



2 Published on California Soil Resource Lab (<http://casoilresource.lawr.ucdavis.edu/drupal>)

3 Point-process modelling with the sp and spatstat packages

4 By dylan

5 Created 2006-10-26 04:38

6 Some simple examples of importing spatial data from text files, converting between R datatype, creation of a point process model and evaluating the model. Input data sources are: soil pit locations with mollic and argillic diagnostic horizons (mollic-pedons.txt and argillic-pedons.txt), and a simplified outline of Pinnacles National Monument? [0] (pinn? [0].txt). The outline polygon is used to define a window in which all operations are conducted.

The 'sp' package for R contains the function `spsample()`, can be used to create a sampling plan for a given region of interest: i.e. the creation of n points within that region based on several algorithms. This example illustrates the creation of 50 sampling points within Pinnacles, according to the following criteria: regular (points are located on a regular grid), nonaligned (points are located on a non-aligned grid-like pattern), random (points are located at random), stratified [1] (collectively exhaustive, see details here [2]).

The 'spatstat' package for R contains several functions for creating point-process models: models describing the distribution of point events: i.e. the distribution of tree species within a forest. If covariate data is present (i.e. gridded precipitation, soil moisture, aspect, etc.) these covariates can be incorporated into the point-process model. Without covariate data, the model is based on an spatial distribution estimator function. Note that the development of such models is complicated by factors such as edge-effects, degree of stochasticity, spatial connectivity, and stationarity. **I am not a statistician, so please remember to read up [3] on any functions you plan to use for your own research.**

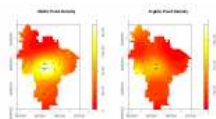
Helpful links

[Spatstat Quick Reference](#) [4]

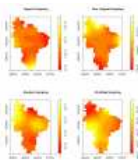
[Print Version with Links](#) [5]



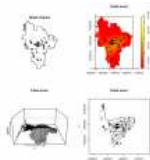
[5]Four sampling designs



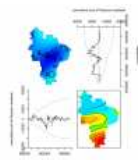
[5]Spatial density of each pedon type



[5]Spatial density of the four sampling designs



[5]Example point-process model of mollic soils



[5]Diagnostics of a simple point-process model

Load required packages and input data (see attached files at bottom of this page)

```
7 # load required packages
8 library(sp)
9 library(spatstat)
10
11 # read in pinnacles boundary polygon: x,y coordinates
12 # use the GRASS vector, as it should be topologically correct
13 # v.out.ascii format=standard in=pinn_bnd > pinn.txt ... edit out extra crud
14 pinn <- read.table('pinn.txt')
15
16 # read in mollic and argillic pedons
17 # see ogrinfo hack
18 m <- read.table('mollic-pedons.txt', col.names=c('x','y'))
19 a <- read.table('argillic-pedons.txt', col.names=c('x','y'))
```

```

20
21 # add a flag for the type of pedon
22 m$type <- factor('M')
23 a$type <- factor('A')
24
25 #combine into a single dataframe
26 pedons <- rbind.data.frame(a,m)

```

Using the functions from the 'sp' package create a polygon object from the pinn.txt coordinates

```

27 # create a spatial polygon class object
28 pinn.poly <- SpatialPolygons(list(Polygons(list(Polygon( pinn )), "x")))
29
30 # inspect this new object with str()
31 # total area of all polygons
32 pinn.poly@polygons[[1]]@area
33
34 # coordinates of first polygon: this is rough syntax!
35 pinn.poly@polygons[[1]]@Polygons[[1]]@coords

```

Generate a sampling plan for 50 sites using regular grid, non-aligned grid, random, and random stratified approaches

```

36 # generate random points within the pinnacled boundary
37 p.regular <- spsample(pinn.poly, n = 50, "regular")
38 p.nalign <- spsample(pinn.poly, n = 50, "nonaligned")
39 p.random <- spsample(pinn.poly, n = 50, "random")
40 p.stratified <- spsample(pinn.poly, n = 50, "stratified")
41
42 # setup plot environment
43 par(mfrow=c(2,2))
44
45 # each of the sampling designs
46 plot(pinn.poly)
47 title("Regular")
48 points(p.regular, pch=16, col='red', cex=0.3)
49
50 plot(pinn.poly)
51 title("Nonaligned")
52 points(p.nalign, pch=16, col='red', cex=0.3)
53
54 plot(pinn.poly)
55 title("Random")
56 points(p.random, pch=16, col='red', cex=0.3)
57
58 plot(pinn.poly)
59 title("Stratified")
60 points(p.stratified, pch=16, col='red', cex=0.3)

```

Convert 'sp' class objects to 'spatstat' analogues note the use of 'slots'

```

61 # pinn boundary:
62 # extract coordinates: and get a length - 1 value
63 p.temp <- pinn.poly@polygons[[1]]@Polygons[[1]]@coords
64 n <- length(p.temp[,1]) - 1
65
66 # create two vectors: x and y
67 # these will contain the reversed vertices, minus the last point
68 # in order to adhere to the spatstat specs: no repeating points, in counter-clockwise order
69 x <- rev(p.temp[,1][1:n])
70 y <- rev(p.temp[,2][1:n])
71
72 # make a list of coordinates: note that we are removing the last vertex
73 p.list <- list(x=y,y=y)
74
75 # make a spatstat window object from the polygon vertices
76 W <- owin(poly=p.list)
77
78 # pedons with their 'marks' i.e. pedon type, and the pinn boundary as the 'window'
79 pedons.ppp <- ppp(pedons$x, pedons$y, xrange=c(min(pedons$x), max(pedons$x)), yrange=c(min(pedons$y), max(pedons$y)), window=W,
marks=pedons$type)

