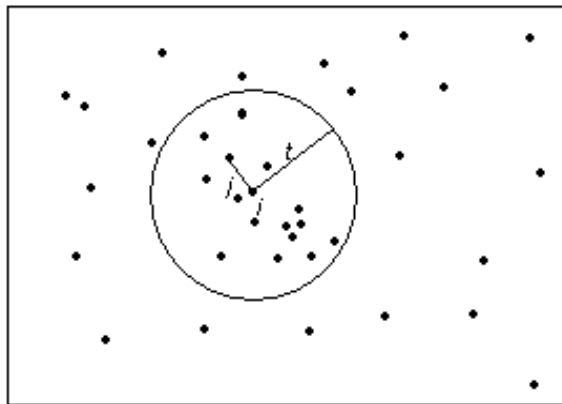


Figura 2. Superfície sendo analisada pela função  $K$ .

Na seção seguinte será apresentado alguns conceitos referentes a análise da configuração espacial.

### 2.2.3 Análise da Configuração Espacial

Os processo pontuais diferem dos dados de superfície e de área pelo fato de que o tempo e local de observação não vêm associados à realização de uma variável aleatória (variável resposta). O interesse aqui é a informação a respeito de quando e/ou onde ocorreram determinados eventos. Particularmente, estamos interessados em analisar o padrão espaço temporal da incidência de um indivíduo (usuário) em um determinado local, e para isso analisar os dados provenientes do local e da data da ocorrência. Mais especificamente, estamos interessados em responder se o local onde um indivíduo aparece é independente de quando ele aparece. Para conduzir tal análise, primeiramente, temos que caracterizar a distribuição espacial da ocorrência dos indivíduos em termos de estimação de propriedades de primeira e segunda ordem (Cressie, 1991). Para análise das propriedades de primeira ordem utiliza-se algum estimador de intensidade pontual do processo (ex. kernel) para toda a região (Scalon et al., 2003). As técnicas mais utilizadas para estimar as propriedades de segunda ordem são baseadas nas funções  $F$ ,  $G$  e  $K$ .

A função  $F$ , introduzida por Ripley (1977), é a função de distribuição acumulada da distância entre um ponto aleatoriamente escolhido e o evento mais próximo. O mais simples estimador para a função  $F$  pode ser obtido pela equação (2.7).

$$\hat{F}(x) = m^{-1} \sum_{i=1}^m I_x(x_i) \quad (2.7)$$

onde  $m$  é o número de pontos aleatórios no mapa,  $x_i$  representa a distância do  $i$ th ponto aleatório para o mais próximo dos  $n$  eventos no mapa analisado e  $I_x(x_i)$  é uma função indicadora igual a 1 quando  $x_i$  é menor ou igual a  $x$ , e 0 caso contrário.

Entretanto, o estimador apresentado na equação (2.7) é viciado, uma vez que, para uma mesma distância, pontos aleatórios localizados nas bordas do mapa terão uma probabilidade menor de ter um evento próximo do que os pontos localizados no centro do mapa. Este efeito ocorre porque, na prática, trabalha-se com regiões de área finita enquanto teoricamente a área é considerada infinita. Assim, a distância entre um ponto aleatório e um evento que se encontra fora do mapa é ignorada, mesmo que esta esteja menor que qualquer distância entre um ponto aleatório e um evento localizados dentro do mapa. Um estimador não viciado que corrige o efeito de bordas é dado pela equação (2.8).

$$\hat{F}(x) = \frac{\sum_{i=1}^n I_x(x_i, r_i)}{\sum_{i=1}^n I_x(r_i)} x > 0, \quad (2.8)$$

onde  $r_i$  é a distância de um evento até o ponto mais próximo na borda do mapa e  $I_x(x_i, r_i)$  é uma função indicadora igual a 1, quando  $x_i$  é menor ou igual a  $x$  e  $r_i$  é maior ou igual a  $x$  e 0, caso contrário.

Pode-se demonstrar que a função  $F$  teórica para um processo de Poisson homogêneo, com área infinita, é dada pela equação (2.9)

$$F(x) = P\{N(\pi x^2) > 0\} = 1 - \exp(-\lambda \pi x^2), \quad x > 0. \quad (2.9)$$

A função  $G$  é a função de distribuição acumulada da distância entre um evento e seu vizinho mais próximo ( $y$ ) (Ripley, 1977). Estimadores (natural e corrige o efeito de borda) da função  $G$  e a função  $G$  teórica para um processo de Poisson homogêneo com área infinita e um estimador são obtidas por equações análogas as da função  $F$ . Para tal, basta apenas trocar a distância  $x$  pela distância  $y$  nas equações de (2.7) a (2.9).

Diggle (2002) mostra que a função  $F$  apresenta grande poder para detectar agrupamentos de eventos, enquanto a função  $G$  é mais poderosa para detectar eventos regularmente distribuídos. Assim, recomenda-se usar uma função que seja igualmente poderosa para detectar os dois padrões em diferentes escalas, como a função  $K$  proposta por Ripley (1977). A função  $K$  é definida como.

A função  $K$  teórica para um processo de Poisson homogêneo com área infinita é dada pela equação  $K(t) = \pi t^2$ . O mais simples e natural estimador de  $K(t)$  que corrige o efeito de bordas é dado por (2.10),

$$\hat{K}(t) = \frac{|A|}{n^2} \sum_{i=1}^n \sum_{j=1}^n \frac{I_t(u_{ij})}{w_{ij}}, \quad (2.10)$$

onde  $u_{ij}$  é a distância entre os eventos  $i$  e  $j$  ( $i \neq j$ ),  $I_t(u_{ij})$  é uma função indicadora igual a 1 quando  $u_{ij}$  é menor que a distância  $t$  e 0, caso contrário,  $n$  é

o número de evento no mapa analisado com área  $|A|$  e  $w_{ij}$  é um fator de correção que representa a proporção da circunferência ao redor de um evento  $i$ , passando sobre o evento  $j$  que esta dentro de  $|A|$ .

As funções empíricas apresentadas anteriormente podem ser utilizadas de duas formas (gráfico e teste formal) para testar a hipótese nula de que os eventos seguem uma distribuição de Poisson homogênea, ou seja, que os eventos estão aleatoriamente distribuídos no mapa. Detalhes sobre estas metodologias podem ser vistas em Scalon et al. (2003) e Diggle (2002). Estas técnicas serão aplicadas para cada uma das espécies (presa e predador) considerando, primeiramente, apenas a distribuição espacial de cada espécie, em seguida consideraremos as configurações espaciais e temporais.

## 2.2.4 Análise Espaço-temporal

Após analisar, individualmente, as configurações espaciais através da função  $K$ , será realizada a análise espaço-temporal. Para tal vamos utilizar a estatística para agrupamentos no espaço e tempo, utilizando a função  $K$ , proposta por Carlslake et al. (2005). A estatística  $K(s,t)$ , para cada combinação de  $s$  e  $t$  (escalas espacial e temporal) é dada pela equação (2.11)

$$K(s,t) = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n I_s(d_{ij}) I_t(t_{ij}) w_{ij} v_{ij} R_s R_t \quad (2.11)$$

onde  $d_{ij}$  é a distância espacial entre os eventos  $i$  e  $j$  ( $i \neq j$ ),  $t_{ij}$  é a distância temporal entre os eventos  $i$  e  $j$  ( $i \neq j$ ),  $I_s$  é uma função indicadora igual a 1 quando  $d_{ij}$  é menor que a distância  $s$  e 0, caso contrário,  $I_t$  é uma função indicadora igual a 1 quando  $t_{ij}$  é menor que a distância  $t$  e 0, caso contrário,  $w_{ij}$  é um fator de correção definido anteriormente na função  $K$ ,  $v_{ij}$  é um fator de correção para o efeito de borda do tempo,  $R_s$  é o número de pontos na grade de estudo,  $R_t$  é o número de pontos no tempo,  $n$  é o número total de eventos no mapa analisado e  $n(n-1)$  é o número de pares disjuntos de eventos.

Estatísticas de agrupamento individuais para espaço e tempo equações (2.12) e (2.13) podem ser obtidas diretamente da equação (2.11), ou seja,

$$K(s) = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n I_s(d_{ij}) w_{ij} R_s \quad (2.12)$$

$$K(t) = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n I_t(t_{ij}) v_{ij} R_t \quad (2.13)$$

Seguindo Diggle et al. (1995), pode-se observar que se os processos temporal e espacial operam independentemente (ausência de interação espaço-temporal), então  $K(s, t) = K(s)K(t)$ . Se  $K(s, t) > K(s)K(t)$  então existe uma interação espaço tempo positiva. Carlslake et al. (2005) propõe uma maneira

intuitiva e fácil de quantificar, separadamente, o grau de agrupamento espacial e temporal através das expressões das equações 2.14 e 2.15.

$$K^*(s) = [k(s) \textit{ observado} - k(s) \textit{ esperado}] / k(s) \textit{ esperado}$$

(2.14)

$$K^*(t) = [k(t) \textit{ observado} - k(t) \textit{ esperado}] / k(t) \textit{ esperado}$$

(2.15)

Para a implementação das estatísticas descritas usaremos alguns dispositivos computacionais existentes para a simulação e análise como também a implementação da estatística de configuração espaço-temporal.

## 2.3 Tecnologias Computacionais

Os Softwares utilizados no desenvolvimento deste trabalho serão descritos nessa seção. A escolha destes Softwares estão diretamente relacionada às ferramentas oferecidas pelos mesmos como os pacotes de funções disponíveis no software R (R Development Core Team 2006) e a facilidade de programar encontrada no ambiente Scilab (SCILAB – DIGITIO). Todas as ferramentas envolvidas na implementação dos métodos estatísticos serão introduzidas nesta seção.



### 2.3.1 O Programa R

O que torna a escolha do R (R Development Core Team , 2006) para a implementação da função K é o grande número de pacote de funções para estatística espacial como o Spatstat, que foi o utilizado neste trabalho para analisar, individualmente, as configurações espaciais.

Tais pacotes escritos para o programa R acrescentam algoritmos avançados, gráficos coloridos e texturizados e técnicas de mineração para vasculhar bancos de dados mais a fundo, R Development Core Team (2006).

A distribuição do R é gratuita. Ele vem na forma de programa de fonte aberta. A linguagem R é semelhante a outras linguagens de programação, como C++ e Java, o que ajuda a o usuário executar uma ampla variedade de tarefas de computação ao lhes fornecer acesso a diversos comandos inclusive orientação a objetos. Para aplicações estatísticas o R é especialmente útil porque contém diversos mecanismos incorporados para a organização de dados, execução de cálculos sobre informações e criação de representações gráficas de conjuntos de dados.

O R pode ser melhorado, o que explica a sua grande aceitação no meio científico. O seu código básico pode ser melhorado ou pode criar outras variações para tarefas específicas. Os pacotes do R acrescentam algoritmos avançados, gráficos coloridos e texturizados e técnicas de mineração para vasculhar bancos de dados, *R Development Core Team (2006)*. Neste trabalho usaremos o pacote Spatstat que contém ferramentas para estatística espacial.

### 2.3.2 Spatstat

Spatstat, Baddeley (2005), é um dos vários pacotes da linguagem R para analisar padrões de pontos em duas dimensões. O pacote Spatstat é de autoria de Adrian Baddeley e Rolf Turner e possui licença pública GNU.

O Spatstat usa os recursos orientados a objeto do R (classes e métodos para facilitar, manipular, analisar os conjuntos de dados de plotagem, Baddeley (2005).

As funções disponíveis no Spatstat para tratar padrões de pontos são ideais para a implementação da função K para os padrões de pontos que serão utilizados neste trabalho.

### 2.3.3 Scilab

O Scilab é um pacote de programas científicos para computação numérica, fornecendo um ambiente aberto poderoso para aplicações científicas.

Desenvolvido desde 1990 pelos pesquisadores do INRIA (*Institut National de Recherche en Informatique et en Automatique*) e do ENPC (*École Nationale des Ponts et Chaussées*), é agora mantido e desenvolvido pelo Consorcio Scilab desde sua criação em Maio de 2003 .

O Scilab é distribuído livremente e de código aberto e atualmente está sendo usado em ambientes educacionais por todo o mundo.

O Scilab inclui centenas de funções matemáticas com a possibilidade de adicionar interativamente programas de várias linguagens ( FORTRAN, C, C++, Java ). Possui sofisticadas estruturas de dados, incluindo listas, polinômios,

sistemas lineares, um interpretador de linguagem de programação de alto nível. O usuário pode definir novos tipos de informação e operações sobre esses tipos. (Scilab – Digitio).

No próximo capítulo serão apresentadas as metodologias e os procedimentos metodológicos empregados para a implementação computacional deste trabalho.



## **3 METODOLOGIA**

Com relação a classificação de Jung (2004) e Marconi & Lakatos (2003) esta pesquisa pode ser classificada quanto a sua natureza como sendo uma pesquisa tecnológica; quanto aos seus objetivos descritiva; quanto aos procedimentos operacional e quanto ao local de realização da mesma em laboratório.

Com a implementação e análise estatística nesse trabalho pretende-se entender melhor a interação espaço-tempo em modelos de dinâmica de população bem como a interação em processos descontínuos no espaço e no tempo.

### **3.1 Procedimentos Metodológicos**

O modelo presa-predador proposto por Carneiro et. Al. (2007) gera como saída uma matriz quadrática 100 por 100 que representa a área onde a população está distribuída. Em lapsos de tempo, ou seja a cada período de tempo é gerada uma nova matriz que simula a distribuição da população de indivíduos presa e predador independentes, ou seja para cada indivíduo é gerada uma matriz de distribuição dos mesmos no espaço que representa que é entendido como as gerações destas populações. Essa matriz é preenchida por zeros e uns onde zeros representa a ausência de indivíduo e um a presença de indivíduo. As coordenadas dos pontos onde contem indivíduos (numero um na matriz) é a base de dados para os métodos propostos. A análise dos padrões associados a essa distribuição no espaço e no tempo é primordial para conduzir a análise das interações destes indivíduos bem como sua dinâmica.

Essa análise será dada por duas maneiras sendo a primeira delas o acompanhamento de indivíduos da população ou seja, para cada saída do programa presa-predador será analisada a distribuição dos indivíduos no espaço. A segunda maneira será a análise espaço-temporal proposta por Carlslake (2003), considerando varias saídas que representa a distribuição destes indivíduos em períodos de tempos diferentes.

Uma comparação entre essas duas abordagens, que serão descritas nas seções seguintes, será útil para verificar a eficácia do método proposto por Carlslake (2003).

### **3.1.1 Análise Individual dos Padrões de Pontos**

A análise individual dos padrões de pontos se dá por meio do uso da função K para a base de dados contendo a distribuição espacial destes pontos.

Esta implementação foi realizada no software R para cada saída do programa presa predador, ou seja para cada período de tempo foi rodada a função K e posteriormente analisados os resultados o qual mostra o padrão da distribuição espacial dos indivíduos. A figura 3.1 mostra o gráfico da função K para a distribuição de presa no intervalo de distribuição no tempo 1.

O gráfico da figura 3.1 mostra as curvas da função K para hipótese nula e para a K observada no tempo 1. E a figura 3.2 é o gráfico da função K para a mesma população no tempo 2.

