Introdução ao **Rcitrus**

Elias T. Krainski & Paulo J. Ribeiro Jr

Última Atualização: 4 de agosto de 2006

Este trabalho descreve a funcionalidade do **Rcitrus**, *software* para análise de dados de incidência de doenças em plantas. O **Rcitrus** foi desenvolvido em estrutura de pacote adicional da linguagem R. Seu desenvolvimento foi motivado pela necessidade de automatizar a análise estatística de dados de morte súbita dos citrus. Foram implementados alguns procedimentos para análise do padrão espacial da incidência de doenças em um talhão. Para a análise por *quadrat count*, foram implementados os modelos de Poisson, binomial e beta-binomial, a lei de Taylor e dois procedimentos de seleção dos *quadrats*: sistemático ou aleatório. Para a análise por processos pontuais, foram adaptadas a técnica de suavização por kernel (simples e razão), a análise por vizinhos próximos e a função K de Ripley. Para a modelagem, foi implementado o modelo autologístico com inferência utilizando bootstrap via amostrador de Gibbs e o método de Monte Carlo. Também foram implementados cinco métodos para simulação de dados binários com dependência espacial. As funcionalidades do **Rcitrus** são demonstradas com dados de morte súbita dos citros.

1 Introdução

O desenvolvimento deste trabalho foi motivado pela necessidade de automatização de análise de padrões espaciais da morte súbita dos citros (MSC).

A MCC é uma doença nova que provoca rápido definhamento de laranjeiras. O primeiro registro oficial da doença foi realizado em fevereiro de 2001 no município de Comendador Gomes, estado de Minas Gerais. A incidência de MSC avançou rapidamente, atingindo pomares do estado de São Paulo em 2002 (Bassanezi, Fernandes & Yammamoto 2003).

Vários trabalhos têm sido conduzidos buscando a compreensão dos mecanismos de propagação e dinâmica da doença. Tais trabalhos abrangem a coleta e análise de dados epidemiológicos provenientes de avaliações feitas em diferentes momentos em talhões de plantas de Citrus.

Nesse contexto, o estudo dos padrões espaciais é uma ferramenta muito útil, fornecendo informações quantitativas sobre o padrão espacial da incidência, utilizadas para analisar o mecanismo de propagação da doença.

O pacote **Rcitrus** está disponível para *download* em `http://www.est.ufpr.br/Rcitrus`. Para instalar o **Rcitrus** é necessário ter o R instalado.

Para instalar o **Rcitrus** em Windows, basta estar conectado à Internet e executar o seguinte comando em R:

```
> install.packages("Rcitrus",contrib="http://www.est.ufpr.br/Rcitrus/windows")
```

Em Linux o comando é:

```
> install.packages("Rcitrus",contrib="http://www.est.ufpr.br/Rcitrus")
```

Após instalado no computador, o pacote é carregado no R com o comando:

2 Dados de doenças de plantas em R

Os dados de doenças de plantas, são comumente armazenados em planilhas, em que cada linha corresponde às linhas de plantas nos talhões e cada coluna corresponde às plantas nas linhas. Além disso, mais de uma avaliação pode ter sido feita. Definimos algumas classes para representação em R, mas existem outras classes em outros pacotes de estatística espacial, tais como **geoR** (Ribeiro Jr. & Diggle 2001), **splancs** (Rowlingson, Diggle, adapted, packaged for R by Roger Bivand, pcp functions by Giovanni Petris & goodness of fit by Stephen Eglen 2006) ou **sp** (Pebesma & Bivand 2005).

