SKATER
Spatial **“K”**luster **A**nalysis **T**hrough **E**dge **R**emoval
Extensions, possible applications and computational implementation

Paulo Justiniano Ribeiro Jr

LEG:Laboratório de Estatística e Geoinformação / UFPR

In collaboration with:

Elias Teixeira Krainski (LEG/UFPR)

Renato Martins Assunção (LESTE/UFMG)

http://www.leg.ufpr.br
e-mail:paulojus@ufpr.br

CHICAS
Lancaster, UK
20 – 22 de Maio de 2010

IBC-2010/Floripa e 55 RBRAS

International Biometrics Conference &

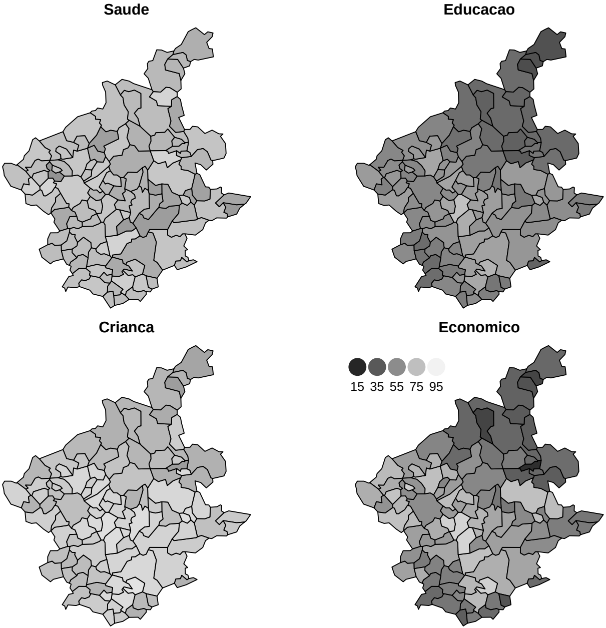
55ª **Reunião da Região Brasileira da Sociedade Internacional de Biometria**

1. Organization: IBS, Rbras, RArg
2. 05 a 10 december de 2010, Florianópolis, SC, Brasil
3. satelite events (opened to proposals)
4. Free/excursion day (and RBras/RArg meeting) on wednesday, 07/12
5. http://www.ibc-floripa-2010.org/ and http://www.tibs.org

Outline

- Motivatition
- Basics of SKATER
- Computational implementation
- Extensions and fundamentals
- Some applications
- Final remarks

Motivating Examples I: Living condition indexes in BH



Clustering and regionalization

- Clustering:
  - n individuals with their “atributes” $\longrightarrow$ k “homogeneous” groups
  - group composition (possibly with minimal number)
  - how many groups?
- Regionalization:
  - classification procedure
  - applied to spatial objects (with an areal representation)
  - groups them into homogeneous contiguous regions
- Why?
  - detecting heterogeneous sub-regions (“step”)
  - exploratory
  - administrative/management purposes
  - . . .
- reduced number of possible groups

