

R-package demonstration: spatgraphs

Tuomas Rajala

University of Jyväskylä, Finland

Current version: 2.03

What?

`spatgraphs` = **spatial graphs**

What `spatgraphs` is:

- R-package for calculating graphs
- Designed for the needs of spatial point pattern analysis
- Free under GPL, available from CRAN

What?

`spatgraphs` = **spatial graphs**

What `spatgraphs` is:

- R-package for calculating graphs
- Designed for the needs of spatial point pattern analysis
- Free under GPL, available from CRAN

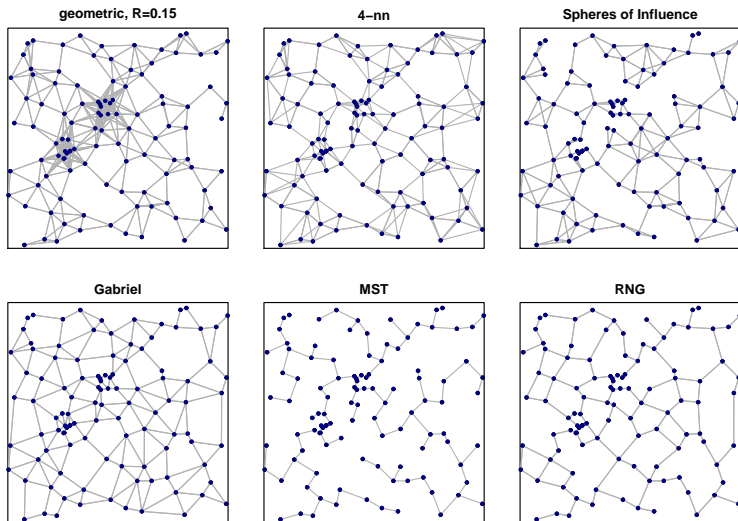
What `spatgraphs` is **not**:

- Package for analysing graphs / networks
- Package for analysing spatial point patterns

∴ The output of `spatgraphs` is a distillation of spatial point pattern features, useful for further analysis.

Example

```
> g<-spatgraph(pp, type="geometric", par=0.03)
> plot(pp) ; plot(g, pp)
```



Motivation

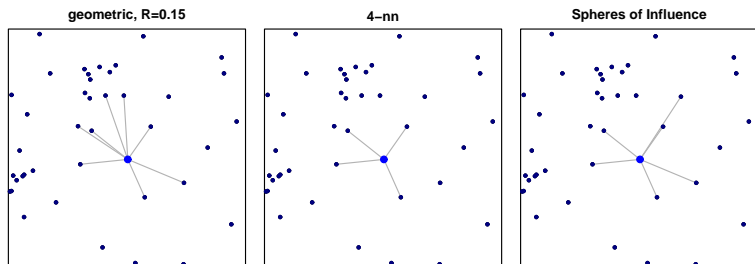
Why graphs?

- Gives a well defined and flexible definition for 'neighbourhood', 'neighbours', 'close by' etc.
- Additional structure $\stackrel{?}{=}$ closer to reality?
- Fast calculation of different 'neighbourhoods' allows novel ideas to be tested easily

Motivation

Why graphs?

- Gives a well defined and flexible definition for 'neighbourhood', 'neighbours', 'close by' etc.
- Additional structure $\stackrel{?}{=}$ closer to reality?
- Fast calculation of different 'neighbourhoods' allows novel ideas to be tested easily



Capabilities of `spatgraphs`:

- Graph types supported: Geometric, k -nearest neighbours, marked geometric, mark crossing, Spheres of Influence, Radial Spanning Tree, Minimal Spanning Tree, Relative Neighbourhood (2D), Gabriel, Delauney (2D), Class Cover Catch.¹
- Component calculation: find the path-connected components in a given graph
- Works also for 3D point patterns (with restriction)
- Export as a dxf-file (CAD/GIS)

¹In v2.1: Signal-to-noise-ratio, convex nearest neighbours.

Capabilities of spatgraphs:

- Graph types supported: Geometric, k -nearest neighbours, marked geometric, mark crossing, Spheres of Influence, Radial Spanning Tree, Minimal Spanning Tree, Relative Neighbourhood (2D), Gabriel, Delauney (2D), Class Cover Catch.¹
- Component calculation: find the path-connected components in a given graph
- Works also for 3D point patterns (with restriction)
- Export as a dxf-file (CAD/GIS)

Package:

- Code: R and C++
- Tools: Eclipse + StatET module
- All code written by the author

¹In v2.1: Signal-to-noise-ratio, convex nearest neighbours.

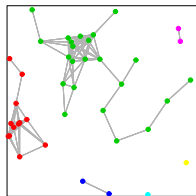
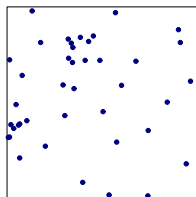
How?

The use of spatgraphs is simple:

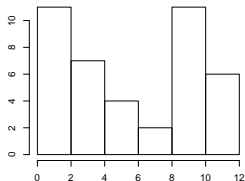
```
> plot(pp)
>
> g<-spatgraph(pp, "geometric", par=0.09)
> g
'Spatgraphs' edge connection list-of-lists:
graph type 'geometric', par=(0.09), for 41 points.
> g$edges
[[1]]
[1] 2 3 4 6 8 11
...
[[41]]
[1] 38

> plot(pp);plot(g,pp)
>
>
> ppc<-spatcluster(g)
> ppc
'Spatgraphs' cluster/component list-of-lists:
graph type 'geometric', par=(0.09), 6 components.

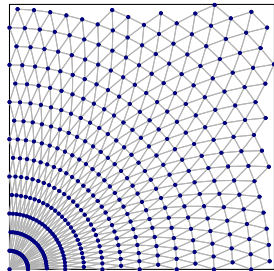
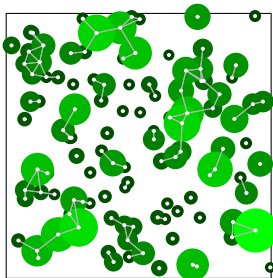
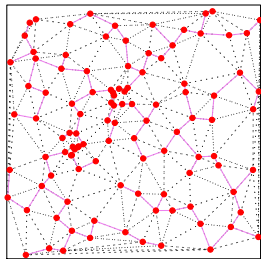
> plot(pp);plot(g,pp);plot(ppc,pp)
>
>
> neighs<-sapply(g$edges, length)
> hist(neighs)
```



Histogram of neighs



Download and give it a go!



R software network CRAN (google 'CRAN spatgraphs')
Author's homepage <http://tuomas.sokkelo.net/code/>

Current version (Wed 6.8.08): 2.03