Uma funcionalidade implementada, foi a importação de dados de arquivos texto:

```
> dat1 <- read.citrus("vv303.csv", nrow = 20, row.id = 1, n.att = 15,
+     sep = ";")
> dat1
```

```
Disease plant data in 25 evaluations of
20 rows of plants and 48 plants in each row.
```

2.1 Validação de dados

Alguns procedimentos fizeram-se necessários:

1. Troca utilizando a função `change.code()`:

```
> table(dat2 <- change.code(dat1, ori = c("O", "G"), mod = c(0,
+     0)))
```

```
    0     1     2     3     F     R
11101  6897  1363  4292    94   253
```

2. Seleção utilizando a funcão `select.code()`:

```
> table(dat3 <- select.code(dat2, unselect = c("F", "R")))
```

```
    0     1     2     3
11079  6897  1363  4286
```

3. Validação temporal, função `validStatusMSC.citrus()`:

```
> table(dat4 <- validStatusMSC.citrus(dat3, corr = TRUE))
```

```
9  inconsistences in  25 evaluations of 945 plants.

    0     1     2     3
11079  6897  1363  4286
```

2.2 Descrição de dados de doenças em plantas

Método `summary()` para as três primeiras avaliações:

```
> summary(dat4, eval = 1:3)
```

```
        01/08/2001 08/08/2001 16/08/2001
Min.       0.00000    0.00000     0.0000
1st Qu.    0.00000    0.00000     0.0000
Median     0.00000    0.00000     0.0000
Mean       0.03175    0.03492     0.1175
3rd Qu.    0.00000    0.00000     0.0000
Max.       3.00000    3.00000     3.0000
NA's      15.00000   15.00000    15.0000
```

O método `table()`:

```
> table.citrus(dat4, eval = 1:3)
```

```
  01/08/2001 08/08/2001 16/08/2001
0        922        919        872
1         17         20         45
2          5          5         18
3          1          1         10
```

O método `table()`, usando proporções:

```
> table.citrus(dat4, eval = 1:3, type = "p")
```

```
   01/08/2001  08/08/2001 16/08/2001
0 0.975661376 0.972486772 0.92275132
1 0.017989418 0.021164021 0.04761905
2 0.005291005 0.005291005 0.01904762
3 0.001058201 0.001058201 0.01058201
```

O método `plot()` produz um mapa para uma avaliação especificada.

```
> par(mfrow = c(1, 3), mar = c(2, 2, 2, 0.1), mgp = c(1, 0.3,
+     0))
> for (i in 1:3) plot(dat4, eval = i, pch = 19, main = paste("Avaliação",
+     i))
```

O método `lines()`:

```
> par(mar = c(2.5, 2.5, 0.5, 0.5), mgp = c(1, 0.5, 0))
> lines(dat4)
```

3 Análise por *quadrat counts*

Foram implemenados os modelos de Poisson, binomial e beta-binomial, (Madden & Hughes 1995).

```
> disp.quadrats(dat4, dx = 5, dy = 9, death = 1:3, eval = 1:3)
```

```
$`5x9`
     n  N  nN       p obs.var theor.var   index p.value  pattern
Av1 45 13 585 0.03590 0.00120   0.00077 1.55585 0.09680   Random
Av2 45 13 585 0.04103 0.00188   0.00087 2.15074 0.01142 Agregate
Av3 45 13 585 0.08889 0.00560   0.00180 3.10976 0.00020 Agregate
```

A mesma análise, utilizando o modelo de Poisson:

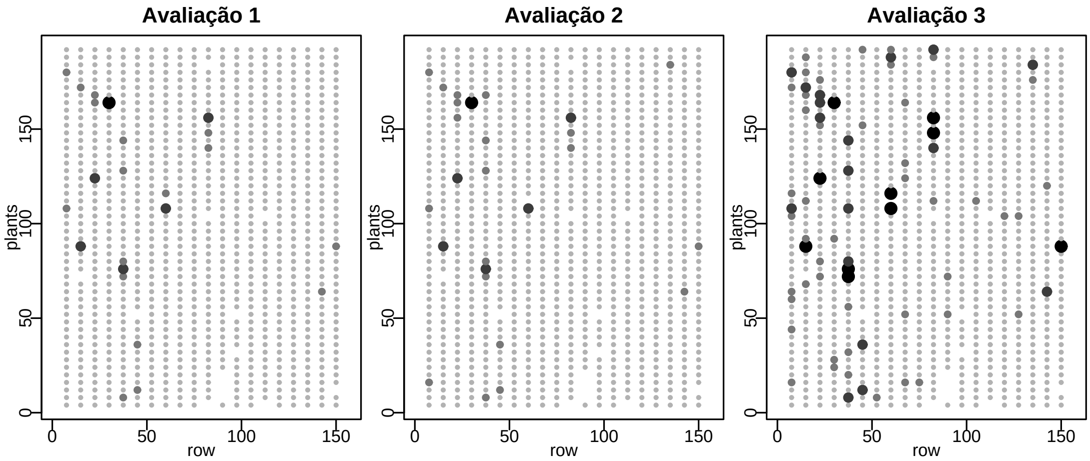

Figura 1: Mapa para o *status* das plantas doentes nas três primeiras avaliações.

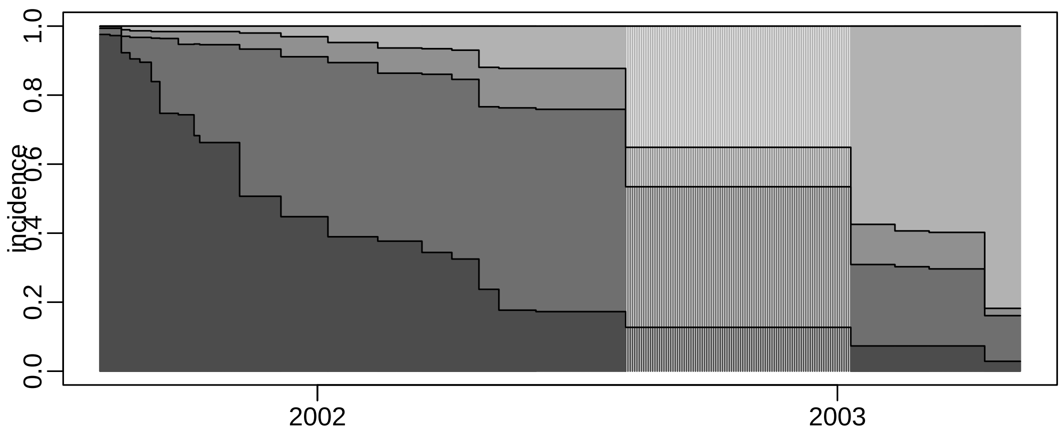

Figura 2: Linhas cumulativas do *status* da doença

```
> disp.quadrats(dat4, dx = 5, dy = 9, death = 1:3, eval = 1:3,
+     model = "Pois")

$`5x9`
     n  np       p  obs.var theor.var   index p.value  pattern
Av1 20 886 0.02596  2.02921   1.16591 1.74045 0.02361 Agregate
Av2 20 886 0.02822  3.14510   1.26862 2.47915 0.00035 Agregate
Av3 20 886 0.07336 10.30405   3.28668 3.13509 0.00000 Agregate
```

No modelo beta-binomial, o p-valor retornado é referente ao teste da hipótese $H_0 : \theta = 0$ (parâmetro de agregação), utilizando o teste da razão de verossimilhanças.

```
> disp.quadrats(dat4, dx = 5, dy = 9, death = 1:3, eval = 1:3,
+     model = "beta")

$`5x9`
     N    n  nN    prob   theta p.value  pattern
Av1 20 44.3 886 0.02582 0.01579 0.22173   Random
Av2 20 44.3 886 0.02801 0.03185 0.03846 Agregate
Av3 20 44.3 886 0.07290 0.04943 0.00235 Agregate
```

A estimação dos parâmetros da distribuição beta-binomial não é feita de forma analítica, mas utilizando algoritmo de minimização numérica. Utilizando a função `betabinom.citrus()`, pode-se explorar mais detalhes da estimação, inclusive as verossimilhanças perfilhadas.