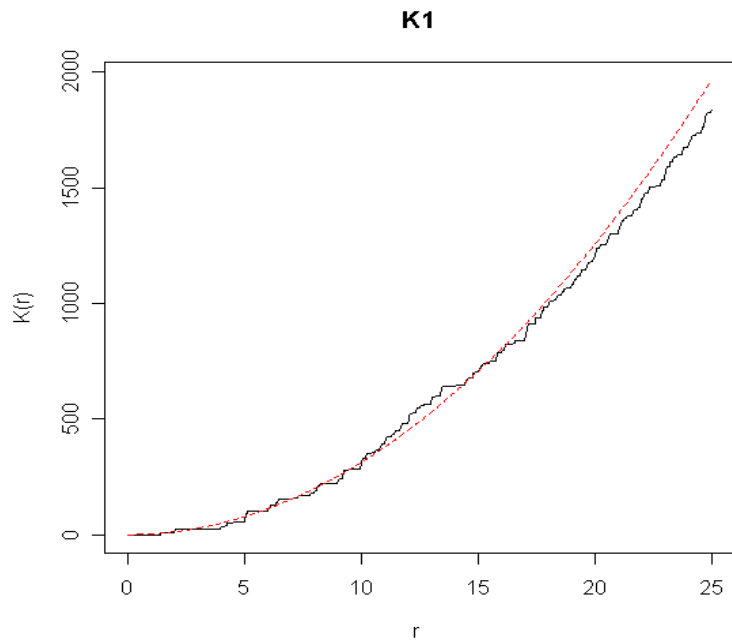


Figura 3.1 - Gráfico da função K para o tempo um.

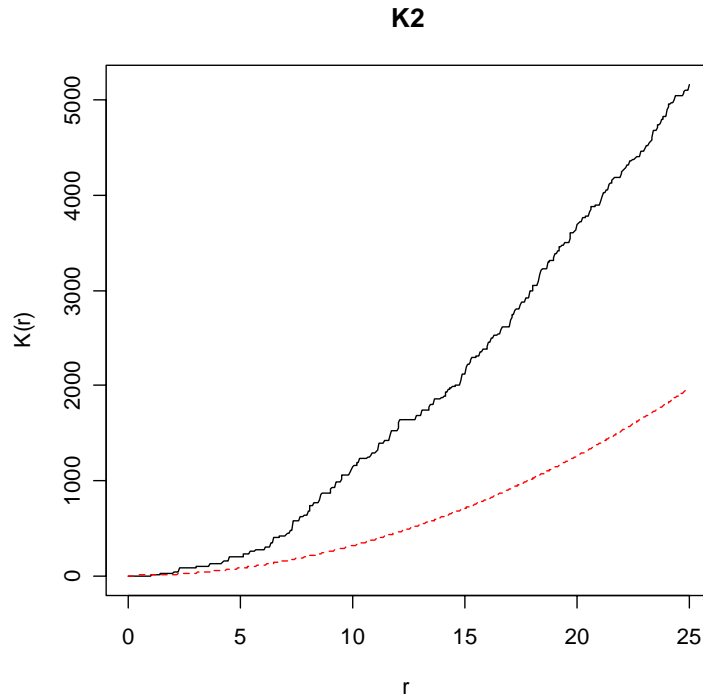


Figura 3.2 - Gráfico da função K para o tempo dois.

A figura 3.2 mostra as curvas da função K para o tempo 2, nota-se a diferença entre as curvas apresentadas na figura 3.1 e 3.2, tal diferença será comentada no decorrer deste trabalho.

### 3.1.2 Análise Espaço-temporal

Para a análise da configuração espaço-temporal da população em estudo será preciso realizar alguns procedimentos antes de entrar com os dados na função. Nessa seção será descrito os passos a serem tomados para proceder com a análise.



### 3.1.2.1 Coleção de Dados

A base de dados que representa em sua totalidade a matriz 100 por 100 gerada pelo programa presa e predador precisa passar por um tratamento, para não trabalharmos com uma base de dados demasiadamente grande o que seria computacionalmente inviável. Esse tratamento foi feito através de funções implementadas no ambiente Scilab, na implementação o intervalo de separação entre as armadilhas pode ser alterado.

Para essa redução do número de pontos a serem analisados dentro dessa área de 100 por 100 unidades de medida armadilhas serão colocadas para capturar os indivíduos, assumindo que esse procedimento não causará divergência no resultado quanto ao real número de pontos. As armadilhas serão espaçadas por 10 unidades de medida conforme pode ser visto na figura 3.3.

	10			(...)			
10							10
	10						
(...)		(...)		(...)			(...)
	10			(...)			

Figura 3.3 Representação das armadilhas na área de distribuição dos indivíduos de uma população.

Na figura 3.3 os quadrados rachurados representam os pontos da matriz que contém armadilha. A presença do indivíduo é armazenada em uma nova base de dados, que contém as coordenadas da posição da armadilha, caso seja verificada a presença do indivíduo naquela coordenada.

O programa presa-predador gera sucessivas distribuição para N intervalos de tempos. Com isso o intervalo de tempo a ser analisado deve ser definido. Definido o intervalo de tempo T para cada distribuição de 1 a T deve ser repetido o procedimento descrito acima, o qual gerará T novas bases de dados com as coordenadas dos indivíduos o intervalo de tempo em que cada nova base de dado foi gerado deve ser armazenado.

Todas essas T bases de dados devem ser agrupadas em uma única base da dados (C) que é uma matriz com três colunas sendo que as duas primeiras contém as coordenadas dos pontos (x e y) e a terceira deve conter o valor do intervalo de tempo em que os pontos foram capturados esses valores variam de 1 a T. A base de dados C será a entrada para a função que fará a análise espaço-temporal dos pontos.

### **3.1.2.2 A Função de Análise Espaço-temporal**

Neste trabalho foi necessário implementar a função de análise espaço-temporal, chamada de KF, essa implementação foi feita no ambiente Scilab pela facilidade em trabalhar com computação numérica e pela geração dos gráficos que é crucial para a análise que o trabalho propõe. Mas a implementação de KF poderia ser perfeitamente realizada com outras linguagem de programação tais como C++, Python, como também no software R.

O custo computacional que a função KF exige, é alto. Devido a esse alto custo, nesse projeto a base de dados foi restringida pelo uso das armadilhas e também pelo uso de poucos intervalos de tempo sem que com isso a eficácia de KF, na análise espaço-temporal, fosse prejudicada.

A partir da base de dados a função KF retorna como resultado o vetor  $K_s$  que contém os resultados da função K para detecção de cluster no espaço, o vetor  $K_t$  que contém os resultados da função K para detecção de cluster no tempo e  $K_{st}$  que é a função para detecção de clusters espaço-temporal. A função KF também gera um gráfico que do aumento proporcional do cluster no espaço-tempo ( $D_{0st}$ ) que é devido a interação espaço-tempo. Esse gráfico é importante pois permite a visualização dos clusters no espaço e no tempo.

A análise dos resultados gerados por KF permitirá dizer se a distribuição dos indivíduos da população, no espaço-tempo, será aleatória, regular ou se apresentará agrupamentos (clusters).

## **3.2 Comparação dos Métodos**

Para a verificação da funcionalidade da função KF os resultados obtidos por ela com os resultados da função K realizada para cada distribuição de dados individualmente. Esta análise será feita observando os gráficos e os comparando de acordo com os valores da função K de cada um e o intervalo de tempo em que foi gerado.

No capítulo seguinte será descrita a modelagem e o desenvolvimento das funções de acordo com a metodologia proposta.



## 4 MODELAGEM E DESENVOLVIMENTO DAS FUNÇÕES DE ANÁLISE

Neste capítulo serão apresentados a modelagem do programa Presa-Predador e as funções para análise espacial e espaço-temporal das distribuições de pontos gerados pelo Presa-Predador.

### 4.1 Modelo Presa-Predador

O modelo é do tipo IBM e consiste de uma população fixa de plantas, uma população de herbívoros (que se alimenta da população de plantas sendo capaz de difundir através do sistema) e uma população de predadores (que se alimenta de herbívoros para procriar e também pode se difundir através do sistema).

O tempo é discretizado em intervalos periódicos que podem ser interpretados como gerações da população. O espaço também é discretizado em uma grade regular bi-dimensional formado por células (*lattice*). Cada célula pode ser vazia, ocupada por uma presa, ocupada por um predador, ou ocupada por ambos presa e predador.

Tendo como base ações simples dos indivíduos na natureza, foram propostas algumas regras para as variáveis do autômato celular, incluindo, movimento dos indivíduos, morte natural, crescimento das plantas, agrupamento de plantas, predação e reprodução.

Os indivíduos (presa e predadores) são inicialmente distribuídos sobre a grade regular. Tendo como base as regras do autômato celular, eventos são gerados formando uma configuração espacial. A atualização das populações

ocorre no final de cada período de tempo. Indivíduos que morrem em um particular tempo  $t$  são removidos do sistema, novos indivíduos são colocados na região e a posição dos indivíduos é atualizada. Indivíduos que foram selecionados para morrer em um dado período de tempo continuaram reproduzindo naquele intervalo, sendo removidos da população no final de cada rodada.

Utilizando programas desenvolvidos em C++ por Carneiro et al. (2007), o modelo simula configurações periódicas das presas e predadores localizados em um *lattice* regular quadrado com 100 x 100 células.

## **4.2 Coleção de dados**

A coleção de dados foi retirada da matriz 100x100 gerada pelo programa presa-predador, que é a área de observação. Esses dados estão dispostos em forma de matriz 2 por 2, que representa as coordenadas  $x$  e  $y$  de cada ponto. Porém para reduzir o número de pontos para serem analisados pela função KF algumas funções foram implementadas para efetuar a redução de pontos sem alterar possíveis resultados da função. Essas funções foram a função de extração dos indivíduos capturados pelas armadilhas em cada intervalo de tempo, a função que agrupa todos os indivíduos capturados nos intervalos de tempo. Nessa seção será mostrado como foi a implementação de cada uma.

### **4.2.1 Função de Extração**

A função de extração (FE) foi desenvolvida no Scilab para extrair os pontos nas armadilhas a partir dos pontos da matriz gerada pelo presa-predador.

A função FE (Função de Extração) tem como parâmetros o arquivo onde se encontra os dados gerados pelo programa presa-predador o intervalo de tempo que foi gerado aquele arquivo é armazenado, o espaçamento entre as armadilhas também foi passado como parâmetro e assumiu valor 10.

Dentro da função de extração, considera-se que as armadilhas são posicionadas nos respectivos pontos de acordo com o espaçamento entre eles. Dois laços são feitos para efetuar comparações não quais são verificados se há na base de dados, no ponto (x,y), captura pela armadilha. Ou seja, se o ponto em que a janela foi colocada estiver presença de indivíduo ele é capturado e colocado em uma nova matriz de dados (capturados). A terceira coluna dessa nova matriz recebe o intervalo de tempo de geração dos pontos. Seja D o arquivo de entrada, T o tempo em que foi gerada a distribuição dos pontos, N a quantidade total de armadilhas, G as coordenadas dos pontos onde se encontram as armadilhas e S a matriz de pontos capturados pela armadilha, o algoritmo 1 descreve o procedimento de extração dos pontos.

```
Para cada i<-1 até D faça  
Para cada j<- 1 até N  
Se D[i,:]=G[j,:] então  
S<-G[i,:]  
Fim  
Fim  
S[:,3]<-T  
Fim.
```

Algoritmo 1 – Procedimento de extração dos pontos capturados pela armadilha.

Essa função é chamada para cada distribuição nos intervalo de tempo gerado pelo presa-predador que estivermos interessados em analisar, salvando cada uma dessas capturas por intervalo de tempo em um arquivo distinto.

#### **4.2.2 Função Para Agrupar os Pontos onde Houve Captura pelas Armadilhas**

A função para agrupar os pontos capturados pelas armadilhas implementada da seguinte forma descrita a seguir.

Para cada arquivo salvo pela função de extração FE será feito o agrupamento. O seguinte procedimento é realizado tendo como parâmetro da função de agrupamento dois arquivos gerados por FE inicialmente.

O primeiro arquivo é percorrido até o final e a partir desse ponto insere os dados do segundo arquivo. Por fim retorna como resultado um arquivo contendo os dados dos dois arquivos. Esse procedimento é realizado para todos os arquivos da função de extração, o procedimento para agrupar os arquivos é mostrado no algoritmo 2 onde W é o tamanho do primeiro arquivo, Y o segundo arquivo, M o tamanho do primeiro arquivo.

*Para  $i <- 1$  até  $M$  faça*

*$W[M+1] <- Y$*

*Fim.*

Algoritmo 2 – Procedimento para agrupar dois arquivos.

Porém a partir dessa primeira passada a função passa trabalhar com os outros arquivos gerados por FE um por vez, pois o primeiro arquivo de entrada na função será o arquivo gerado na passagem anterior da mesma.



### 4.3 A função K no R

Para a análise do padrão de distribuição espacial individual para cada distribuição do presa-predador foi utilizada funções do software R e o pacote Spatstat para análise de padrões de pontos em duas dimensões. Onde os pontos gerados pelo programa presa-predador foram a base de dados de entrada para a função K. Esses procedimentos são ilustrados no fluxograma da figura 4.1.

A princípio foi utilizada a função K para a distribuição até o intervalo de tempo seis da população de presa. A função retornou o gráfico de K para cada intervalo de tempo, que será mostrado no capítulo seguinte. A descrição de como utilizar o pacote Spatstat e suas funções está nos anexos (capítulo 8), sessão 8.1.

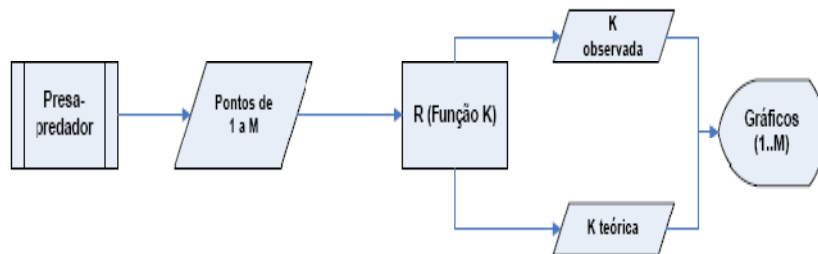


Figura 4.1 – Fluxograma da Função K no R.

### 4.4 A Função KF

A função KF é a implementação da adaptação da função K proposta por Carlslake et. Al. (2005) e foi implementada no software Scilab.

A função KF calcula o vetor  $K_s$  (Função K para detecção de cluster no espaço),  $K_t$  (Função K para detecção de cluster no tempo),  $K_{st}$  (para detecção de cluster no espaço e no tempo) para a entrada de dados a ser analisada, vetores  $s$  e  $t$  de coordenadas espaciais e temporais e um vetor com os valores máximos e mínimos da base de dados.

O arquivo de entrada são os pontos de uma área a ser analisada. Esses pontos precisam estar na forma de uma matriz com colunas  $x$ ,  $y$ ,  $z$  onde  $x$ ,  $y$  são as coordenadas espaciais e  $z$  é o intervalo de tempo.

É necessário fornecer para a função KF um vetor coluna com a escala espacial ( $s$ ), um vetor coluna com a escala temporal ( $t$ ) e as coordenadas limites em uma matriz três por dois onde cada coluna representa  $x$ ,  $y$  e  $z$  a primeira linha contém os valores das coordenadas limites inferiores e a segunda os valores superiores de  $x$ ,  $y$  e  $z$ . Um algoritmo para buscar os valores máximos e mínimos dessa matriz é necessário quando a base de dados de entrada é grande e não estiver ordenada, para facilitar esse procedimento. A matriz com as coordenadas limites é chamada de *ranges*.

E  $s$  assume valores no intervalo de um ao valor igual a 25% (vinte e cinco por cento) da maior distância máxima de separação entre dois pontos. Essa distância é calculado com o Teorema de Pitágoras. Onde a maior distância entre dois pontos é a diagonal de uma matriz quadrática de tamanho igual ao maior ponto da base da dados.

Os valores do vetor coluna  $t$  são números dentro de um intervalo entre 0 e 25% do maior tempo dentro da base de dados. Detalhes de como é a sintaxe para passagem desses dados para a função KF é mostrado nos anexos.

#### 4.4.1 Algoritmo da função KF

Agora será apresentado detalhes da implementação da função KF, será descrito alguns procedimentos necessários para se chegar aos resultados esperados.