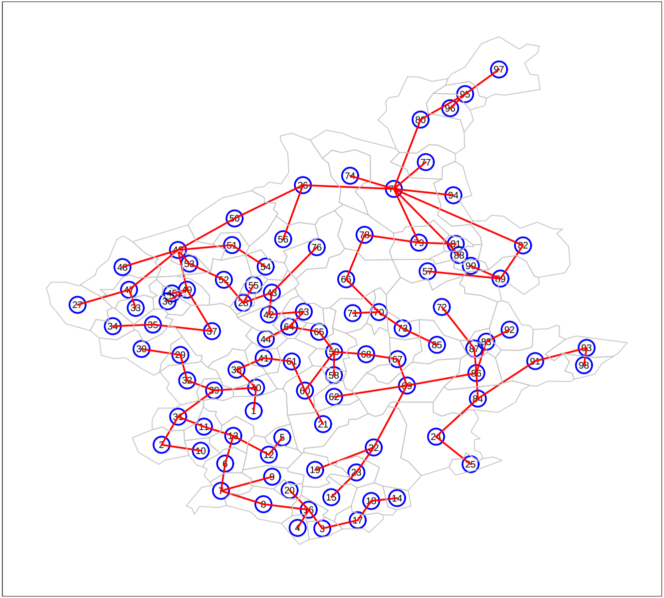
Approaches

1. non-spatial clustering + neighbouring preserving classification
   - 2 steps
   - possibly too many groups
2. clustering including (weighted) spatial covariates
   - SAGE (spatial analysis GIS environment)
   - objective function involves *homogeneity*, *compactness* and *equality*
3. explicit use of neighbouring structure in optimization
   - implicit constraints
   - AZP (automatic zonning procedure)
   - computationally expensive

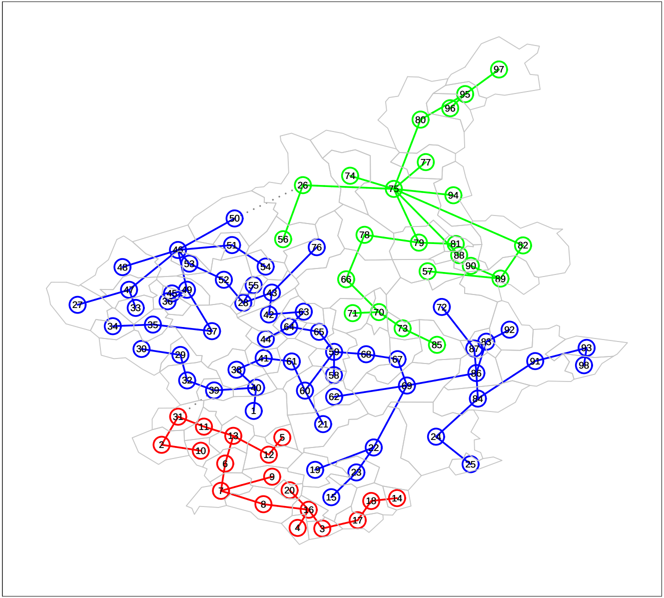
The SKATER approach

- algorithm of type (3)
- graph to introduce/represent neighbourhood
  - vertices (units) + edges
  - “cost” of edges: dissimilarities
- heuristics for prunning
  - minimal 1 group graph (MST: minimal spanning tree)
  - regionalization $\longrightarrow$ optimal graph partitioning

Skater in action I



Skater in action II



An "R"-ish template for the analysis

1. spatial packages: **sp**, **spdep**, . . .
2. Read attributes and the map
   `read.table()` and `readShapePoly()`
3. standardize data (`scale()`)
4. neighbourhood list (`poly2nb()`)
5. costs for edges (dissimilarities) (`nbcosts()`)
6. weighted neighbourhood structure (`nb2listw()`)
7. minimum spanning tree (`mstree()`)
8. partition (`skater()`)

Yet another example

Extending SKATER

- costs of edges
  - euclidean, mahalanobis, . . .
  - density based measurements (non-Gaussian data)
- measures of homogeneity
  - distance to the mean(s)
  - likelihood based measures, criteria for number of groups
- applications
  - areal data
  - point processes, geostatistical data, time series, independent observations

Extending SKATER

- costs of edges
  - euclidean, mahalanobis, . . .
  - density based measurements (non-Gaussian data)
- measures of homogeneity
  - distance to the mean(s)
  - likelihood based measures, criteria for number of groups
- applications
  - areal data
  - point processes, geostatistical data, time series, independent observations

Extending SKATER

- costs of edges
  - euclidean, mahalanobis, . . .
  - density based measurements (non-Gaussian data)
- measures of homogeneity
  - distance to the mean(s)
  - likelihood based measures, criteria for number of groups
- applications
  - areal data
  - point processes, geostatistical data, time series, independent observations

Extending SKATER

- costs of edges
  - euclidean, mahalanobis, . . .
  - density based measurements (non-Gaussian data)
- measures of homogeneity
  - distance to the mean(s)
  - likelihood based measures, criteria for number of groups
- applications
  - areal data
  - point processes, geostatistical data, time series, independent observations