3.1 Lei de Taylor

O ajuste da Lei de Taylor pode ser visualizado em um gráfico da reta, utilizando-se o método `plot()` implementado, e também em forma de um sumário:

```
> summary(Taylor.citrus(dat4, dx = 5, dy = 9, death = 1:3))

Summary of disease incidence:
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0359  0.2701  0.6137  0.5368  0.8308  0.9761
Estimates and confidence intervals of Taylor Law:
     2.5 % estimate   97.5 %
a 3.432306 4.447081 5.461857
b 1.313381 1.482636 1.651890
Thue an evidences of an agregatedpattern.
```

4 Análise dos vizinhos próximos

Essa idéia é amplamente utilizada em análise de processos pontuais. Aqui foram feitas adaptações para considerar a natureza dos dados de incidência de doenças em plantas.

4.1 Distância mínima média

O conjunto de dados disponíveis é formado por N plantas, em que y plantas estão doentes. O teste para a análise do padrão espacial utilizando a distância mínima média consiste em três passos:

1. calcular a distância mínima média para os dados observados;
2. sortear aleatóriamente as y plantas doentes nas N posições;

3. calcular a distância mínima média;
4. repetir o passo 2 e 3 anterior s vezes;
5. calcular o valor p.

É razoável assumir que dados com padrão agregado terão distâncias mínimas médias menores que dados com padrão aleatório.

```
> mmdt <- mmdist.test(dat4, death = 1:3, eval = 1:3, NMC = 199)

test evaluation: 1 2 3

> summary(mmdt)

Results for 199 Monte Carlo simulations!
Observed: 11.65319 14.96830 9.472777
Randoms:
           e1    e2     e3
Min.    12.97 12.26  9.234
1st Qu. 18.48 16.99 10.610
Median  20.08 18.14 10.960
Mean    19.88 18.33 10.960
3rd Qu. 21.49 19.65 11.380
Max.    26.07 23.49 12.980
P-value: 0.005 0.03 0.02
```

Pode-se também fazer o histograma e o plot das distâncias (Figura 3) usando os comandos:

```
> par(mfrow = c(2, 3), mar = c(2, 2, 2, 0.1), mgp = c(1.2,
+     0.5, 0))
> hist(mmdt, main = "Distäncia Mínima", evaluation = 1:3)
> plot(mmdt, main = "Distäncia Mínima", evaluation = 1:3)
```

4.2 Número médio de vizinhos doentes

O número de vizinhos doentes é o número de plantas doentes dentro de um raio em torno de cada planta doente. O número médio de vizinhos doentes é a média do número de vizinhos doentes para cada planta doente.

Uma atenção deve ser dada ao efeito de borda que se tem neste caso. Adaptamos a função `khat()` do pacote **splancs** que incorpora a correção de borda.

```
> neigh.test(dat4, death = 1:3, eval = 1:3, NMC = 199)

test evaluation: 1 2 3
Results for 199 Monte Carlo simulations!
P-value: 0.015 0.005 0.005
```

5 Análise de processos pontuais

Os métodos abordados até o momento levam em conta todas as plantas na análise. Na análise de processos pontuais estuda-se a ocorrência de eventos no espaço, considerando a localização dos eventos como aleatória. Vários métodos de processos pontuais estão implementados no pacote **splancs**. Alguns foram adaptados para a análise de doenças de plantas.

Os detalhes podem ser vistos utilizando-se o método `summary()` e a visualização gráfica pode ser visualizada pelo método `plot()`.