A função primeiramente faz uma checagem onde é verificado se os valores de  $s$ ,  $t$  e  $ranges$  existem, se estão corretamente passados a KF e se estão em conformidade com a base de dados passada e também é checado se a base de dados  $xyz\_data$  está correta. Caso alguma destas checagem falhe uma mensagem ao usuário será emitida a informando o problema, as mensagens são:

“ERROR. Spatial Scales (s), not defined”

“ERROR. Temporal scales (t), not defined”

“ERROR. Limits x, y, z (ranges) not defined”

“ERROR. xyz\_data must contain three columns”

“ERROR. Some xyx\_data fall outside ranges”

Após a checagem é definido e criado os vetores de saída  $K_s$ ,  $K_t$  e  $K_{st}$ .

Em seguida para cada par de coordenada  $i$  e  $j$  será examinado por vez para calcular os valores de  $d_{ij}$  (distância espacial entre os eventos  $i$  e  $j$ ), valores de  $t_{ij}$  (distância temporal entre os eventos  $i$  e  $j$  onde  $i$  é diferente de  $j$ ), valores de  $w_{ij}$  (fator de correção de efeito de borda) e valores de  $v_{ij}$  (fator de efeito de correção de efeito de borda no tempo). Como é efetuado o cálculo destes valores é apresentado no código em anexo.

O próximo passo é calcular os resultados de  $K_s$ ,  $K_t$  e  $K_{st}$ , o qual é calculado atribuindo a cada valor de  $s$ ,  $t$  e  $(s, t)$  os valores de  $K_s$ ,  $K_t$  e  $K_{st}$  respectivamente.

Por fim é feito o calculo de Dst e D0st e plota-se a matriz D0st.

Onde Dst é obtido pela implementação da equação 4.1 e D0st pela equação 4.2.

$$D(s,t) = K(s,t) - K(s)*K(t) \quad (4.1)$$

$$D0(s,t) = D(s,t)/[K(s)*K(t)] \quad (4.2)$$

A figura 4.2 é mostra o fluxograma que representa a estrutura da implementação de KF, descrita nessa seção.

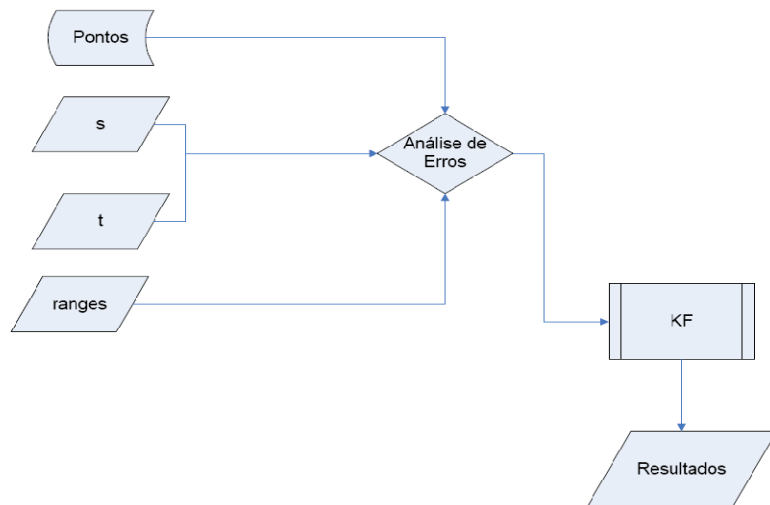


Figura 4.2 – Fluxograma de KF, dadas as entradas os resultados.

No próximo capítulo serão abordados os resultados e discussões decorrentes do desenvolvimento das funções apresentadas.

## 5 RESULTADOS E DISCUSSÃO

Os resultados serão apresentados em duas partes: Na primeira, serão apresentados os resultados da função  $K$  para cada configuração espacial gerada pelo programa presa-predador. Nesta primeira parte, o objetivo é apenas detectar presença de agrupamentos no espaço em cada unidade de tempo. Na segunda parte, serão apresentados os resultados da implementação da função  $KF$  no Scilab com o objetivo de detectar interação espaço-temporal no conjunto de dados.

### 5.1 Análise espacial

O objetivo da implementação no R foi verificar se os pontos onde foi capturado os indivíduos exibe algum padrão sistemático, tipo regularidade, agrupamento, aleatoriedade. Essa análise foi feita para uma amostra pequena de apenas 6 distribuições do programa presa-predador, que representa 6 períodos de tempo diferentes.

Foram gerados seis gráficos, onde no eixo  $y$  temos a função  $K$  observada que é dada em preto e a função  $K$  esperada, sob a hipótese de aleatoriedade espacial, que é dada em vermelho, contra a distância (eixo  $x$ ).

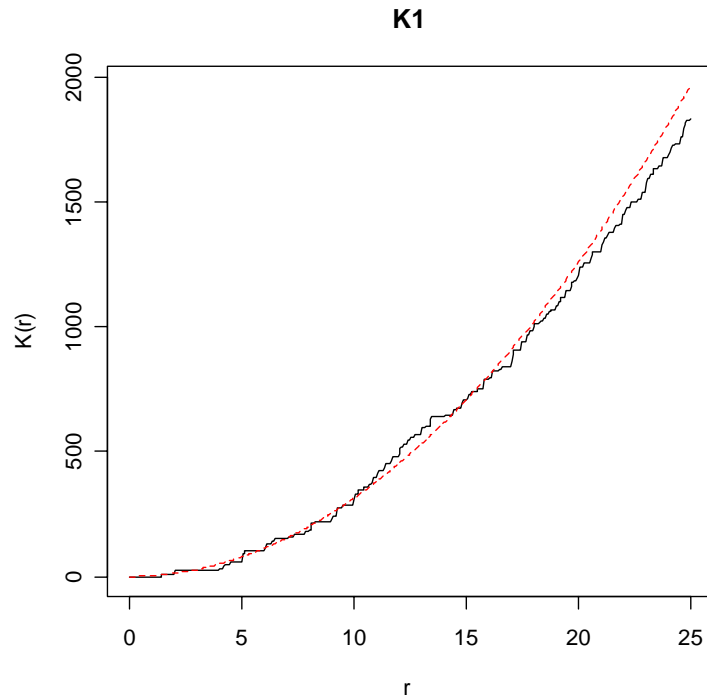


Figura 5.1 Gráfico da função K para o período de tempo um.

Analisando a figura 5.1 é observada a regularidade na distribuição da população de presas pela observação das curvas da função K observada (cor preto) e da função K teórica (cor vermelha), a sobreposição da mesma evidencia regularidade. Vale observar também que para um raio  $r$  maior que 15 a K observada assume valores inferiores aos da K teórica o que evidencia repulsão entre os pontos, ou seja eles vão ficando mais esparsos.

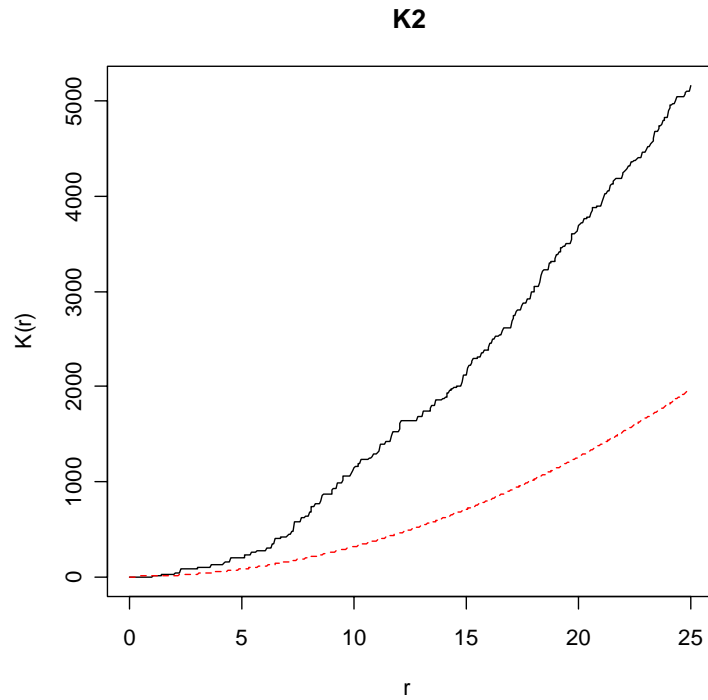


Figura 5.2 Gráfico da função K para tempo dois.

A análise da figura 5.2 observou a formação de conglomerados que se acentua a partir do raio  $r$  de valor 5. A função  $K$  observada acima da função  $K$  teórica evidencia a presença de clusters.

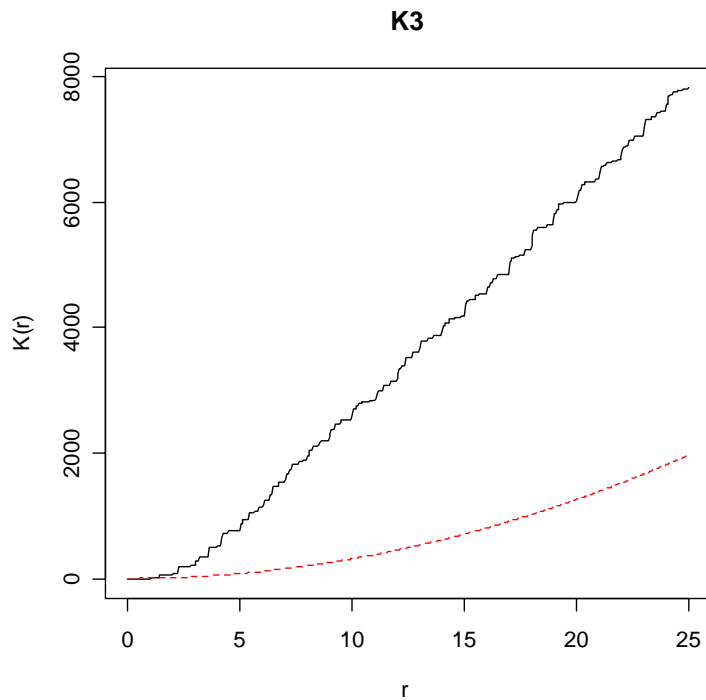


Figura 5.3 Gráfico da função K para tempo três.

Na figura 5.3 análise se faz igual aos gráficos anteriores, onde observamos a presença de clusters para a distribuição da população. Observando a escala de  $K(r)$  percebe-se um aumento em relação a mesma escala da figura 5.2, este aumento se dá pelo fato de a distribuição da população para o tempo 3 representado pela figura 5.3 ser maior que a distribuição da população 2, figura 5.2.



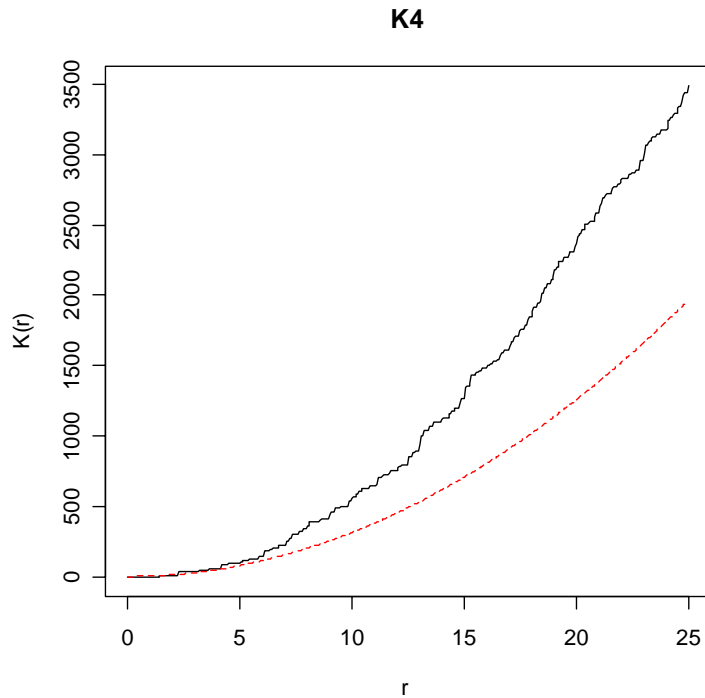


Figura 5.4 Gráfico da função K para tempo quatro.

Há a formação de cluster na figura 5.4 fato observado pela comparação das curvas, como realizado nas distribuições anteriores. Observando novamente a escala de  $K(r)$  é perceptível a diminuição dos valores, isso devido a diminuição da distribuição da população para esse intervalo de tempo. Como se trata de um modelo de dinâmica de população presa-predador, é possível inferir que essa discrepância entre as distribuições é devida a influência do outro indivíduo no meio, no caso influência do predador já que estamos analisando a distribuição das presas. O estudo da causa dessa interação entre os indivíduos fica a cargo de um profissional da área.

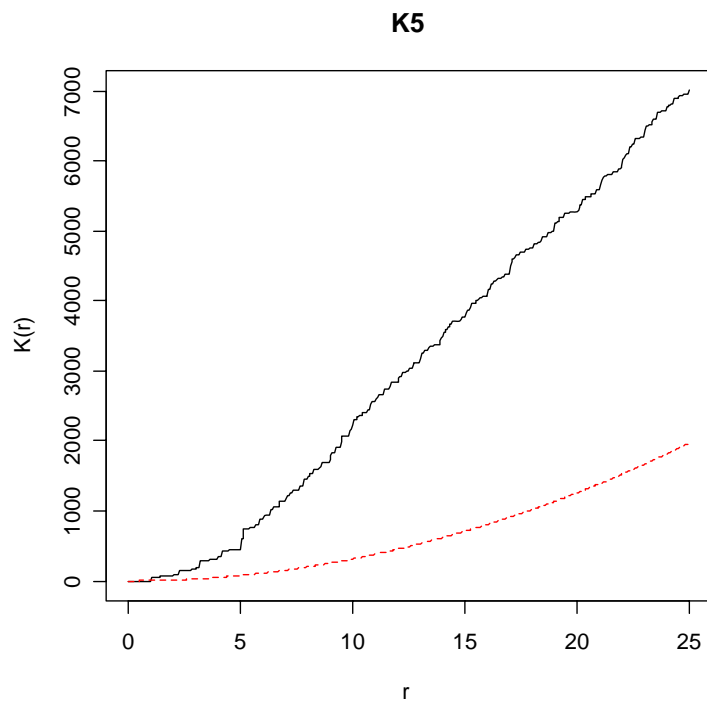


Figura 5.5 Gráfico da função K para tempo cinco.

Na figura 5.5 permanece a aglomeração da população de presa, porém com uma distribuição maior em relação a figura 5.4.

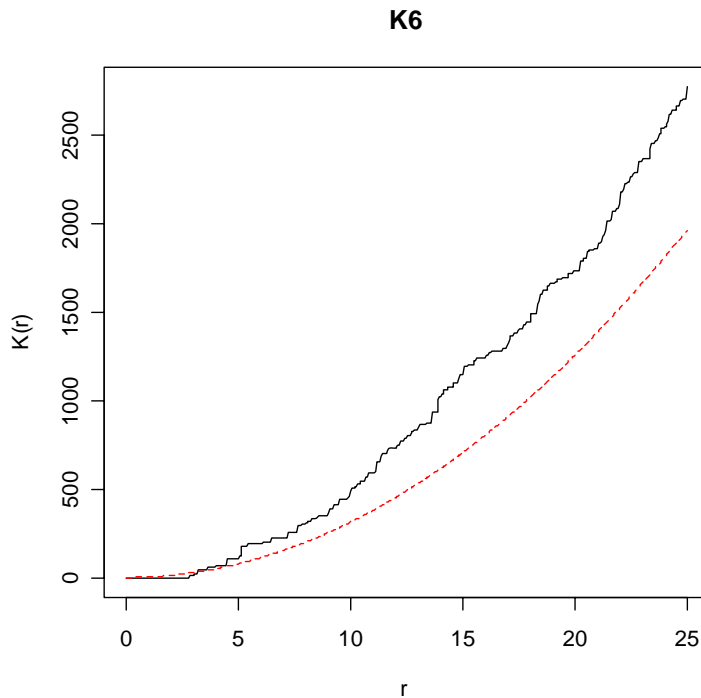


Figura 5.6 Gráfico da função K para tempo seis.