```

Plot and summarize the new combined set of pedon data

```

80 # plot and summarize the pedons data:
81 # note the method used to subset the two 'marks'
82 par(mfrow=c(1,2))
83 plot(density.ppp(pedons.ppp[pedons.ppp$marks == 'M']), main="Mollic Point Density")
84 points(pedons.ppp[pedons.ppp$marks == 'M'], cex=0.2, pch=16)
85
86 plot(density.ppp(pedons.ppp[pedons.ppp$marks == 'A']), main="Argillic Point Density")
87 points(pedons.ppp[pedons.ppp$marks == 'A'], cex=0.2, pch=16)
88
89 summary(pedons.ppp)

```

```

90 Marked planar point pattern: 151 points
91 Average intensity 1.38e-06 points per unit area
92 Marks:
93 frequency proportion intensity
94 A 62 0.411 5.67e-07
95 M 89 0.589 8.14e-07
96
97 Window: polygonal boundary
98 single connected closed polygon with 309 vertices
99 enclosing rectangle: [ 657228.3125 , 670093.8125 ] x [ 4030772.75 , 4047986.25 ]
100 Window area = 109337135.585938

```

Convert the sampling design points (from above) to 'spatstat' objects and plot their density

```

101 # convert the random datasets: using the same window:
102 ppp.regular <- ppp(p.regular@coords[,1], p.regular@coords[,2], window=W)
103 ppp.nalign <- ppp(p.nalign@coords[,1], p.nalign@coords[,2], window=W)
104 ppp.random <- ppp(p.random@coords[,1], p.random@coords[,2], window=W)
105 ppp.stratified <- ppp(p.stratified@coords[,1], p.stratified@coords[,2], window=W)
106
107 # visually check density of random points:
108 par(mfrow=c(2,2))
109 plot(density.ppp(ppp.regular), main="Regular Sampling")
110 points(ppp.regular, pch=16, cex=0.2)
111
112 plot(density.ppp(ppp.nalign), main="Non-Aligned Sampling")
113 points(ppp.nalign, pch=16, cex=0.2)
114
115 plot(density.ppp(ppp.random), main="Random Sampling")
116 points(ppp.random, pch=16, cex=0.2)
117
118 plot(density.ppp(ppp.stratified), main="Stratified Sampling")
119 points(ppp.stratified, pch=16, cex=0.2)

```

Simple, and probably flawed attempt to use a point-process model for the pedon data Third order polynomial model for the distribution of pedons with a mollic epipedon. See manula page for ppm() for detailed examples.

```

120 # model the spatial occurrence of Mollic epipedons with a 3rd-order polynomial, using the Poisson Process Theory [6]
121 fit <- ppm( unmark(pedons.ppp[pedons.ppp$marks == 'M']), ~polynom(x,y,3), Poisson())
122
123 # view the fitted model
124 par(mfcol=c(2,2))
125 plot(unmark(pedons.ppp[pedons.ppp$marks == 'M']), main="Mollic Pedons")
126 plot(fit)
127
128 # plot some diagnostics on the fitted model: Pearson residuals (see references)
129 diagnose.ppm(fit, type="pearson")

```

```

130 #another example using a built-in dataset: the Lansing Forest
131 # fit nonstationary marked Poisson process
132 # with different log-cubic trend for each species
133 data(lansing)
134 fit <- ppm(lansing, ~ marks * polynom(x,y,3), Poisson())
135 plot(fit)

```

Point-process model diagnostic references from the spatstat manual

136 Baddeley, A., Turner, R., Moller, J. and Hazelton, M. (2005) Residual analysis for spatial point processes. Journal of the Royal Statistical Society, Series B 67, 617–666.
137 Stoyan, D. and Grabarnik, P. (1991) Second-order characteristics for stochastic structures connected with Gibbs point processes. Mathematische Nachrichten, 151:95–100.

< [Using ColorBrewer to assist with thematic map color selection](#) [6][up](#) [6][Time Series Experiments](#) > [6]

Attachment	Size
argillic-pedons.txt [7]	1.12 KB
mollic-pedons.txt [8]	1.6 KB
pinn.txt [9]	8.17 KB

Source URL:

<http://casoilresource.lawr.ucdavis.edu/drupal/drupal/node/319>

Links:

[1] http://en.wikipedia.org/wiki/Stratified_sampling
[2] http://en.wikipedia.org/wiki/Stratified_sampling
[3] <http://www.spatstat.org/spatstat/current/Quickref.pdf>
[4] <http://www.spatstat.org/spatstat/current/Quickref.pdf>
[5] <http://casoilresource.lawr.ucdavis.edu/drupal/node/319/print>
[6] http://en.wikipedia.org/wiki/Poisson_process#Spatial_Poisson_process
[7] <http://casoilresource.lawr.ucdavis.edu/drupal/files/argillic-pedons.txt>
[8] <http://casoilresource.lawr.ucdavis.edu/drupal/files/mollic-pedons.txt>
[9] <http://casoilresource.lawr.ucdavis.edu/drupal/files/pinn.txt>