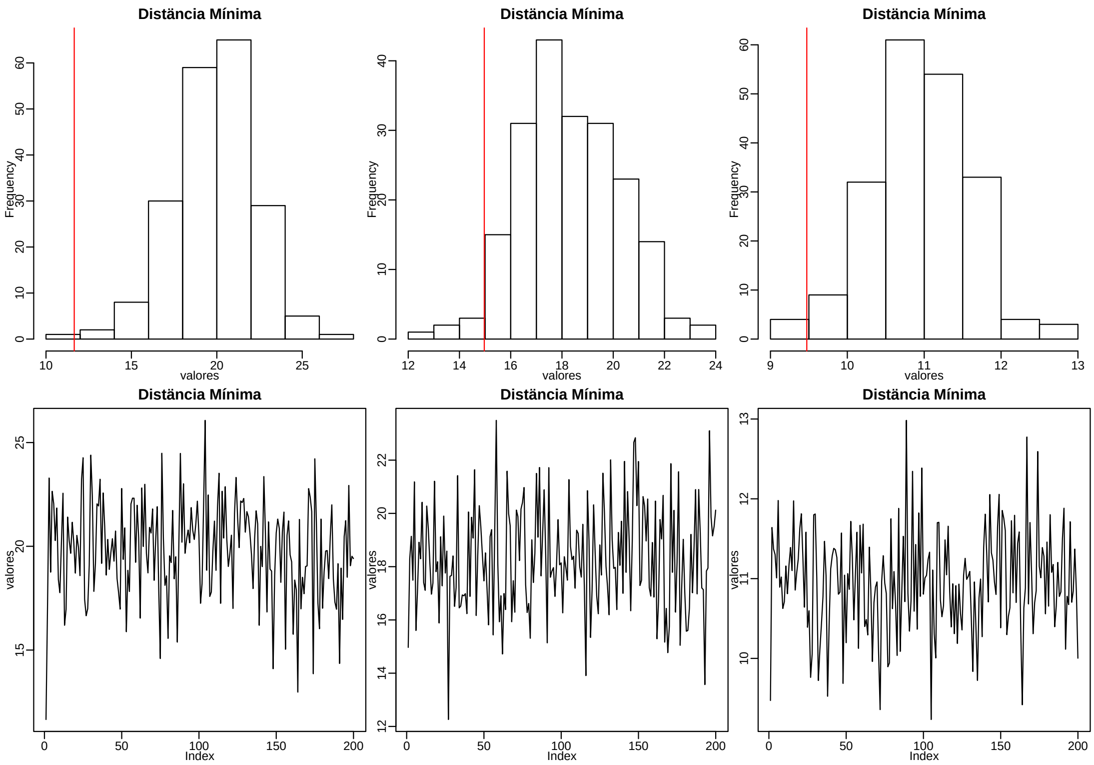

Figura 3: Vizualização dos resultados do teste de Monte Carlo para a distância mínima média.

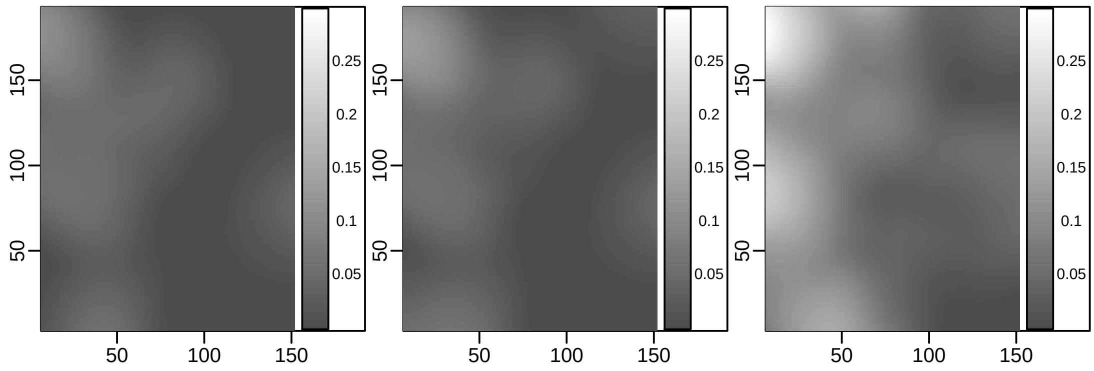

Figura 4: Mapas da proporção de plantas doentes para as três primeiras avaliações, utilizando função quartica e escala de cores global