Na figura 5.6 já é possível observar a diminuição no tamanho do cluster, pela visualização da diminuição dos valores de  $K(r)$ , como também a diminuição da distribuição dos indivíduos no espaço.

Como já comentado os gráficos da função  $K$  podem ser interpretados da seguinte maneira: Se a função observada está acima da função teórica, então o número de observações é maior do que seria esperado para uma distribuição aleatória para uma determinada distância. Caso contrário, há indícios de regularidade. Assim, o primeiro gráfico mostra que a população começa com distribuição aleatória ( $k1$ ) e com o passar do tempo vai formando conglomerados.

A análise pelo método também permitiu ver as variações da distribuição dos indivíduos no decorrer do tempo.

## 5.2 Análise espaço-temporal

A implementação para o conjunto de dados do modelo Presa-Predador da distribuição 1 a 20 ou seja, 20 intervalos de tempo, gerou o gráfico mostrado na figura 5.7, sendo o eixo x em metros e o eixo y em uma unidade de tempo, semana por exemplo:

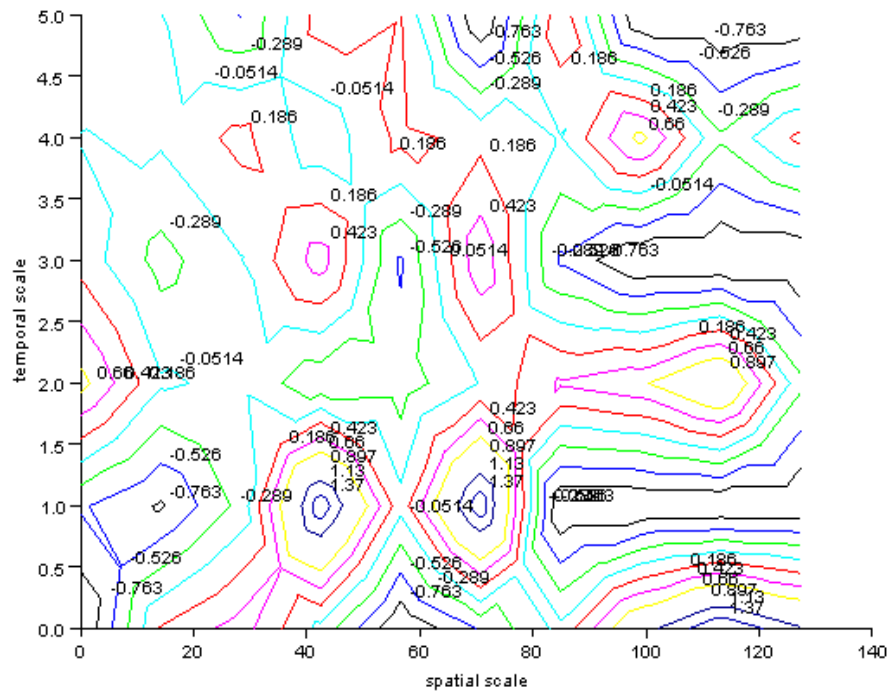


Figura 5.7 Gráfico de  $D0st$ .

Para a análise dos resultados é preciso entender o que é o gráfico de  $D0st$ , que é definido pelas equações 5.1 e 5.2, abaixo.

$$D(s,t) = K(s,t) - K(s)K(t) \quad (5.1)$$

$$D0(s,t) = D(s,t)/[K(s)K(t)] \quad (5.2)$$

$D(s,t)$  é o cluster adicional no espaço-tempo e  $D0(s,t)$  é o aumento proporcional do cluster no espaço-tempo que é devido a interação espaço-tempo, e ambos esperam valores de 0 (zero) para a hipótese nula de independência espaço-temporal.

Analisando o gráfico nota-se claramente as regiões de agrupamento. Sendo que quanto mais acima estiverem os valores de  $D0(s,t)$  em relação hipótese nula (0), mais há interação, assim maior é o aumento do cluster no espaço-tempo (interação positiva). Nos valores observados onde assume valores negativos (interação negativa) conclui-se que há agrupamento no tempo ou no espaço, mas não em ambos, ou seja há uma interação negativa entre espaço e tempo.

Os resultados dessa implementação são obtidos através da interpretação do gráfico mostrado na figura 5.7.

Para garantir a eficiência da função KF iremos comparar todos os gráficos gerados por KF e pela implementação da função K para cada distribuição individualmente.

### **5.3 Comparação dos Métodos**

A comparação entre os métodos é necessária para verificar se KF está produzindo resultados que satisfazem as expectativas quanto a sua finalidade que é a análise de conglomerados no espaço e no tempo em modelos de dinâmica de populações. Como o objetivo dessa comparação é apenas verificar se a função KF produz resultados satisfatórios julgamos neste trabalho comparar uma pequeno conjunto de dados, por isso utilizou na implementação da função K tempos de um a seis e suas respectivas distribuições.

A comparação é feita visualmente, analisando os gráficos da implementação da função K, para cada distribuição nos períodos de tempo, com o gráfico de Kt (Função K no tempo), gerados por KF. Os gráficos da função K são apresentados na figura 5.8 e o de Kt, na figura 5.9.

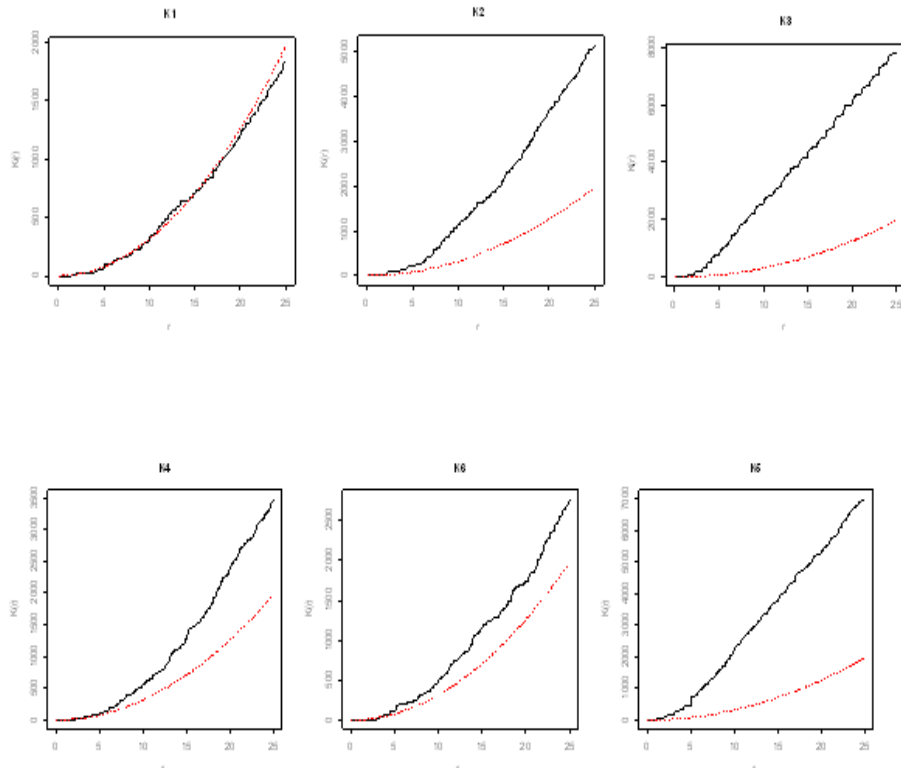


Figura 5.8 – Gráficos da função K.

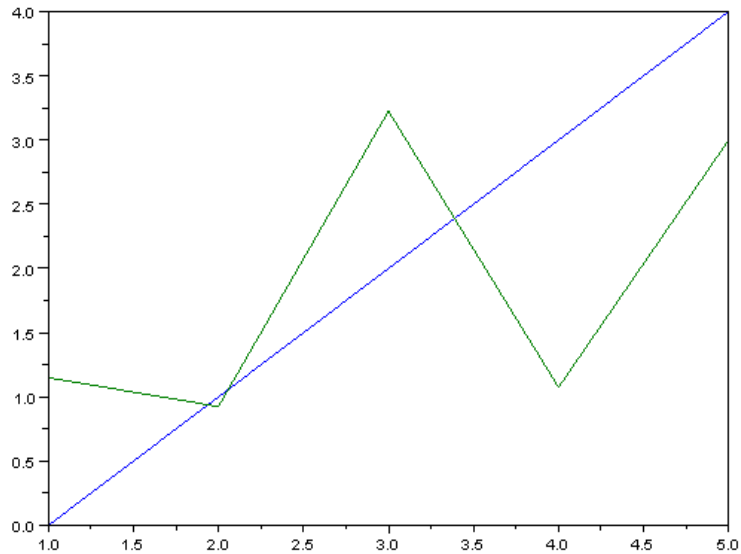


Figura 5.9 – Gráfico de Kt (Tempo x Kt)

Analisando os gráficos das figuras 5.8 e 5.9 paralelamente é possível ver a similaridade dos tempos em que há a formação de clusters. Para tal visualização basta observar o valor do eixo x (tempo) da figura 5.9 em que há picos. Os picos representam os valores máximos de clusters. Observado o valor do tempo em que há picos basta ir até o a figura 5.8 e localizar o gráfico para a distribuição para esse tempo e verificar se há mesmo formação de cluster. No tempo 3 há um pico (figura 5.9) indo até a distribuição K3 da figura 5.8 observamos a curva característica de formação de cluster o que é suficiente para validar os resultados obtidos pela função KF.

Com isso chegamos às conclusões e propostas futuras, no próximo capítulo.



## 6 APLICAÇÕES EM REDES SEM FIO

Neste capítulo será discutido a aplicação da função de análise espaço-tempo ( função KF) em redes sem fio.

Uma rede de computadores consiste em dois ou mais computadores e/ou outros dispositivos conectados entre si de modo que possam compartilhar serviços, que podem ser dados, mensagens, etc. Para o estudo espaço-temporal proposto neste capítulo, trataremos de redes sem fio (redes wireless), abordando duas situações: redes de sensores sem fio e redes de telefonia móvel.

### 6.1 Redes de sensores sem fio (RSSF)

Os avanços na tecnologia da microeletrônica possibilitaram a construção e a instalação de densas redes de comunicação sem fios destinadas a coletar e disseminar uma grande variedade de dados das mais diversas áreas, envolvendo grandezas físicas, químicas, biológicas, entre outras (RABAEY, 2000). Uma rede de comunicação deste tipo, coleta dados do ambiente através de sensores instalados em vários de seus nós (denominados “nós sensores”), e transmite esses dados por meios não guiados (meios óticos ou sinais de rádio), sendo então chamada de rede de sensores sem fio (RSSF), ou em inglês, *wireless sensor networks* (WSN) (BULUSU, 2004; HILL, 2004). As RSSFs diferem das redes de computadores tradicionais em vários aspectos. Normalmente essas redes possuem um grande número de nós distribuídos (até milhares), têm restrições de energia porque são alimentados por baterias, e devem possuir mecanismos para auto-configuração e adaptação, visto que os nós são depositados, geralmente, de

uma forma aleatória, possuem comunicação *ad hoc* e eles podem falhar ou se mover (YE, 2003; SZEWCZYK, 2004).

Como estes nós da rede estão limitados a uma área geográfica, podem falhar ou se mover dentro dessa área, as RSSF podem ser analisadas como uma dinâmica de eventos no espaço-tempo. Obtendo as coordenadas dos nós da RSSF em intervalos de tempo periódicos será possível a análise espaço-tempo através da função KF.

Por meio da é obtido o padrão de distribuição da dos nós sensores no decorrer do tempo possibilitando com isso uma aproximação do comportamento da rede. Dessa forma é possível utilizar esses dados para possíveis previsões da configuração da rede em intervalos de tempo futuros.

A seguir é apresentada a simulação realizada no software R de um RSSF.

### **6.1.1 Simulação em Uma RSSF**

Foi simulado no R algumas distribuições de pontos que representa os nós sensores no espaço a ser monitorado, cada distribuição representa a observação dos nós sensores ligados em um determinado período de tempo, dentro da área de observação conforme figuras 6.1, 6.3, 6.5, 6.7 e 6.9 . Em seguida para cada uma das simulações foi feita a análise pela função K conforme figuras 6.2, 6.4, 6.6, 6.8 e 6.10 respectivamente. A distribuição dos nós sensores foi feita através de uma função randomica de geração de padrões de pontos disponível na biblioteca Spatstat

**Simulação 1**

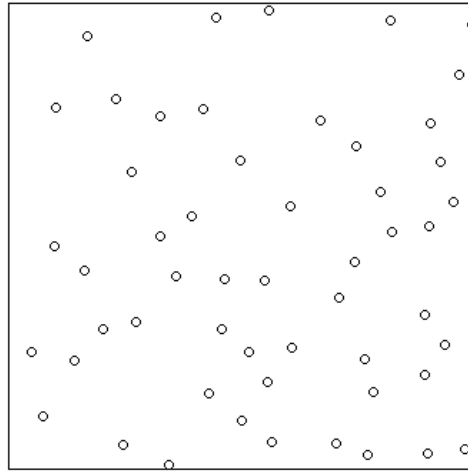


Figura 6.1 Primeira simulação da distribuição dos nós sensores – Rede esparsa.

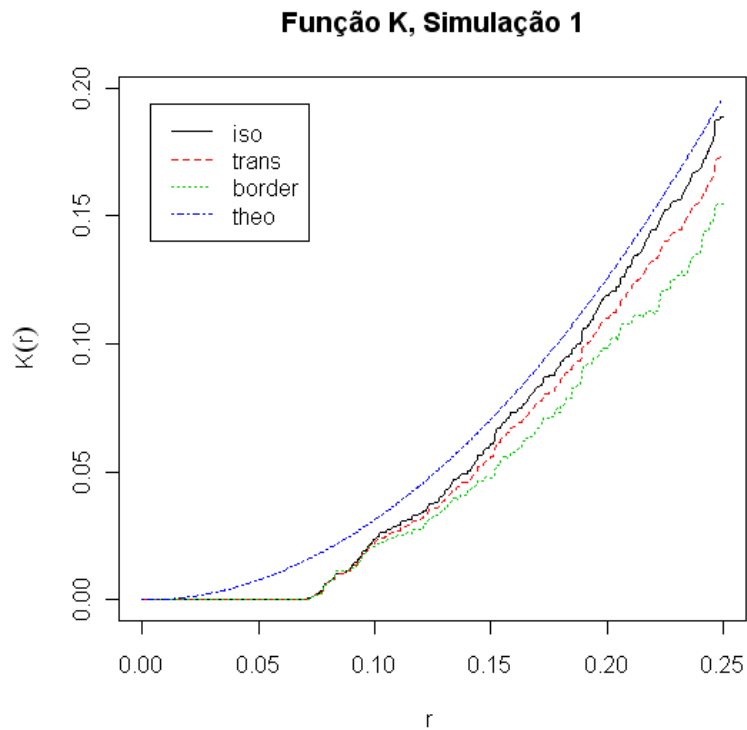


Figura 6.2 Função K para a primeira simulação.

A figura 6.2 mostra as curvas para os valores de K para os pontos da primeira simulação, pela análise das curvas de acordo com a metodologia de comparação entre a curva da K teórica e da K observada é possível constatar a irregularidade da distribuição dos nós sensores o que é comprovado observando a figura 6.1.

