Topic 2: Analysis of Noise Complaints

This topic discusses another example of spatial data and demonstrate some applications of the studied spatial point processes methods to urban data.

For this example we use the data from the website `www.kaggle.com/datasets/dbtjdals/noise-complaints-2018` . The data set was originally extracted from the New York Open Data Portal (https://opendata.cityofnewyork.us/) and include information from 2018.

By analysing these noise complaints, the NYC police and city administration can determine appropriate measures or alerts to implement, as well as evaluate the effectiveness of these measures over time.

In this brief analysis, we will explore some spatial aspects of the data, in particular, to determine which areas in NYC get the most noise complaints and how some of them change over time and related to each other.

The data set can be downloaded from the LMS folder Data as the file `Noise_Complaints.csv`.

We will use the following R libraries:

```
> library(sf)
> library(spatstat)
> library(lubridate)
> library(stpp)
> library(mapview)
```

First, we import the data in R:

```
> MyData <- read.csv("Noise_Complaints.csv", header = TRUE)
> str(MyData)
'data.frame': 436691 obs. of  39 variables:
...
$ Closed.Date : chr  "01/01/2018 01:05:25 PM"  ...
...
$ Descriptor : chr  "Loud Music/Party"
"Loud Music/Party" "Loud Music/Party" "Banging/Pounding" ...
$ Latitude: num  40.7 40.7 40.7 40.7 40.8 ...
$ Longitude: num  -73.9 -73.9 -73.8 -74 -73.8 ...
```

Then we subset the data using only the following attributes

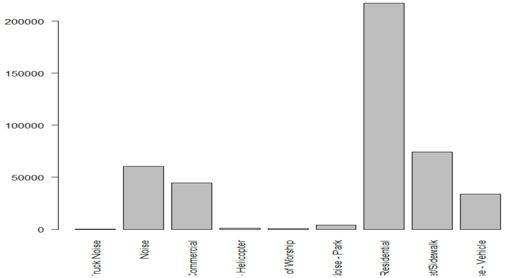
- `Created.Date` - The date that complaint was reported
- `Complaint.Type` - Type of complaint reported
- `Descriptor` - Description of the complaint
- `Latitude` - Latitude coordinate where it was reported
- `Longitude` - Longitude coordinate where it was reported

by the R command

```
> MyData1 <- MyData[, c( "Created.Date", "Complaint.Type",
+ "Descriptor",  "Latitude", "Longitude" )]
> str(MyData1)
'data.frame': 436691 obs. of  4 variables:
$ Created.Date  : chr  "01/01/2018 12:04:05 AM" ...
$ Complaint.Type: chr  "Noise - Residential" "Noise - Residential"
$ Latitude: num  40.7 40.7 40.7 40.7 40.8 ...
$ Longitude: num  -73.9 -73.9 -73.8 -74 -73.8 ...
```

To visualise information about different complaints we use the commands

```
> table(MyData1$Complaint.Type)
Collection Truck Noise    Noise   Noise - Commercial
 171                    60609                        44686
Noise - Helicopter Noise - House of Worship   Noise - Park
 1033                     753                          4237
Noise - Residential  Noise - Street/Sidewalk   Noise - Vehicle
217199                  74101                          33902
 > barplot(table(MyData1$Complaint.Type),  las = 2)
```



Then we remove the cases with missing information:

```
> MyData1 <- MyData1[complete.cases(MyData1), ]
```

Consider two types of complaints, noise by helicopters and noise recorded in NYC parks:

```
> MyData1 <- MyData1[MyData1$Complaint.Type %in%
+ c("Noise - Helicopter", "Noise - Park"), ]
```

Using the function `mdy_hms` we determine the fist time when such noise was recorded in 2018.

```
> time1 <- round_date(mdy_hms(MyData1$Created.Date[1]),
+ unit = "hour")
```

This time (in hours) we will use as an initial time reference moment for subsequent incidents.