Minimum spanning tree

- connected graph (nodes v, edges e, attributes y) path between any pair (v_i, v_j)
- tree (no circuit)
- spanning tree n nodes $V = (v_1, \ldots, v_n)$ and $n-1$ edges $E = (e_1, \ldots, e_{n-1})$
- costs for each edge (dissimilarities) $d(v_i, v_j) = d(e_k) = \sum_l (y_{il} - y_{jl})^2$
- minimum spanning tree (MST): set $\{e_1, \ldots e_{n-1}\}$ minimizing $\sum_{k=1}^{n-1} d(e_k)$
- removal of an edge results in two graphs (also MSTs)
- Prim (1957) algorithm: start from an empty set and include a first node in V
    - compute dissimilarities between nodes V_{in} and V_{out}
    - select the node from V_{out} with minimum dissimilarity
    - iterate until all included
- unique under certain conditions

Minimum spanning tree

- Y with p.d.f $p(y/\theta,\phi)$, location (θ) and ϕ (scale)
- $pl_{\theta,\phi}(\theta,\phi/X,y) = \frac{L(\theta,\phi/X,y)}{max_{\theta',\phi'}[L(\theta',\phi'/X,y)]}$
- similarly for y : $pl_y(y/X,\theta,\phi) = \frac{p(y/X,\theta,\phi)}{max_{y'\in\Re}[p(y/X,\theta,\phi]}$
- distance measures
  - common scale: $d_{ij} = log\{[p(y_i/X_i,\theta,\phi)p(y_j/X_j,\theta,\phi)]^{-1}\}$
  - more general: $d_{ij} = log\{[pl_y(y_i/X_i,\theta,\phi)pl_y(y_j/X_j,\theta,\phi)]^{-1}\}$

Gaussian case

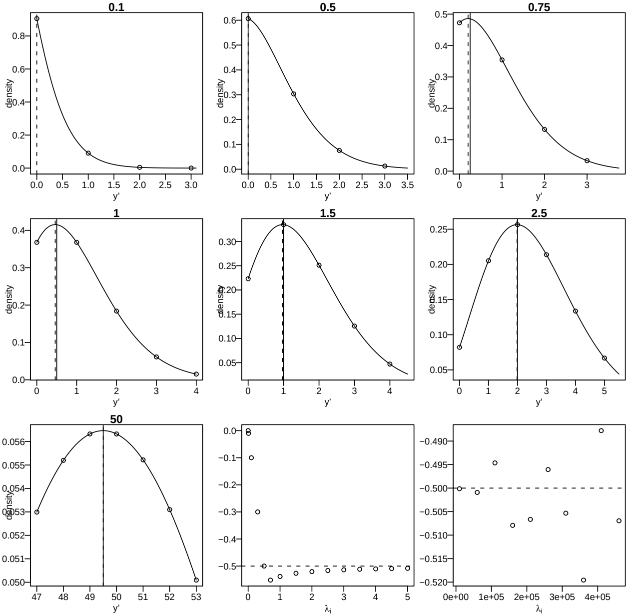
- $y_i \sim N(\mu_i, \sigma^2)$
  - $d(i,j) = log(2\pi\sigma^2) + \frac{1}{2\sigma^2}[(y_i - \mu_i)^2 + (y_j - \mu_j)^2]$
  - $\hat{\mu}_i = \hat{\mu}_j = (y_i + y_j)/2$
  - $d(i,j) = log(2\pi) + (y_i - y_j)^2$

- $y_i \sim N(\mu_i, \sigma_i^2)$.
  - $d(i,j) = log(2\pi\sigma_i^2) + \frac{1}{2\sigma_i^2}[(y_i - \mu_i)^2 + (y_j - \mu_j)^2]$
  - $\hat{\mu}_i = \hat{\mu}_j = (y_i + y_j)/2$ and $\hat{\sigma}_i^2 = \hat{\sigma}_j^2 = (y_i - y_j)^2/2$
  - $d(i,j) = log(\pi(y_i - y_j)^2)$