### Simulação 2

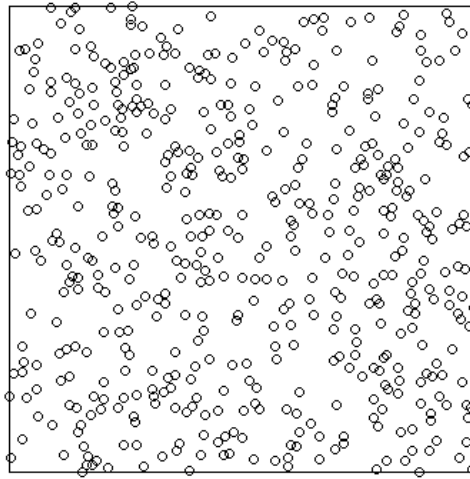


Figura 6.3 Segunda simulação da distribuição dos nós sensores – Rede densa.

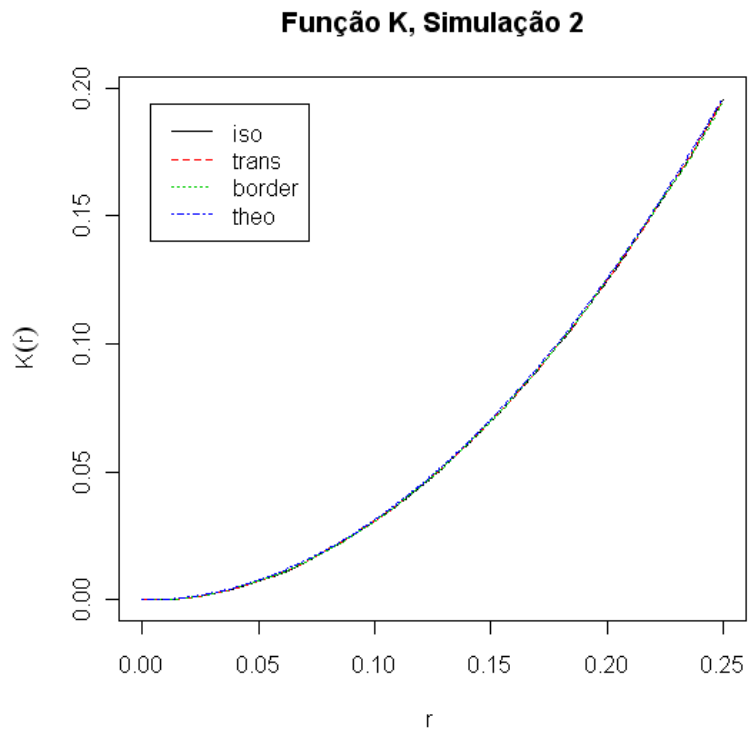


Figura 6.4 Função K para a segunda simulação.

Pela análise do gráfico da figura 6.4 constata-se a regularidade da distribuição da figura 6.3.

**Simulação 3**

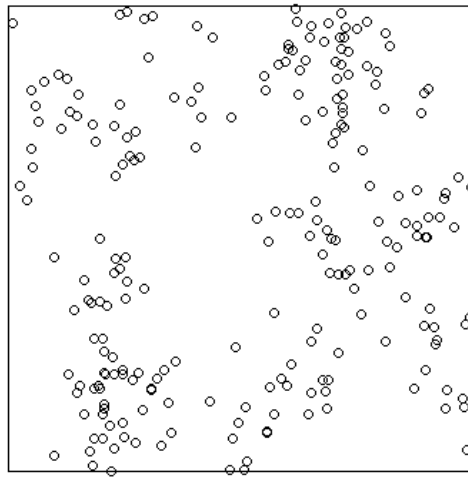


Figura 6.5 Terceira simulação da distribuição dos nós sensores – Rede esparsa com agrupamentos.

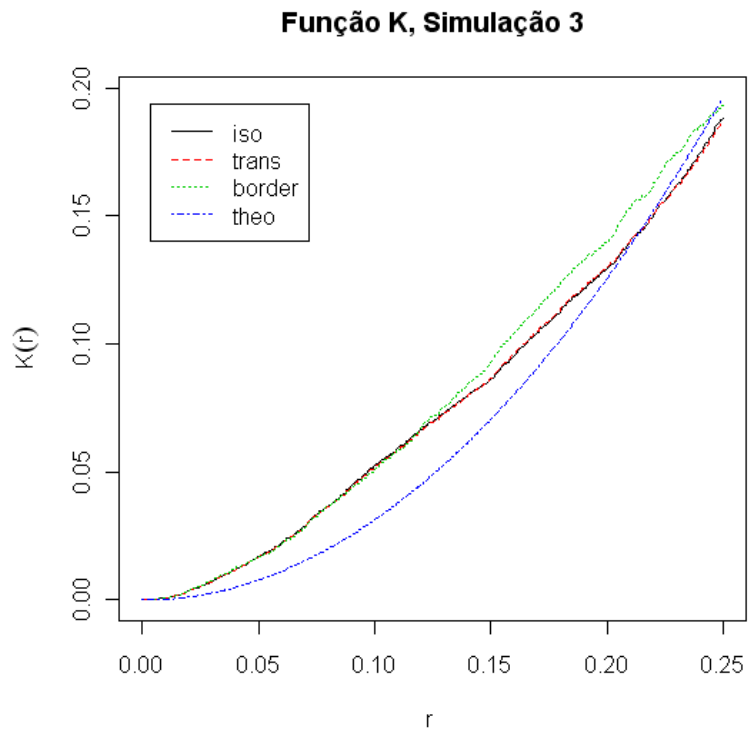


Figura 6.6 Função K para a terceira simulação.

Na figura 6.6 há indícios de agrupamento pela comparação entre as curvas teórica e observada, pela visualização da figura 6.5 comprovamos formação discreta de agrupamentos.



#### Simulação 4

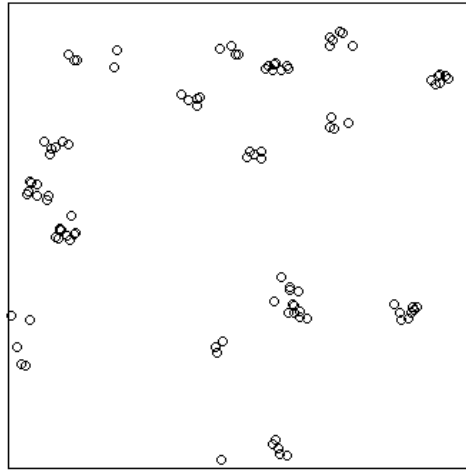


Figura 6.7 Quarta simulação da distribuição dos nós sensores.

### Simulação 4

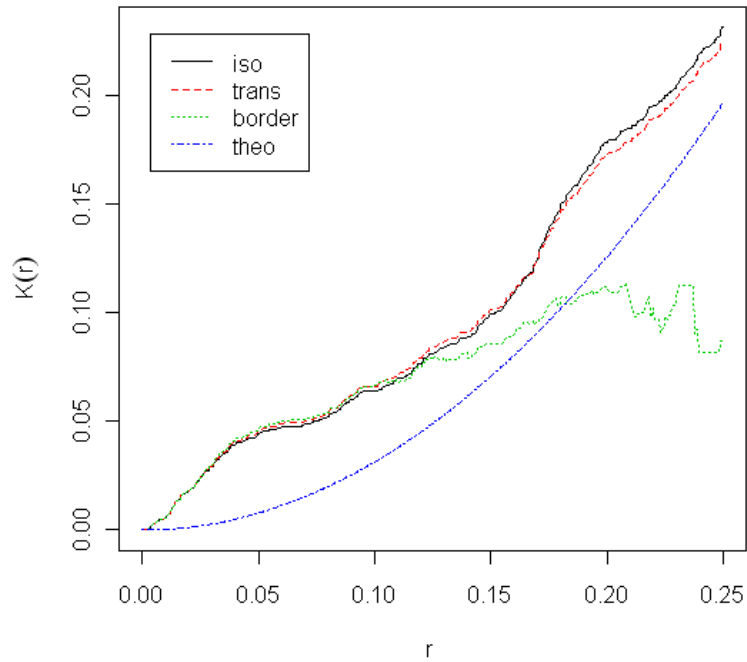


Figura 6.8 Função K para a quarta simulação.

O gráfico da figura 6.8 diz acontecer agrupamentos na quarta simulação, observando a distribuição dos nós sensores da figura 6.7, comprova-se tal indicio, há discretos agrupamentos.

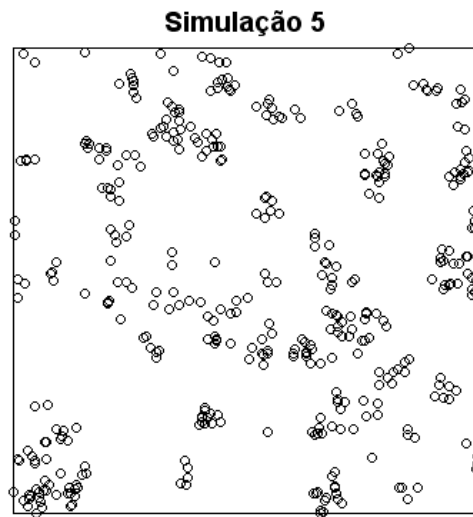


Figura 6.9 Quinta simulação da distribuição dos nós sensores.

### Simulação 5

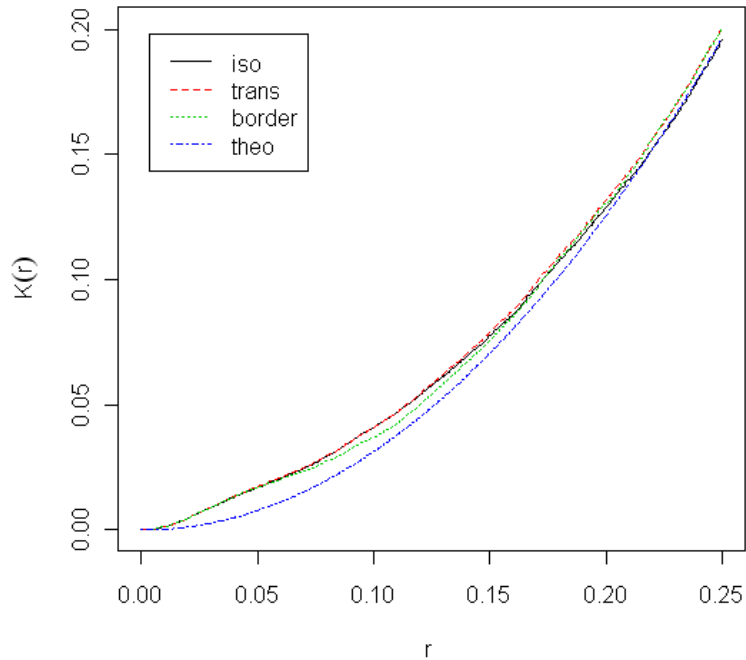


Figura 6.10 Função K para a quinta simulação.

A análise de 6.10 diz haver agrupamentos porém em uma distribuição um tanto quanto regular o que é observado pela figura 6.9.

Pela análise geral pode-se inferir que a rede tem a tendência de se regularizar. Com isso poderia o projetista da rede eliminar ou desligar nós sensores desnecessários para o monitoramento da área em questão.

Uma segunda análise poderia ser feita através da função KF para detectar a interação espaço tempo e ver os intervalos de tempo com maior interação, o que possibilitaria eliminar nós desnecessários naqueles intervalos de maior interação especificamente, observa-se que com maior interação mais

agrupamentos serão formados e com isso existirá coleta de dados redundantes pelos nós sensores que estão agrupados.

## **6.2 Redes de Telefonia Móvel**

A área de abrangência de uma rede móvel é decorrente da área de cobertura de uma estação rádio base (ERB). Uma ERB nada mais é que um sistema de rádio e antenas de comunicação que permite a cobertura de uma área específica, chamada de célula (o termo telefonia celular provém deste conceito). A ERB está ligada a uma central de comutação celular que é uma central telefônica com funções específicas para uma rede de telefonia móvel.

Os usuários dessa rede de telefonia se movem dentro dessa rede como também encerram sua comunicação. Com isso a configuração espacial destes usuários apresenta uma dinâmica no espaço. Observando a distribuição destes usuários em intervalos de tempo é possível analisar a interação espaço-tempo que ocorre na área de cobertura de uma ERB específica. A figura 6.11 representa os usuários móveis dentro da área de cobertura de uma ERB.

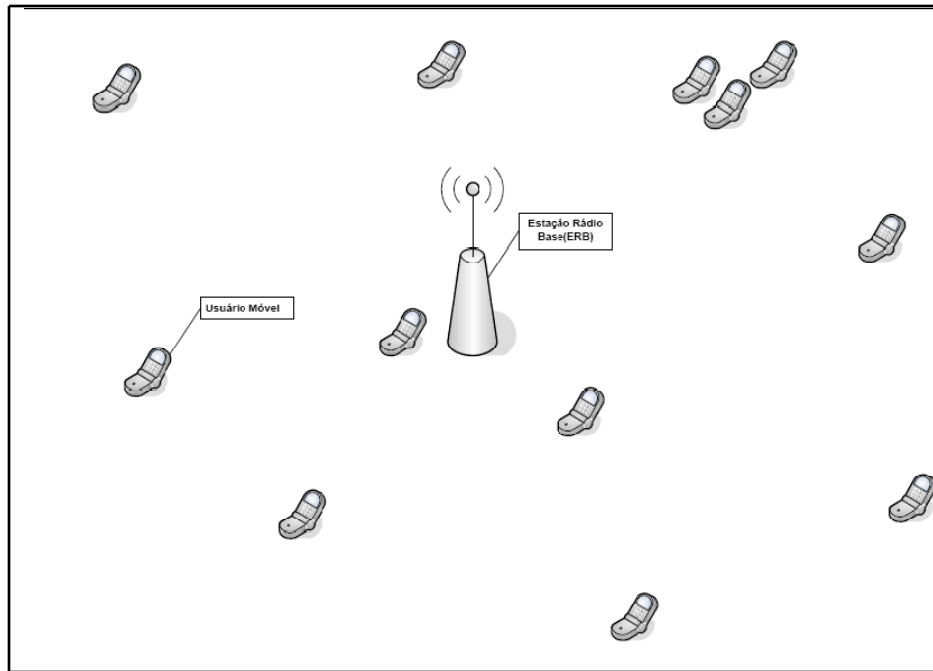


Figura 6.11 Usuários móveis dentro da área de cobertura de uma ERB.

### 6.2.1 Simulação de uma rede de celular

Para a rede de celulares foi gerado uma matriz randomicamente com 3 colunas, representando as coordenadas dos eventos (pontos onde o usuário móvel foi “capturado”) nas duas primeiras colunas e o tempo em que o evento ocorreu na terceira. Foram gerados pelo R, nesta simulação, 715 eventos.

A análise realizada por meio da função KF em cima desse conjunto de dados gerou o gráfico da figura 6.12 abaixo.

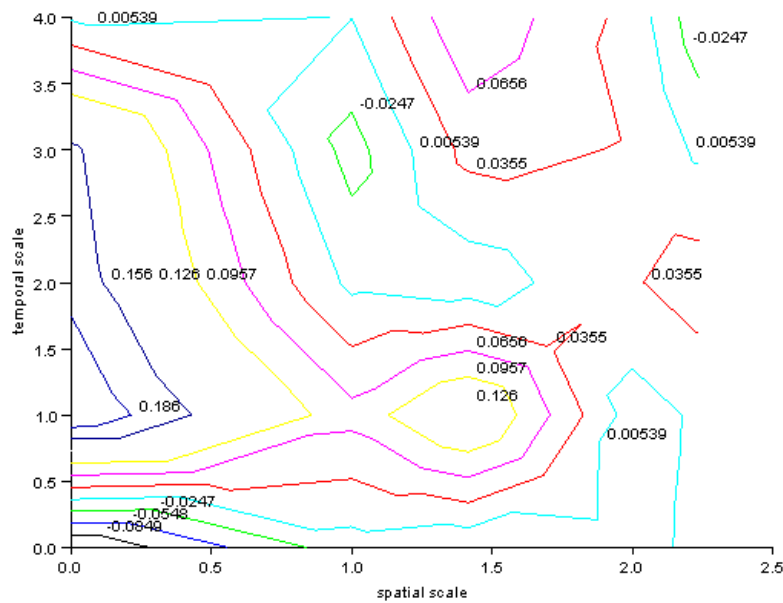


Figura 6.12 Gráfico da interação espaço-tempo (D0st).