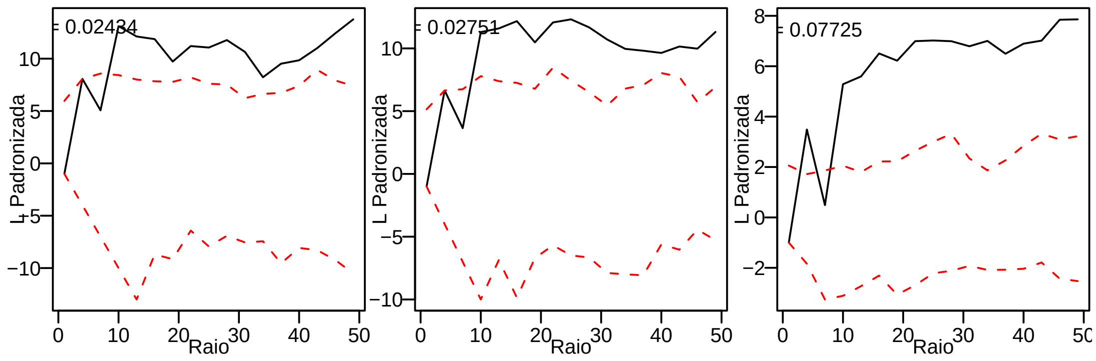

Figura 5: Envelope simulado (linhas tracejadas) para a função K de Ripley observada (linha contínua) para as plantas doentes nas três primeiras avaliações

5.1 Suavização por kernel

A técnica de suavização é uma análise exploratória que visa a construção de mapas de intensidade do processo. Consiste em estimar o número esperado de eventos por unidade de área. Adicionalmente, podemos ter o mapa de suavização considerando todas as plantas e utilizá-lo para construir um mapa da proporção de plantas doentes, o mapa de razão de kernel.

5.2 Função K de Ripley

O padrão espacial de eventos está muito associado à escala na qual fazemos a análise. A função K é uma ferramenta que permite analisar o padrão espacial em diferentes escalas. Essa função é uma medida da propriedade de segunda ordem do processo.

A conclusão sob o padrão da doença pode ser feita analisando o gráfico da função estimada e padronizada em função do raio, juntamente com o envelope simulado sob padrão aleatório.

A linha contínua é a função K estimada para os dados e as linhas tracejadas são o envelope simulado sob padrão aleatório. Observa-se que o padrão é agregado, para raios maiores que 10 metros.

6 Modelo autologístico

Além de avaliar o padrão espacial, podemos estar interessados em modelar a probabilidade de ocorrência da doença e considerar o *status* das plantas vizinhas como covariadas. Esse modelo é denominado modelo de regressão autologística. A autocorrelação é evidentemente induzida, pois a mesma informação é utilizada como resposta e covariada (Besag 1972).

No modelo autologístico, pode-se considerar diferentes estruturas de vizinhança. É interessante avaliar o efeito de plantas vizinhas na linha, vizinhas em linhas adjacente e vizinhas nas diagonais separadamente, para buscar possíveis efeitos direcionais, pois cada coeficiente dá uma estimativa do acréscimo na probabilidade da presença ou não da doença nesses vizinhos (Gumpertz & Ristaino 1997). Esta estrutura de vizinhança é particularmente interessante para o caso de doenças em citros, onde o espaçamento entre linhas é diferente do espaçamento dentro das linhas.

Na função `autologistic.citrus()` foi implementado o modelo autologístico com estimação dos parâmetros pelo método da pseudo-verossimilhança, bem como o procedimento de reamostragem bootstrap via algoritmo amostrador de Gibbs e o método de Monte Carlo. Nesta função, pode-se considerar as covariadas de vizinhança separadamente, considerar interações entre as covariadas e considerar as covariadas de vizinhança no tempo anterior. Para exemplificar, vamos ajustar o modelo para as observações da terceira avaliação e considerar as covariadas de vizinhança na segunda observação:

```
> mod <- autologistic.citrus(dat4[, , 3], obj2 = dat4[, , 2],
+     death = 1:3, verbose = FALSE)
> summary(mod)$coef
```

```
              Estimate Std. Error     z value     Pr(>|z|)
(Intercept) -2.9606421  0.1923434 -15.3924788 1.838520e-53
R            0.8828508  0.2753265   3.2065601 1.343323e-03
C           -0.1496822  0.4010516  -0.3732243 7.089815e-01
dA           0.2343161  0.3554166   0.6592717 5.097213e-01
dB           0.8577925  0.2994055   2.8649857 4.170281e-03
```