Next, calculate the time duration in hours between two incidents for each of the following events with respect to the reference moment:

```
> MyData1$Created.Date <- round_date(mdy_hms(MyData1$Created.Date),
+ unit = "hour")
> MyData1$hours_diff <- as.numeric(difftime(MyData1$Created.Date,
+ time1, units = "hours"))
```

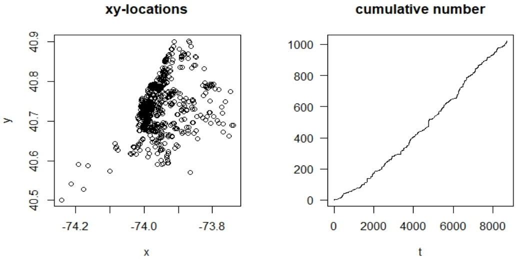
Analysis of complains about noise produced by helicopters

First, we restrict our data only to complains about noise produced by helicopters. From all fields we select only geographic coordinates and time of incidents after the first complain in 2018 (in hours), i.e. `MyData1$hours_diff`:

```
> MyData1_Helic <-  MyData1[MyData1$Complaint.Type ==
+ "Noise - Helicopter", ]
> df1 <-  data.frame(x = MyData1_Helic$Longitude,
+ y = MyData1_Helic$Latitude, t = MyData1_Helic$hours_diff )
```

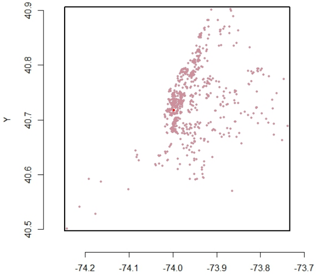
To visualise these spatial data over time by using stpp format, we transform them as

```
> X1 <- as.3dpoints(df1)
> str(X1)
'stpp' num [1:1021, 1:3] -73.9 -73.8 -74 -74 -74 ...
- attr(*, "dimnames")=List of 2
..$ : NULL
..$ : chr [1:3] "x" "y" "t"
> plot(X1)
```



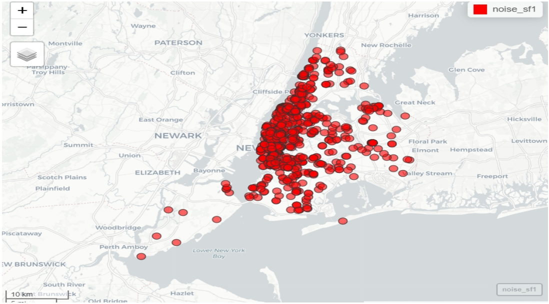
The plot suggests that data are not spatially uniformly distributed over the area, but as the cumulative plot is close to a strait line it means that the rate over time is almost constant.
To visualise their spatial-temporal occurrence we will use the following R code for animation:

```
> dev.new()
> animation(X1, runtime = 20)
> dev.off()
```

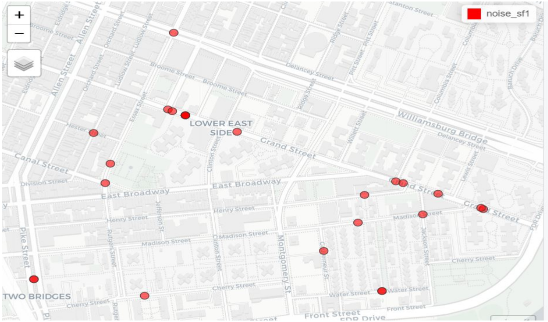


Finally, we create a sf object with locations of incidents and visualise it on the NYC map:

```
> noise_sf1 <- st_as_sf(df1[, 1:2], coords = c("x", "y"))
> str(noise_sf1)
Classes sf and 'data.frame': 1021 obs. of  1 variable:
$ geometry:sfc_POINT of length 1021;...
- attr(*, "sf_column")= chr "geometry"...
> st_crs(noise_sf1) <- 4326
> mapview(noise_sf1, col.regions = "red")
```



Zooming in on the obtained map, one observes that points appear in close proximity to streets. Despite the fact that incidents may occur at random locations, authorities typically record them in relation to the street network. It can introduce some errors or change the actual spatial distribution.