A análise do gráfico da figura 6.12 mostra que os valores acima de 0 (zero) representam interações espaço-tempo positiva, e abaixo de 0 interação negativas e 0 interação nula.

Com a análise espaço-tempo é possível observar os períodos de tempo que ocorrem interação espaço-tempo positiva (em que há agrupamentos no espaço e no tempo), tendo estes resultados como uma possível previsão dos períodos de tempo em que ocorrem maiores agrupamentos a partir desta observação a ERB passaria a disponibilizar mais canais de comunicação para estes usuários nos períodos de picos (interação positiva), e com isso evitar uma possível sobrecarga da rede.

## **6.3 Conclusão**

Com essas implementações conclui-se que as funções de análise espaço-tempo podem ser uteis para o gerenciamento de redes de sensores sem fio e também para evitar sobrecarga em redes de telefonia móvel.



## **7 CONCLUSÕES**

### **7.1 Conclusão**

Considerando os objetivos levantados no início deste trabalho, conclui-se que foi desenvolvida uma função que oferece as finalidades para uma completa análise da interação espaço-temporal em processos pontuais descontínuos no tempo e no espaço, a qual se mostrou eficaz para o modelo de dinâmica de populações.

#### **7.1.1 Função K no R**

Conclui-se que com o método proposto foi possível não somente detectar a presença de conglomerados espaciais, mas também visualizar por meios gráficos no decorrer do tempo a dinâmica temporal das distribuições espaciais da população, que começou com uma distribuição aleatória e ao decorrer do tempo foi formando agrupamentos.

Também é possível verificar a interação entre os indivíduos de diferentes espécies, porém este estudo não foi aprofundado neste trabalho.

### 7.1.2 A função KF

Os resultados mostram que os métodos estatísticos possibilitaram detectar a presença de conglomerados no tempo e espaço em populações cujos indivíduos estão em movimento em uma área regular contínua. A interação espaço-tempo pode ser vista através dos resultados da implementação do método, onde pode se ver a caracterização de agrupamentos (clusters), regularidade e aleatoriedade.

Através da comparação entre os gráficos gerados pela função K no R (figura 5.8) e pelo gráfico de Kt (figura 5.9) gerado por KF foi possível verificar a eficácia do método de KF para a análise espaço-temporal em modelos de dinâmica de população.

## 7.2 Propostas Futuras

A análise pela função K para diferentes espécies que convivem no mesmo habitat, possibilita ver a interferência que uma causa na outra na distribuição dessas espécies no meio. Um estudo detalhado aplicando a função K e a função para análise espaço-tempo (KF) para diferentes espécies que convivem no mesmo habitat seria de grande valor científico.

A função KF foi desenvolvida em um ambiente de programação interpretada, o Scilab. Apesar de apresentar resultados satisfatórios esse ambiente limita o uso da função KF. A implementação da função KF em uma linguagem de programação como C++ poderia sanar essa deficiência.

O uso das funções K e KF para detecção da interação em processos pontuais pode ser estendido a outros casos visto o seu desempenho eficaz como forma de

representar os fenômenos e interações entre diferentes indivíduos e sistemas diante das modificações realizadas em seu meio. Além disso, através dessa modelagem, os experimentos podem ser reproduzidos, verificados e comprovados por qualquer pesquisador em diferentes áreas que se possam usar processos pontuais.

Também é proposto a aplicação do método para os casos mostrado no capítulo 6 em redes sem fio, em modelos reais.



## 8 REFERENCIAL BIBLIOGRÁFICO

- AGUIAR, M. A. M.; RAUCH, E. M.; BAR-YAM, Y. (2004). Invasion and Extinction in the mean field approximation for a spatial host-pathogen model. *Journal of Statistical Physics* 114: 1447-1451.
- BADDELEY, A., TURNER, R. (2005) *spatstat*: an R package for analyzing spatial point patterns. *Journal of Statistical Software* **12** (2005) 1-42.
- BAILEY, T. C.; GATRELL, A. C. (1995). *Interactive spatial data analysis*. Essex: Longman Scientific.
- BULUSU, N.; HEIDEMANN, J.; ESTRIN, D.(2004) *Self-Configuring Localization Systems: Design and Experimental Evaluation*; ACM Transactions on Embedded Computing Systems, vol. 3, nº 1, pp. 24-60, 2004
- CARNEIRO, M. V., CHARRET, I. C. (2007). Spontaneous emergence of spatial patterns in a predator-prey model. *To appear*.
- CARSLAKE, D., BENNETT, M., BROWN, K., HAZEL, S., TELFER, S., BEGON, M. (2005). Space time clustering of cowpox virus infection in wild rodent populations. *Journal of Animal Ecology* 74: 647-655.
- CARVALHO, K. C., TOME, T. (2004). Probabilistic cellular automata describing a biological two species system. *Modern Physics Letter B* 17: 873:880.
- CARVALHO, K. C., TOME, T. (2007). Self organized patterns of coexistence out of a predator prey cellular automaton. *To appear Modern Physics Letter.*
- COSTA, F. A. P. L. (2004). O estudo de populações biológicas. *Ciência Hoje on line*. Disponível em <http://cienciahoje.uol.com.br/view/2347>.
- CRESSIE, N. (1991). *Statistics for spatial data*. Chichester: John Wiley.
- DÁVALOS, R.V. *O ensino de simulação de sistemas nos cursos de engenharia e informática*. UNISUL.
- DIGGLE, P. J. (2002). *Statistical analysis of spatial point patterns*. London: Arnold.

- DIGGLE, P. J.; CHETWYND, A. G.; HAGGKIVIST, R., MORRIS, S. E. (1995).  
Second order analysis of space-time clustering. *Statistical Methods in  
Medical Research* 4: 124:136.
- GUTIERREZ, A. P. (1996). *Applied Population Ecology: A supply-demand  
approach*. New York: John Wiley & sons.
- HILL, J. et al.(2004) The Platforms Enabling Wireless Sensor Networks;  
Communications of the ACM, vol. 47, nº 6, pp. 41-46, 2004.
- JUNG, C. F. (2004). *Metodologia para pesquisa & desenvolvimento: aplicada a  
novas tecnologias, produtos e processos*. Rio de Janeiro/RJ: Axcel Books  
do Brasil Editora, 2004.
- KING, A. A.; HASTINGS, A. (2003). Spatial mechanisms for coexistence of species  
sharing a common natural enemy. *Theoretical Population Biology* 64:  
431:438.
- KING, A. A., SCHAFFER, W. (2001). The geometry of a population cycle: A  
mechanism model of snowshoe hare demography. *Ecology* 82: 814-830.
- LAKATOS, E.V.; MA, M. de A.. *Metodologia científica*. São Paulo/SP: Editora  
Atlas. 1988.
- LOTKA, A. (1920). Analytical note on certain rhythmic relations in organic systems.  
*Proceedings of the National Academy of Sciences* 6: 410-415.
- PAEZ, M. S., GAMERMAN, D. (2005). *Modelagem de processos espaço-temporais*.  
11 Escola de Series Temporais. São Paulo: ABE.
- R Development Core Team (2006). *R: a language and environment for statistical  
computing*. Vienna: R Foundation for Statistical Computing. www.r-  
project.org. Version 2.3.1
- RABAEY, J. et al(2000); *Picoradio Supports Ad Hoc Ultra Low Power Wireless  
Networking*; IEEE Computer, pp. 42-48.
- RIPLEY, B. D. (1977). Modelling Spatial Patterns (with discussion). *Journal of the  
Royal Statistical Society B* **39**: 172 - 212.

- SCALON, J.D., FIELLE, N. R. J., STILLMAN, E., ATCKINSON, H. (2003). Spatial pattern analysis of second-phase particles in composite materials. *Material Science and Engineering A – Structural Materials Properties Microstructure and Processing* 356: 245-257.
- SCHMIDT, A. M., SANZO, B. (2006). *Modelagem bayesiana da estrutura de covariância de processos espaço-temporais*. 17 Simpósio Nacional de Probabilidade e Estatística. São Paulo: ABE.
- SHERRATT, J. (1996). *Periodic travelling waves in a family of deterministic cellular automata*. *Physica D* 95: 319-335.
- SCILAB, DIGITEO (2003). *The open source platform for numerical computation*. <http://www.scilab.org/>.
- SILVA, G. L., DEAN, C. B. (2006). *Uma introdução à análise de modelos espaço-temporais para taxas, proporções e processos de multi-estados*. 17 Simpósio Nacional de Probabilidade e Estatística. São Paulo: ABE.
- SOLOMON, M. E. 1980. *Dinâmica de populações*. SP, EPU & Edusp.
- SZEWCZYK, R.; OSTERWEIL, E.(2004) Polastre, J.; *Habitat Monitoring with Sensor Networks*; Communications of the ACM, v. 47, pp.34-40.
- TILMAN, D.; KAREIVA, P. (1997). *Spatial ecology: The role of space in populations dynamics and interspecific interactions*. Princenton: Princenton University Press.
- VOLTERRA, V. (1931). *Leçon sur la théorie Mathématique de la lutte pour la vie*. Paris: Gauthier-Villars.
- Ye, W. & Heidemann, J.(2003)Medium Access Control in Wireless Sensor Networks.USC/ISI Technical Report, 2003, 8 p.
- WILSON, W. G., ROSS, A. M., McCAULEY, E. (1993). Spatial instabilities within the diffusive Lotka-Volterra system: Individual-based simulation results. *Theoretical Population Biology* 43: 91:127.







## 9 ANEXOS

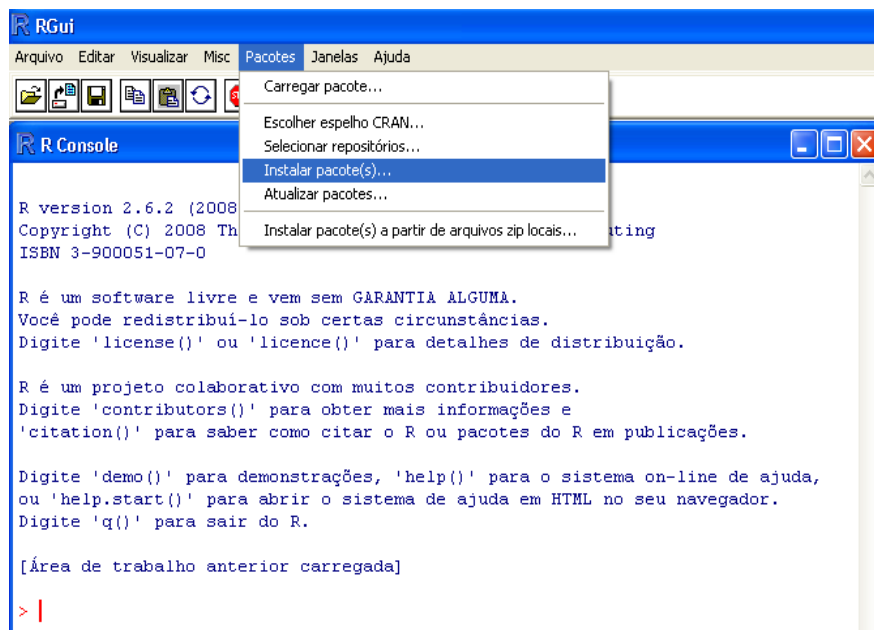
### 9.1 Spatstat

Nesta sessão serão descritos os procedimentos para a instalação e utilização do pacote Spatstat no R.

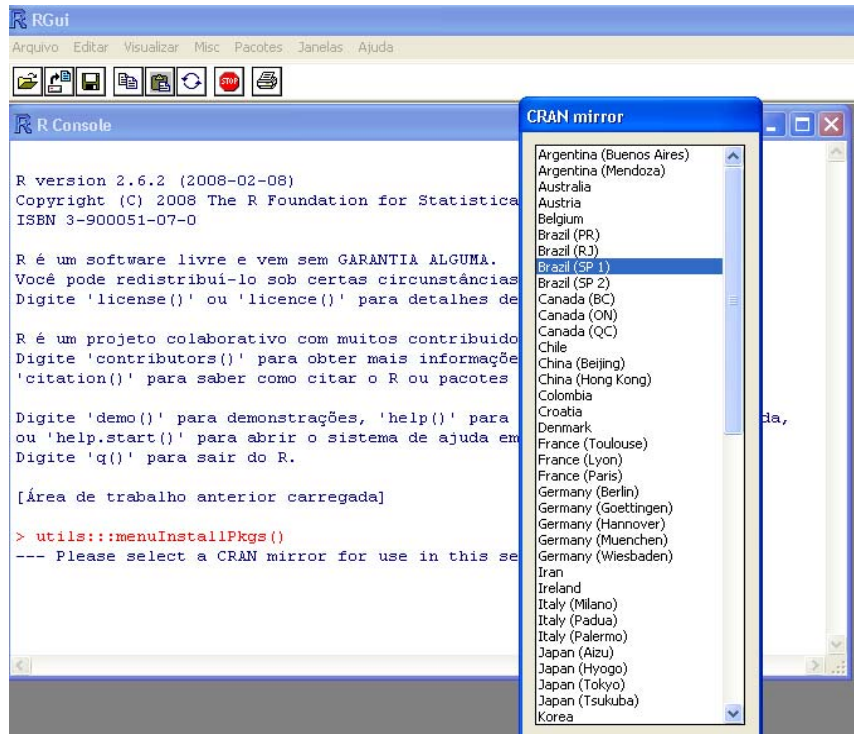
#### 9.1.1 Instalando o Spatstat

Para instalar o pacote Spatstat no R os seguintes passos devem ser seguidos:

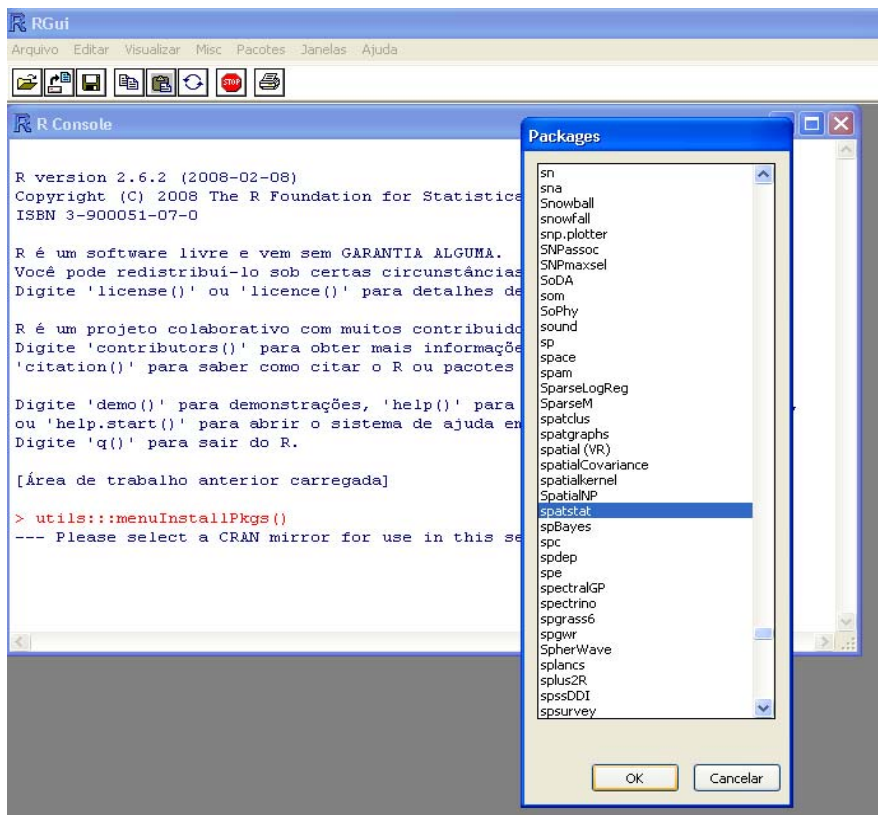
Abra o Software R, no menu Pacotes, escolher instalar pacote(s), conforme figura.