Aplicando o método de Monte Carlo:

```
> (mc.mod <- autologistic.citrus(dat4[, , 3], obj2 = dat4[,
+     , 2], death = 1:3, inf.method = "mc", N = 299, verbose = FALSE))
```

```
OK!
Results of pseudo-likelihood
Coeficients:
(Intercept)           R           C          dA          dB
 -2.9606421   0.8828508  -0.1496822   0.2343161   0.8577925
Variances:
(Intercept)           R           C          dA          dB
 0.03699599  0.07580466  0.16084240  0.12632098  0.08964366
P-values for random pattern
witch Monte Carlo method(Intercept)           R           C          dA          dB
  0.0100000   0.1066667   0.7500000   0.6366667   0.1400000
```

7 Simulando dados com padrão espacial

Um procedimento importante na estatística moderna é a simulação. Simular dados de um modelo ajustado é muito utilizado para avaliar o desempenho de metodologias de análise e na

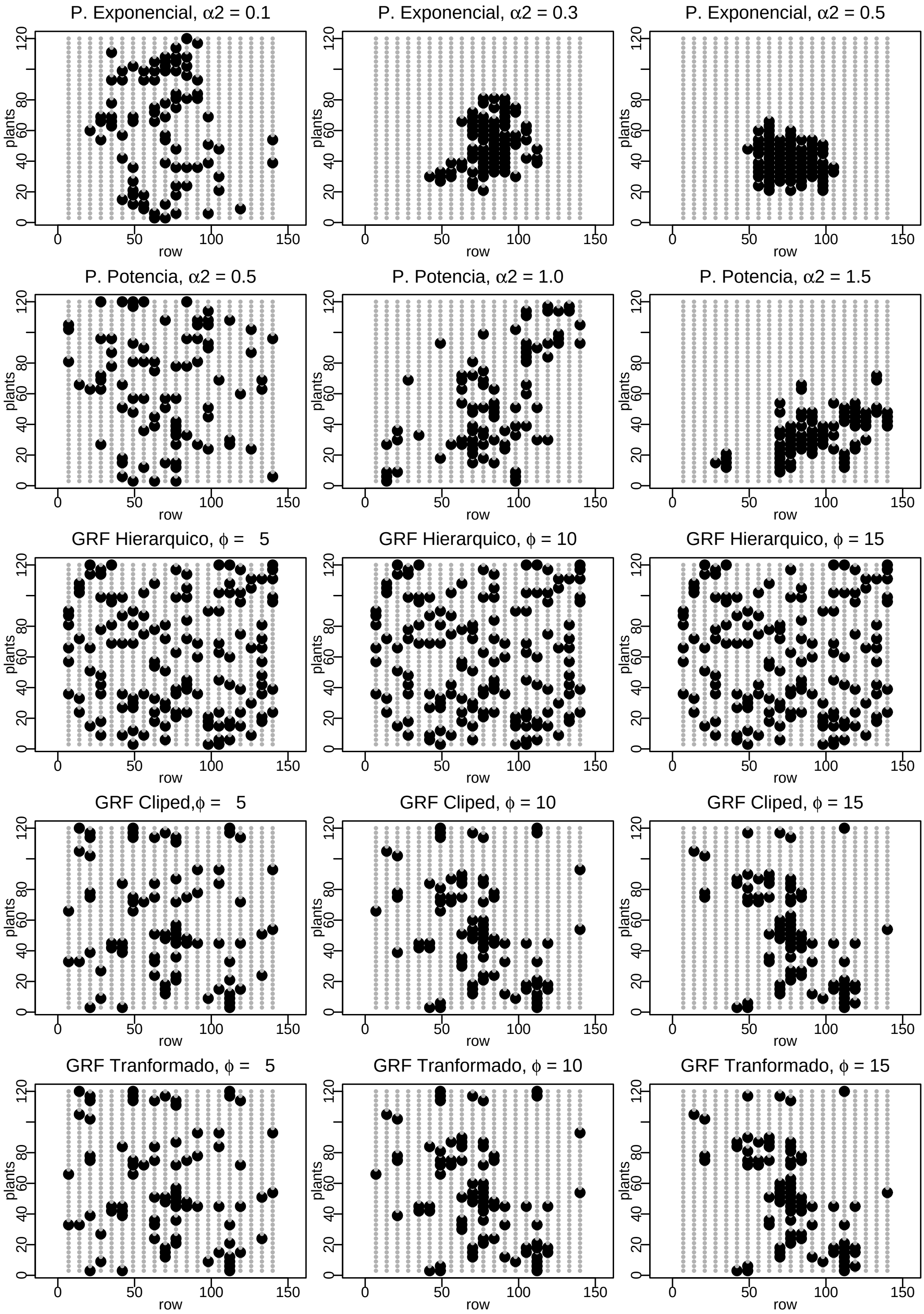

Figura 6: Vizualização de dados simulados segundo três diferentes modelos (colunas) com variação dos parâmetros em dada modelo (linhas)

verificação de pressupostos. Simular um conjunto de plantas doentes com padrão aleatório é bastante simples. Porém, quando se deseja simular um conjunto de dados com padrão não aleatório, temos infinitas possibilidades de padrões espaciais.

O modelo geoestatístico considera que os dados são uma realização de um processo estocástico multivariado. Toda a informação sobre a dependência espacial nesse modelo é dada pela matriz de covariância. Porém se temos apenas uma realização do processo, devemos modelar a matriz de covariância de forma a utilizar poucos parâmetros. Os modelos adotados para a covariância são funções da distância entre os pontos, em que as observações mais próximas são mais correlacionadas (Diggle & Ribeiro Jr. 2000).

A partir de um conjunto de dados de um processo gaussiano S com dependência espacial, podemos obter dados binários com dependência espacial. Foram considerados três procedimentos para simular dados binários utilizando o modelo geoestatístico: 1) Simular amostras bernoulli a partir dos valores gaussianos simulados, 2) truncar os valores gaussianos simulados ou 3) efetuar uma transformação quantílica.