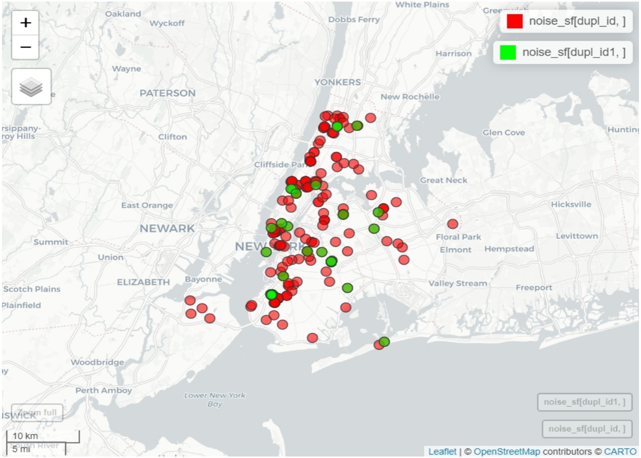


There are several instances where spatial locations are duplicated.

Some of these cases may have been reported by different persons at the same time, or introduced by a mistake. Others may be related to different incidents reported from the same location, or officers recorded them with a delay.

To distinguish between such cases, we first identify them and then plot them using different colours.

```
> dupl_id <-  which(duplicated(MyData1_Helic[,
+ c("Latitude", "Longitude")]))
> dupl_id1 <- which(duplicated(MyData1_Helic[,
+ c("Created.Date", "Latitude", "Longitude")]))
> mapview(noise_sf[dupl_id, ], col.regions = "red") +
+ mapview(noise_sf[dupl_id1, ], col.regions = "green")
```

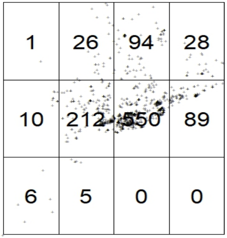

noise_sf[dupl_id,]
noise_sf[dupl_id1,]
Oakland
Wyckoff
Wayne
PATERSON
Clifton
Cliffside Park
East Orange
NEWARK
Summit
Union
ELIZABETH
Bayonne
Scotch Plains
Plainfield
Woodbridge
Perth Amboy
Lower New York Bay
White Plains
Dobbs Ferry
Harrison
YONKERS
New Rochelle
Glen Cove
Great Neck
Hicksville
Floral Park
Elmont
Hempstead
Levittown
Valley Stream
Freeport
Zoom full
10 km
5 mi
Leaflet | © OpenStreetMap contributors © CARTO

Now, let us create a ppp object with spatial locations only:

```
> range_lat <- range(MyData1_Helic$Latitude)
> range_lat
[1] 40.50127 40.90236
> range_lon <- range(MyData1_Helic$Longitude)
> range_lon
[1] -74.24139 -73.73832
> myppp1 <-  ppp(MyData1_Helic$Latitude,
+     MyData1_Helic$Longitude, range_lat, range_lon)
> str(myppp1)
List of 5
$ window    :List of 4
..$ type  : chr "rectangle"
..$ xrange: num [1:2] 40.5 40.9
...
$ n         : int 1021
$ x         : num [1:1021] 40.9 40.7 40.6 40.8 40.8 ...
$ y         : num [1:1021] -73.9 -73.8 -74 -74 -74 ...
```