Em seguida o menu com o local de onde será feito o download dos pacotes será iniciado. Escolha um país e clique em OK.



Um menu com os pacotes disponíveis será aberto, selecione spatstat e clique em OK.



Aguarde o Download e o Spatstat será instalado no R.

## 9.1.2 Carregando o Spatstat

O carregamento de pacotes no R pode ser feito por duas formas por comando ou pelo menu em Pacotes – Carregar Pacotes.

Em comando o procedimento é da seguinte maneira:

```
> require(spatstat)
```

### 9.1.3 Carregando os dados

Primeiramente a forma como os dados estão em arquivo deve ser observada.

Para trabalhar com padrões de pontos em duas dimensões os pontos devem estar representados por coordenadas (x,y) em uma matriz.

#### 9.1.3.1 Carregando os dados a partir de um diretório

Para extrair os dados no R a partir de um diretório o comando `read.table` é utilizado da seguinte maneira. Uma variável no caso C1 irá receber os dados lidos, conforme o comando abaixo:

```
>C1<-read.table("E:\\esppresa0.dat")
```

Note que o diretório em que os dados se encontram deve ser colocado entre “” e deve ser utilizado contra barras duplas.

Esse procedimento deve ser repetido para todo o conjunto de dados para cada distribuição de tempo. No caso foi repetido o processo até t6.

### 9.1.4 Criando uma janela de observação bi-dimensional

O comando para criar a janela é o `owin` e deve ser executado da seguinte maneira.

```
>win<-owin(c(0,100),c(0,100))
```

Onde uma variável (win) receberá a janela que no caso tem 100x100 unidades de medida.

### **9.1.5 Transformando as coordenadas em padrão de pontos (point patterns)**

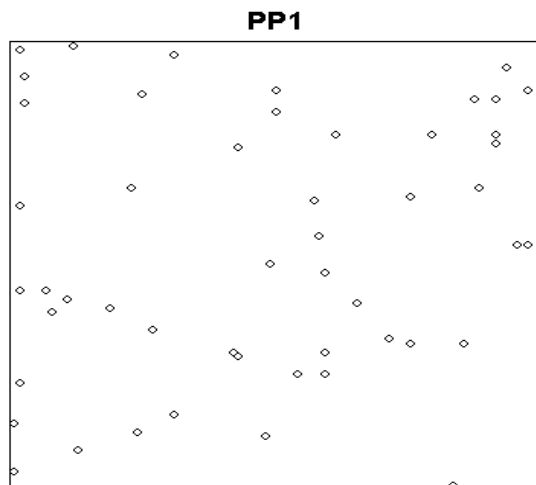
O comando para obter os point patterns é o ppp e uma variável deverá receber o comando ppp. Onde deve ser passados por parametros as coordenadas x e y de C1 e a janela criada anteriormente. Conforme abaixo:

```
PP1<-ppp(C1[1:50,1],C1[1:50,2], window=win)
```

Para visualizar os pontos no espaço basta dar o commando plot.

```
>plot(PP1)
```

O seguinte gráfico será gerado:



### 9.1.6 A função K

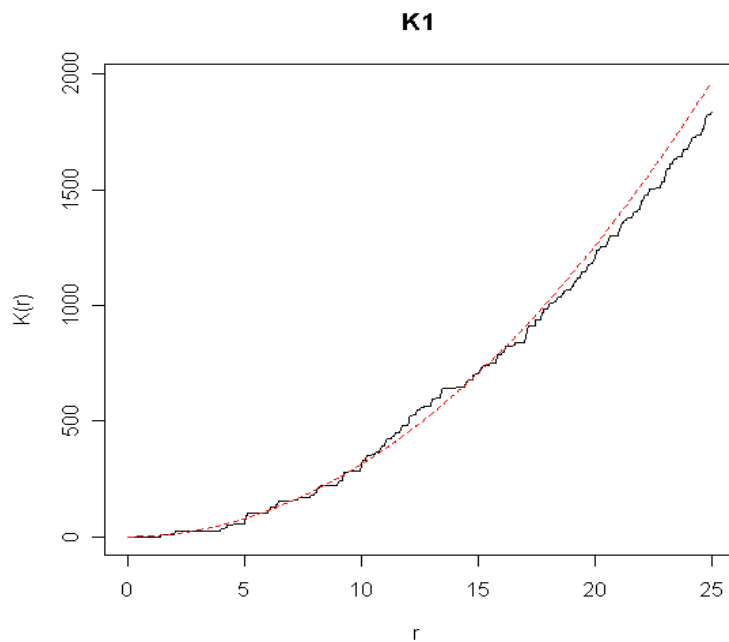
Tendo já o padrão de pontos para executar a função K basta chamar a função `kest` no R. Para extrair a correção de Ripley que é o nosso interesse deve ser passado o parâmetro `correction="Ripley"` para obter *Ripley isotropic correction estimate of  $K(r)$* .

```
>K1<-Kest(PP1, correction="Ripley")
```

Para observar o gráfico da função K basta utilizar o comando `plot`.

```
>plot(K1)
```

O gráfico abaixo será gerado onde a curva tracejada vermelha é o resultado de  $K$  teórica para a hipótese nula e a curva negra é a  $K$  observada.



Esses passos foram realizados para toda a distribuição nos intervalos de tempo de 1 a 6.

## 9.2 Implementação da função KF no Scilab

Abaixo a descrição de cada procedimento na função.

Primeiramente é efetuado um checagem para verificar se os valores de  $s$ ,  $t$  e



ranges foram passados e se estão corretos. Conforme abaixo:

```
if exists('s')==0 then printf("ERROR. Spatial scales, (s) not
defined"); abort; end
if exists('t')==0 then printf("ERROR. Temporal scales, (t) not
defined"); abort; end
if exists('ranges')==0 then printf("ERROR. Limits of x, y, z
(ranges) not defined"); abort; end
if size(xyz_data(1,:))<>[1 3] then printf("ERROR. xyz_data must
contain three columns"); abort; end
if min(xyz_data(:,1))<ranges(1,1) then printf("ERROR. Some
xyz_data fall outside ranges"); abort; end
if max(xyz_data(:,1))>ranges(2,1) then printf("ERROR. Some
xyz_data fall outside ranges"); abort; end
if min(xyz_data(:,2))<ranges(1,2) then printf("ERROR. Some
xyz_data fall outside ranges"); abort; end
if max(xyz_data(:,2))>ranges(2,2) then printf("ERROR. Some
xyz_data fall outside ranges"); abort; end
if min(xyz_data(:,3))<ranges(1,3) then printf("ERROR. Some
xyz_data fall outside ranges"); abort; end
if max(xyz_data(:,3))>ranges(2,3) then printf("ERROR. Some
xyz_data fall outside ranges"); abort; end
```

Em seguida é definido e criado os vetores de saída.

```
n=length(xyz_data(:,3));
Ks(:,1)=s; Ks(:,2)=zeros(s);
Kt(:,1)=t; Kt(:,2)=zeros(t);
for i=1:length(s)*length(t)
    Kst(i,1)=s(ceil(i/length(t)));
    Kst(i,2)=t(modulo(i-1,length(t))+1);
    Kst(i,3)=0;
End
```

Em seguida um loop duplo para examinar cada par de caso, ij, por vez.

```
for i=1:n
    printf("Doing i = "+sci2exp(i)+" of "+sci2exp(n)+"\n");
    for j=1:n
        // Primeiro, calcula dij, tij, wij e vij:
        xi=xyz_data(i,1);
        xj=xyz_data(j,1);
        yi=xyz_data(i,2);
```

```

yj=xyz_data(j,2);
zi=xyz_data(i,3);
zj=xyz_data(j,3);
dij = sqrt((xi-xj)^2+(yi-yj)^2);
tij = abs(zi-zj);

```

Calcular Wij requer essa preparação:

```

real_grid_count=0;
for xk=ranges(1,1):ranges(2,1)
  for yk=ranges(1,2):ranges(2,2)
    distance=sqrt((xi-xk)^2+(yi-yk)^2);
    if distance==dij then
real_grid_count=real_grid_count+1; end
    end
  end
infinite_grid_count=0;
roundup_smax=ceil(max(s(:,1)));
for xk=ranges(1,1)-roundup_smax:ranges(2,1)+roundup_smax
  for yk=ranges(1,2)-roundup_smax:ranges(2,2)+roundup_smax
    distance=sqrt((xi-xk)^2+(yi-yk)^2);
    if distance==dij then
infinite_grid_count=infinite_grid_count+1; end
    end
  end
if i==j then wij=0;
else wij= infinite_grid_count/real_grid_count;
end

```

Calculo de Vij.

```

if i==j then vij=0;
elseif zi+abs(zi-zj)<=ranges(2,3) & zi-abs(zi-
zj)>=ranges(1,3) then vij=1;
else vij=2;
end

```

Agora pega os valores do vetor coluna s e adiciona os valores de Ks apropriados para cada valor de s.

```

for k=1:length(Ks(:,1))
  if dij==Ks(k,1) then

```

```

Ks(k,2)=Ks(k,2)+wij*(ranges(2,1)-ranges(1,1)+1)*(ranges(2,2)-
ranges(1,2)+1)/(n*(n-1)); end
end

```

O mesmo procedimento acima é realizado para o vetor coluna t, adicionando os valores de Kt apropriados para cada valor de t.

```

for k=1:length(Kt(:,1))
    if tij==Kt(k,1) then
Kt(k,2)=Kt(k,2)+vij*(ranges(2,3)-ranges(1,3)+1)/(n*(n-1)); end
end

```

Para cada valor de s e t agora é adicionado um valor de Kst apropriado.

```

for k=1:length(Kst(:,1))
    if dij==Kst(k,1)&tij==Kst(k,2) then
        Kst(k,3)=Kst(k,3)+wij*vij*(ranges(2,1)-
ranges(1,1)+1)*(ranges(2,2)-ranges(1,2)+1)*(ranges(2,3)-
ranges(1,3)+1)/(n*(n-1));
    end
end

```

Em seguida é feito o calculo de Dst e D0st.

```

for k=1:length(Ks(:,1))
    for l=1:length(Kt(:,1))
        // D(s,t) goes in the fourth column:
        Kst((k-1)*length(Kt(:,1))+1,4)=Kst((k-
1)*length(Kt(:,1))+1,3)-Ks(k,2)*Kt(l,2);
        // D0(s,t) goes in the fifth column:
        Kst((k-1)*length(Kt(:,1))+1,5)=Kst((k-
1)*length(Kt(:,1))+1,4)/(Ks(k,2)*Kt(l,2));
    end
end

```

O código abaixo foi implementado para plotagem dos resultados:

```

clf()
D0st_matrix=matrix(Kst(:,5),length(Kt(:,1)),length(Ks(:,1)))';
contour(Ks(:,1),Kt(:,1),D0st_matrix,10,xtitle("", "spatial
scale", "temporal scale"))
a=get("current_axes");
a.sub_ticks=[0,0];

```

## 9.2.1 Rodando a função KF

Considerando que a função KF esteja salva no diretório D, o procedimento para carregar ela no Scilab é simples, basta utilizar o comando `getf`, como abaixo, a função KF está em um arquivo .txt com o nome KF.txt:

```
getf("D:\KF.txt", "n");
```

Para carregar a base de dados é necessário chamar o comando `fscanfMat`. Os dados estão salvos com o nome de RHBVdata.txt

```
RHBVdata=fscanfMat("D:\RHBVdata.txt");
```

Após esses comandos a função e a base de dados já estão carregadas no Scilab. Agora os valores de s, t e ranges devem ser passados como no exemplo:

```

s=[0;1;sqrt(2);2;sqrt(5)];
t=[0;1;2;3;4];
ranges=[1 1 0;10 10 48];

```

Para rodar a função KF deve ser passado como parametro a base de dados.

```
KF(RHBVdata);
```

Agora a função está sendo calculada.

```
Doing i = 1 of 715
```

```
Doing i = 2 of 715 (...)
```

```
Doing i = 715 of 715
```

Ela termina e é gerada uma matriz com os resultados.

```
Completed after 1527.469 seconds
```

```
ans =
```

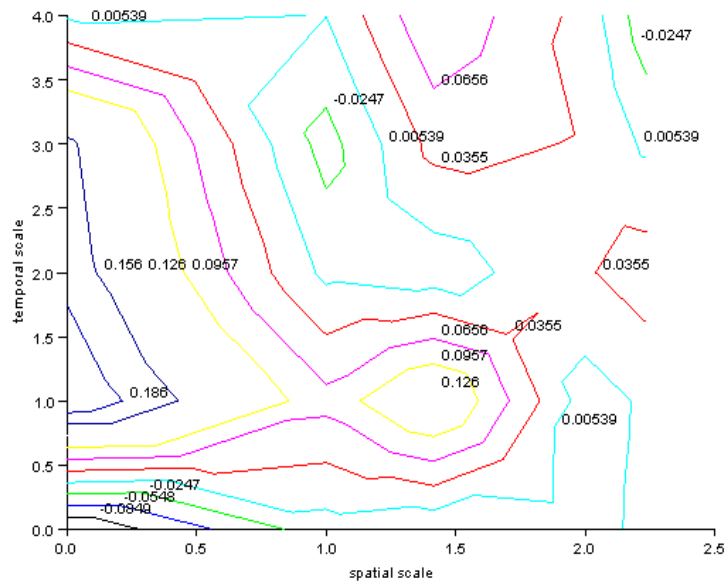
0.	0.	1.881256	- 0.2445485	- 0.1150381
0.	1.	5.0006856	0.8887748	0.2161464
0.	2.	4.3959962	0.6578292	0.1759764
0.	3.	4.0792541	0.5741119	0.1637913
0.	4.	3.1002331	0.0042697	0.0013791
1.	0.	6.7731615	- 0.0528380	- 0.0077407
1.	1.	14.202203	0.9987777	0.0756454
1.	2.	11.97861	- 0.0247171	- 0.0020592
1.	3.	10.839618	- 0.4154617	- 0.0369133
1.	4.	9.9981718	0.0569730	0.0057310
1.4142136	0.	5.6629645	- 0.1029650	- 0.0178575
1.4142136	1.	12.708076	1.555128	0.1394365
1.4142136	2.	10.010969	- 0.1282542	- 0.0126493
1.4142136	3.	9.9341835	0.4270051	0.0449140
1.4142136	4.	9.1759221	0.7785803	0.0927175
2.	0.	6.0085013	- 0.0567292	- 0.0093532
2.	1.	11.626674	- 0.1052075	- 0.0089677
2.	2.	11.009187	0.3436510	0.0322207
2.	3.	10.33411	0.3334282	0.0333405
2.	4.	9.0543443	0.2211088	0.0250315
2.236068	0.	10.737237	0.1558136	0.0147252
2.236068	1.	20.677819	0.2103361	0.0102766
2.236068	2.	19.570821	0.9636888	0.0517914

2.236068	3.	17.44641	- 0.0008164	- 0.0000468
2.236068	4.	14.701312	- 0.7091829	- 0.0460195

Observe que a tabela que aparece acima é a saída "Kst", contendo (da esquerda para a direita) s, t, KST, DST e D0st. Para salvar está e as outras saídas Ks e Kt como objetos no Scilab, você tem que especificá-los como saídas da função, assim que a chamar a função:

```
[Kst Ks Kt] = KF (RHBVdata);
```

A plotagem do gráfico da matriz:



## 9.2.2

```

function [Kst, Ks, Kt] = KF(xyz_data)
tic();
// I check for the existence of s, t and ranges, and that xyz_data fall within
ranges:
if exists('s')==0 then printf("ERROR. Spatial scales, (s) not defined"); abort;
end
if exists('t')==0 then printf("ERROR. Temporal scales, (t) not defined"); abort;
end
if exists('ranges')==0 then printf("ERROR. Limits of x, y, z (ranges) not
defined"); abort; end
if size(xyz_data(1,:))<>[1 3] then printf("ERROR. xyz_data must contain
three columns"); abort; end
if min(xyz_data(:,1))<ranges(1,1) then printf("ERROR. Some xyz_data fall
outside ranges"); abort; end
if max(xyz_data(:,1))>ranges(2,1) then printf("ERROR. Some xyz_data fall
outside ranges"); abort; end
if min(xyz_data(:,2))<ranges(1,2) then printf("ERROR. Some xyz_data fall
outside ranges"); abort; end
if max(xyz_data(:,2))>ranges(2,2) then printf("ERROR. Some xyz_data fall
outside ranges"); abort; end
if min(xyz_data(:,3))<ranges(1,3) then printf("ERROR. Some xyz_data fall
outside ranges"); abort; end
if max(xyz_data(:,3))>ranges(2,3) then printf("ERROR. Some xyz_data fall
outside ranges"); abort; end

// I define n, and create output vectors:
n=length(xyz_data(:,3));
Ks(:,1)=s; Ks(:,2)=zeros(s);
Kt(:,1)=t; Kt(:,2)=zeros(t);
for i=1:length(s)*length(t)

```

```

Kst(i,1)=s(ceil(i/length(t)));
Kst(i,2)=t(modulo(i-1,length(t))+1);
Kst(i,3)=0;
end

// I set up a double loop, to consider each pair of cases, ij, in turn:
for i=1:n
printf("Doing i = "+sci2exp(i)+" of "+sci2exp(n)+"\n");
for j=1:n
// First, I calculate dij, tij, wij and vij:
xi=xyz_data(i,1);
xj=xyz_data(j,1);
yi=xyz_data(i,2);
yj=xyz_data(j,2);
zi=xyz_data(i,3);
zj=xyz_data(j,3);
dij = sqrt((xi-xj)^2+(yi-yj)^2);
tij = abs(zi-zj);

// Calculation of wij requires some preparation:
real_grid_count=0;
for xk=ranges(1,1):ranges(2,1)
for yk=ranges(1,2):ranges(2,2)
distance=sqrt((xi-xk)^2+(yi-yk)^2);
if distance==dij then real_grid_count=real_grid_count+1; end
end
end
end

```



```

infinite_grid_count=0;
roundup_smax=ceil(max(s(:,1)));
for xk=ranges(1,1)-roundup_smax:ranges(2,1)+roundup_smax
  for yk=ranges(1,2)-roundup_smax:ranges(2,2)+roundup_smax
    distance=sqrt((xi-xk)^2+(yi-yk)^2);
    if distance==dij then
infinite_grid_count=infinite_grid_count+1; end
    end
  end
  if i==j then wij=0;
  else wij= infinite_grid_count/real_grid_count;
  end

// Calculation of vij is a little simpler:
if i==j then vij=0;
elseif zi+abs(zi-zj)<=ranges(2,3) & zi-abs(zi-zj)>=ranges(1,3) then
vij=1;
else vij=2;
end

// Then I take each s in turn, and add to the value of Ks if appropriate:
for k=1:length(Ks(:,1))
  if dij==Ks(k,1) then Ks(k,2)=Ks(k,2)+wij*(ranges(2,1)-
ranges(1,1)+1)*(ranges(2,2)-ranges(1,2)+1)/(n*(n-1)); end
  end

// Then I take each t in turn, and add to the value of Kt if appropriate:
for k=1:length(Kt(:,1))

```

```

        if tij==Kt(k,1) then Kt(k,2)=Kt(k,2)+vij*(ranges(2,3)-
ranges(1,3)+1)/(n*(n-1)); end
    end

    // Then I take each s and t in turn, and add to the value of Kst if
appropriate:
    for k=1:length(Kst(:,1))
        if dij==Kst(k,1)&tij==Kst(k,2) then
            Kst(k,3)=Kst(k,3)+wij*vij*(ranges(2,1)-
ranges(1,1)+1)*(ranges(2,2)-ranges(1,2)+1)*(ranges(2,3)-
ranges(1,3)+1)/(n*(n-1));
        end
    end

end

end

// Then I calculate Dst and D0st:
for k=1:length(Ks(:,1))
    for l=1:length(Kt(:,1))
        // D(s,t) goes in the fourth column:
        Kst((k-1)*length(Kt(:,1))+l,4)=Kst((k-1)*length(Kt(:,1))+l,3)-
Ks(k,2)*Kt(l,2);
        // D0(s,t) goes in the fifth column:
        Kst((k-1)*length(Kt(:,1))+l,5)=Kst((k-
1)*length(Kt(:,1))+l,4)/(Ks(k,2)*Kt(l,2));
    end
end

end

//Here is some optional code to plot the results:

```

```

clf()
D0st_matrix=matrix(Kst(:,5),length(Kt(:,1)),length(Ks(:,1)))';
contour(Ks(:,1),Kt(:,1),D0st_matrix,10,xtitle("", "spatial scale", "temporal
scale"))
a=get("current_axes");
a.sub_ticks=[0,0];
//Altering the tick labels only seems to work if you don't want to change the
number of them...
//a.x_ticks.locations=Ks(:,1);
//for i=1:length(Ks(:,1)) xlab(i)=sci2exp(Ks(i,1)); end
//a.x_ticks.labels=xlab;

printf("Completed after "+sci2exp(toc())+" seconds\n")
endfunction

```

## 9.3 Slides



## Roteiro

- 1 INTRODUÇÃO
  - Contextualização
  - Objetivos
- 2 REFERENCIAL TEÓRICO
- 3 METODOLOGIA
- 4 MODELAGEM E DESENVOLVIMENTO
- 5 RESULTADOS E DISCUSSÃO
- 6 CONCLUSÕES
- 7 PROPOSTAS FUTURAS
- 8 REFERENCIAL BIBLIOGRÁFICO



## Contextualização

- Populações Biológicas
  - Coleção de indivíduos de uma mesma espécie
  - Compartilham temporariamente um mesmo habitat
  - Relação não apresenta continuidade no espaço e no tempo
- Modelagem e Simulação de Populações
  - Configurações espaciais no tempo (Carneiro et. al., 2007)



## Contextualização

- **Comportamento das Configurações Espaciais**
  - Agrupamento
  - Regularidade
  - Aleatoriedade
- **A Função K (Ripley, 1977)**
  - Diggle et. al. (1995) propõe para analisar interação espaço-tempo em processos pontuais
  - Adaptação do método proposta por Carlsake et. al. (2005)



## Objetivos

- **Objetivo Geral**
  - Análise de processos pontuais em modelos de dinâmica de populações
- **Objetivo Especifico**
  - Implementação dos métodos estatísticos baseados na função K



## REFERENCIAL TEÓRICO

- **Estatística Espacial**

- Eventos de padrões espaciais (processos pontuais)

- ✓ São dados observados em um período de tempo contínuo ou discreto.



## REFERENCIAL TEÓRICO

- **A Função K (Ripley, 1977)**

- K teórica:

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



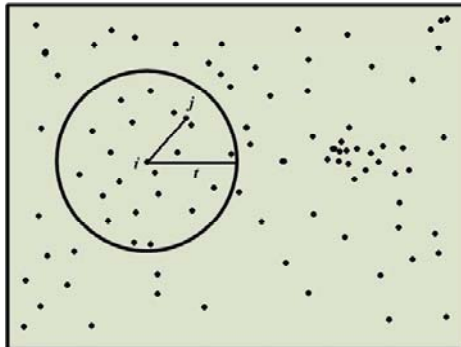
## REFERENCIAL TEÓRICO

$$\hat{K}(t) = \frac{|A|}{n^2} \sum_{i=1}^n \sum_{j=1}^n \frac{I_t(u_{ij})}{w_{ij}}$$

$u_{ij}$  é a distancia entre os eventos  $i$  e  $j$  ( $i \neq j$ ),  $I_t(u_{ij})$  é uma função indicadora igual a 1 quando  $u_{ij}$  é menor que a distancia  $t$  e 0, caso contrario,  $n$  é o número de evento no mapa com área  $A$  e  $w_{ij}$  é um fator de correção.



## REFERENCIAL TEÓRICO



- Raio  $t$ , centro  $i$  e evento  $j$ .



## REFERENCIAL TEÓRICO

- **Análise Espaço-Temporal**
  - . **Estatística de agrupamentos proposta por Carlslake et al. (2005)**
  - . **Estatística K(s,t)**

$$K(s,t) = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n I_s(d_{ij}) I_t(t_{ij}) w_{ij} v_{ij} R_s R_t$$



## REFERENCIAL TEÓRICO

- **Análise Espaço-Temporal**
  - . **Estatística para agrupamento no tempo:**

$$K(t) = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n I_t(t_{ij}) v_{ij} R_t$$

- . **Estatística para agrupamento no espaço:**

$$K(s) = \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n I_s(d_{ij}) w_{ij} R_s$$





## REFERENCIAL TEÓRICO

- **Análise Espaço-Temporal**

.Diggle et. al. (1995), processos espaciais e temporais operam independentemente

Então, Ausência de interação espaço-temporal :

$$K(s,t) = K(s)K(t)$$

Se  $K(s,t) > K(s)K(t)$ , então existe interação espaço tempo positiva.



## REFERENCIAL TEÓRICO

- **Análise Espaço-Temporal**

. Carslake et. al. (2005):

$$D(s,t) = K(s,t) - K(s)K(t)$$

$$D0(s,t) = D(s,t)/K(s)K(t)$$

*D(s,t) é o cluster adicional no espaço-tempo e D0(s,t) é o aumento proporcional do cluster no espaço-tempo que é devido a interação espaço-tempo.*



## METODOLOGIA

- Delineamento da pesquisa (Jung ,2004; Marconi & Lakatos, 2003)
  - Natureza tecnológica
  - Objetivo descritiva
  - Realizada em laboratório
- Procedimento da pesquisa:
  - Pesquisas bibliográfica e documental
  - Levantamento do conjunto de dados (pontos)
  - Análise individual dos padrões de pontos
  - Análise Espaço-temporal (uso de “armadilhas”)
  - Comparação entre métodos



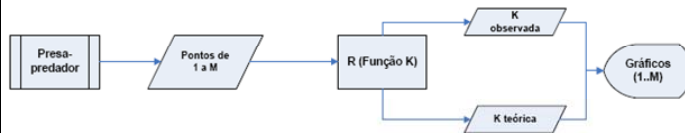
## MODELAGEM E DESENVOLVIMENTO

- Modelo presa-predador (Carneiro et. al. (2007) tipo IBM com autômato celular (*lattice*)
  - População fixa de plantas
  - População de herbívoros (pode se difundir através do sistema)
  - População de predadores
  - Tempo discretizado em intervalos periódicos (geração da população)
- Coleção de dados
  - Dados do programa Presa-predador
  - Extrair os pontos capturados pela “armadilha”



## MODELAGEM E DESENVOLVIMENTO

- A Função K no R  
- Pacote Spatstat

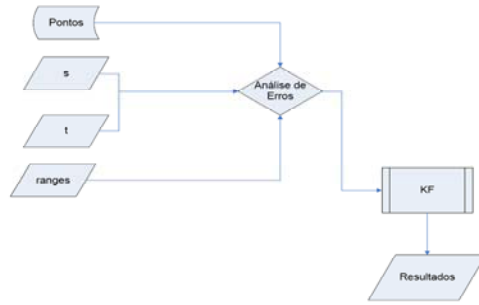


## MODELAGEM E DESENVOLVIMENTO

- A Função KF (Estatística espaço-tempo)
- Implementação da adaptação proposta por Carslake et. al. (2005)
- Software Scilab – interpretador de linguagem de programação

## MODELAGEM E DESENVOLVIMENTO

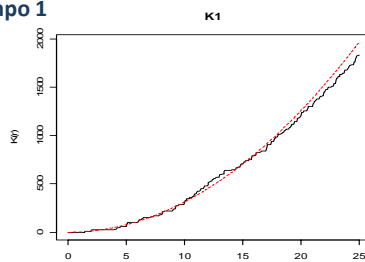
### • A Função KF



## RESULTADOS E DISCUSSÕES

### • Análise Espacial

- Tempo 1

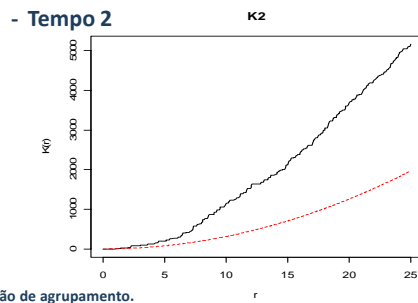


- Padrão identifica regularidade na distribuição da população.



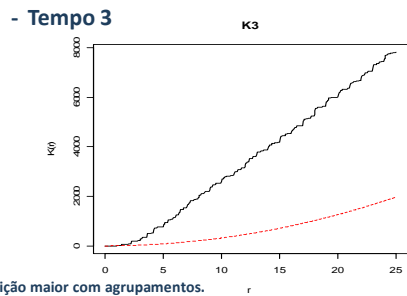
## RESULTADOS E DISCUSSÕES

### • Análise Espacial



## RESULTADOS E DISCUSSÕES

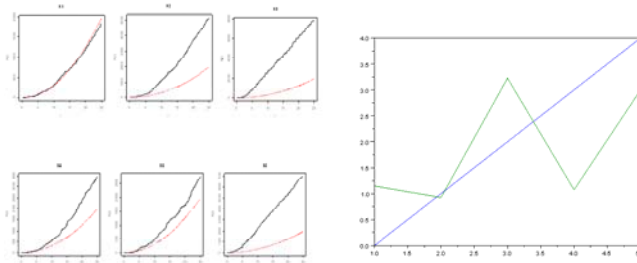
### • Análise Espacial





## RESULTADOS E DISCUSSÕES

### •Comparação dos Métodos



- Pico no tempo 3



## CONCLUSÕES

### •Geral

Resultados satisfatórios, considerando os objetivos levantados.

### •Análise Espacial

Acompanhamento da dinâmica da população no espaço no decorrer do tempo.

### •Análise Espaço-temporal

Foi possível visualizar a interação espaço-temporal



## PROPOSTAS FUTURAS

- Análise de diferentes espécies (predador).
- Implementação em uma linguagem de programação mais robusta como c++.
- Uso das funções em outras áreas do conhecimento.



## REFERENCIAL BIBLIOGRÁFICO

- CARNEIRO, M. V., CHARRET, I. C. (2007). Spontaneous emergence of spatial patterns in a predator-prey model. *To appear*.
- CARSLAKE, D., BENNETT, M., BROWN, K., HAZEL, S., TELFER, S., BEGON, M. (2005). Space time clustering of cowpox virus infection in wild rodent populations. *Journal of Animal Ecology* 74: 647-655.
- DIGGLE, P. J.; CHETWYND, A. G.; HAGGKIVIST, R., MORRIS, S. E. (1995). Second order analysis of space-time clustering. *Statistical Methods in Medical Research* 4: 124:136.
- JUNG, C. F. (2004). *Metodologia para pesquisa & desenvolvimento: aplicada a novas tecnologias, produtos e processos*. Rio de Janeiro/RJ: Axcel Books do Brasil Editora, 2004
- LAKATOS, E.V.; MARCONI, M. de A.. *Metodologia científica*. São Paulo/SP: Editora Atlas. 1988.
- RIPLEY, B. D. (1977). Modelling Spatial Patterns (with discussion). *Journal of the Royal Statistical Society B* 39: 172 - 212.





## Perguntas