Os métodos de simulação apresentados foram implementados na função `sim.citrus()`. Como exemplo, para cada um dos cinco métodos apresentados, vamos simular três conjunto de dados, cada um com parâmetros diferentes. Na Figura 6 observamos esses dados.

Agradecimentos

Este trabalho foi desenvolvido como parte das atividades do convênio firmado entre o Fundo de Defesa da Citricultura (FUNDECITRUS) e o Departamento de Estatística da Universidade Federal do Paraná e financiado pelo FUNDECITRUS.

Referências

Bassanezi, R. B., Fernandes, N. G. & Yammamoto, P. T. (2003). Morte súbita do citros, *Technical report*, Fundecitrus.

Besag, J. (1972). Nearest-neighbour systems and the auto-logistic model for binary data, *Journal of the Royal Statistics Society, Series B* **1**(34): 75–83.

Diggle, P. J. & Ribeiro Jr., P. J. (2000). *Model Based Geostatistics*, 1 edn, 14 SINAPE.

Gumpertz, M. L. ; Graham, J. M. & Ristaino, J. B. (1997). Autologistic model of spatial pattern of phytophthora epidemic in bell pepper: Effects of soil variables on disease presence, *Journal of Agricultural, Biological and Environmental Statistics* **2**(2): 131–156.

Madden, L. V. & Hughes, G. (1995). Plant disease incidence: Distributions, heterogeneity, and temporal analysis, *Phytopathology* **33**: 529–564.

Pebesma, E. J. & Bivand, R. S. (2005). Classes and methods for spatial data in R, *R News* **5**(2): 9–13.
*http://CRAN.R-project.org/doc/Rnews/

Ribeiro Jr., P. & Diggle, P. (2001). geoR: A package from geostatistical analysis, *R-NEWS* . URL: http://cran.R-project.org/doc/Rnews.

Rowlingson, B., Diggle, P., adapted, packaged for R by Roger Bivand, pcp functions by Giovanni Petris & goodness of fit by Stephen Eglen (2006). *splancs: Spatial and Space-Time Point Pattern Analysis*. R package version 2.01-17.
*http://www.maths.lancs.ac.uk/ rowlings/Splancs/