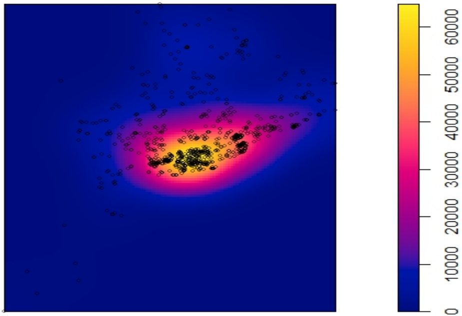
From the previous plots it is obvious that the distribution of incident locations is not uniform in the considered region, which can be confirmed by the quadrat counts and the corresponding plot:

```
> quadratcount(myppp1, nx = 4, ny = 3)
x y              [40.5,40.6) [40.6,40.7) [40.7,40.8) [40.8,40.9]
[-73.9,-73.7]              1          26          94          28
[-74.1,-73.9)             10         212         550          89
[-74.2,-74.1)              6           5           0           0
> Q <- quadratcount(myppp1, nx = 4, ny = 3)
> plot(myppp1, cex = 0.5, pch = "+")
> plot(Q, add = TRUE, cex = 2)
```



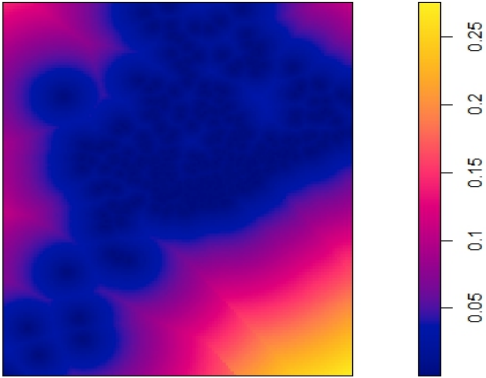
Then, by estimating the intensity of points in the region, one determines that the majority of helicopter noise incidents are expected to occur within a specific subarea:

```
> den1 <- density(myppp1, sigma = 0.03)
> plot(den1)
> plot(myppp1, add = TRUE, cex = 0.5)
```



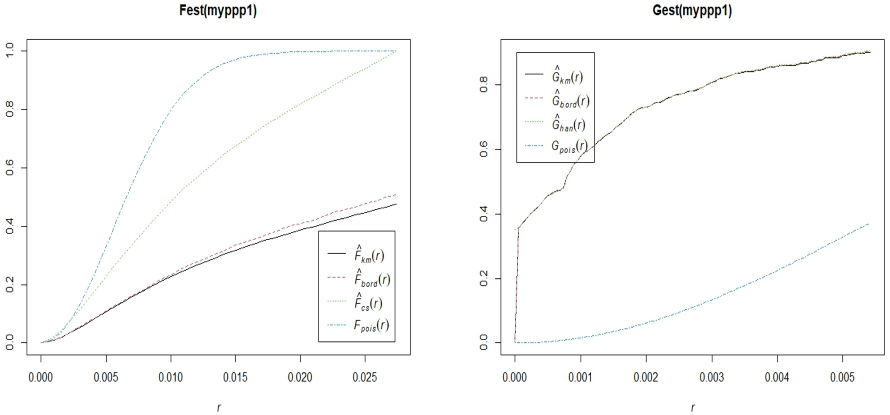
The plot of empty space distances displays locations that are most affected by noise in dark blue and those that are distant from incidents in yellow:

```
> emp1 <- distmap(myppp1)
> plot(emp1)
```



Finally, as expected, the both F and G functions indicate a clustering pattern of the incidents:

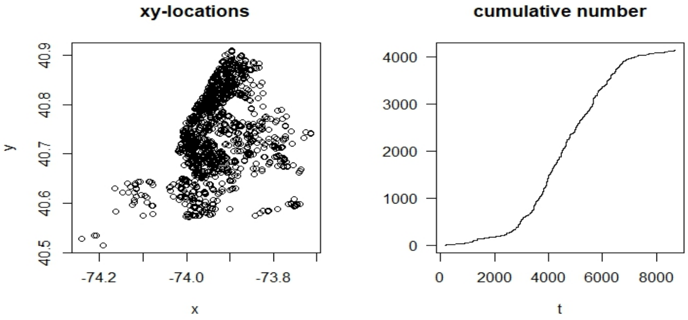
```
> plot(Fest(myppp1))
> plot(Gest(myppp1))
```



Analysis of complains about noise in parks

In this section, we are repeating some of the steps from the previous analysis, but this time focusing on the incidents of noise that were recorded in parks.