- $y_i \sim N(0, \sigma_i^2)$
  - $d(i,j) = \frac{1}{2}[log(2\pi\sigma_i^2) + log(2\pi\sigma_j^2)] + \frac{y_i^2}{2\sigma_i^2} + \frac{y_j^2}{2\sigma_j^2}$
  - $\hat{\sigma}_i^2 = \hat{\sigma}_j^2 = \hat{\sigma}_{ij}^2 = (y_i^2 + y_j^2)/2$
  - $d(i,j) = log(\pi(y_i - y_j)^2) + 1$

Poisson case

- Common parameter for mean an variance
- pl_y:
  - to compare y_k with y_i and y_j on different groups
  - y_i and y_j on the same group with different offset (e.g.population sizes) $\lambda_i = \theta_i \times o_i$
- $y_i \sim Poisson(\lambda_i)$ $p(y_i/\theta_i, o_i) = p(y_i/\lambda_i) = \lambda_i^{y_i} e^{-\lambda_i} / \prod y_i$
- $max_{y'}\{p(y'/\lambda_i)\} = max_{y'}\{\lambda_i^{y'} e^{-\lambda_i}/(y'!)$
- 0 for $\lambda_i < 0.5$ and $\approx \lambda_i - 0.5$ for $\lambda_i \geq 0.5$


0.1
0.5
0.75
1
1.5
2.5
50
density
y'
λi

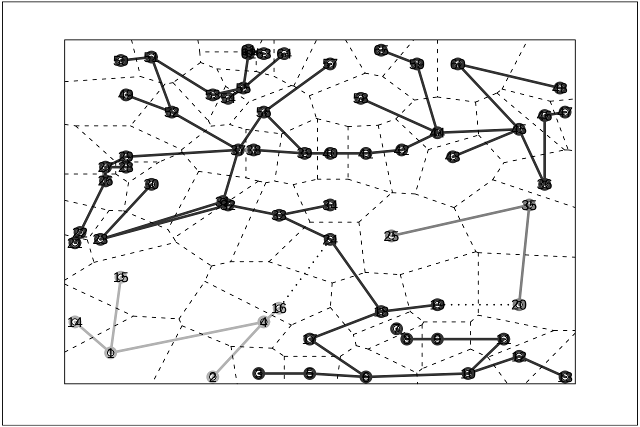
Partitioning the MST

- hierarquical division of MST (groups)
- For each edge k:
  - remove the edge
  - compute homogeneity (H_g) for the groups (e.g. $H_g = \sum_i \sum_l (y_{il} - \bar{y}_l)^2$)
  - remove edge minimizing $\sum_g H_g$
  - iterate
- objective function: $H_g - (H_{g1} + H_{g2})$
- stopping criteria: number of groups, minimum number within groups, reduction of $\sum_g H_g$, etc
- efficient algorithm (Assunção et. al. 2006)

Alternative Homogeneity measures

- likelihood based
- for group i: $H_i = -\sum_j log(p(y_j|\hat{\theta}_i, \hat{\phi}_i))$,
- for K groups: $\sum_{i=1}^{k} H_i$
- “deviance”: $D_k = H_k - H_{k-1}$
- $D_k \sim \chi_p^2$: stopping criteria

Poins proccess



R code

```
require(spdep); require(spatstat); data(japanesepines)
del = deldir(japanesepines, rw=c(0,1,0,1))
d.area = data.frame(del$summary)$dir.area
nb.del = tapply(c(del$dirs[,5], del$dirs[,6]), c(del$dirs[,6], del$dirs[,5]), as.integer)
class(nb.del) ¡- ”nb”
costs.a = nbcosts(nb.del, d.area)
nbw.a = nb2listw(nb.del, costs.a, style=”B”)
mst.a = mstree(nbw.a)
ldnorm = function(x, id) -sum(dnorm(x[id], mean(x[id]), sd(d.area), log=TRUE))
sk5.a = skater(mst.a[,1:2], d.area, 4, method=ldnorm)
```