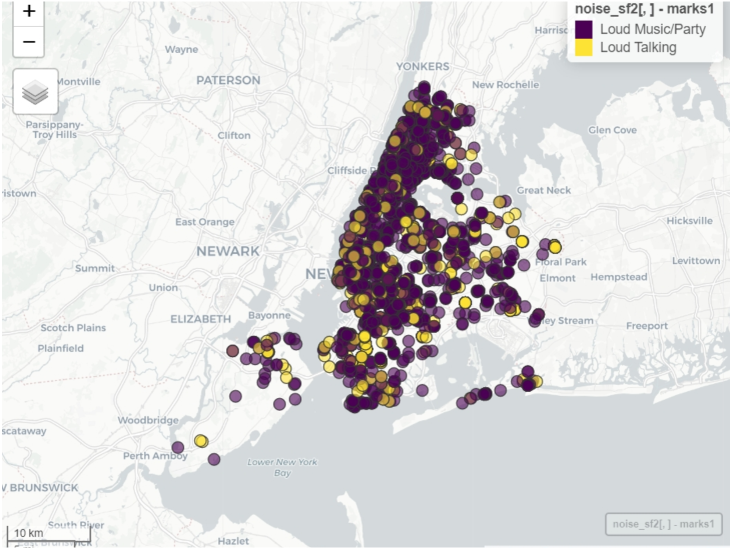
```
> MyData1_Park<- MyData1[MyData1$Complaint.Type=="Noise - Park", ]
> df2<-data.frame(x=MyData1_Park$Longitude,y=MyData1_Park$Latitude,
+ t = MyData1_Park$hours_diff, marks1 = MyData1_Park$Descriptor)
> X2 <- as.3dpoints(df2)
> plot(X2)
```



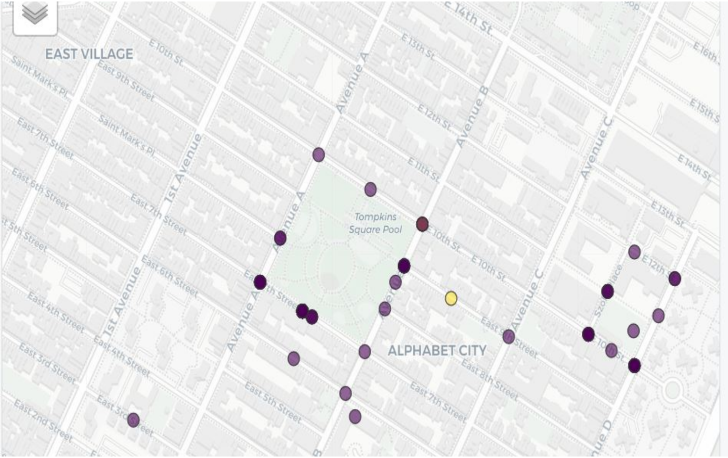
The plots indicate that incidents in parks have a wider spatial distribution compared to helicopter incidents, and their frequency is changing over time. The number of accidents is lower at the beginning and end of the year, while their frequency is much higher during the middle of the year.

After creating and plotting an sf object for the two values of the Descriptor, "Loud Music/Party" and "Loud Talking," it appears that there are more instances of noise due to "Loud Music/Party".

```
> noise_sf2 <- st_as_sf(df2, coords = c("x", "y"))
> st_crs(noise_sf2) <- 4326
> str(noise_sf2)
Classes sf and 'data.frame': 4136 obs. of  3 variables:
$ t  : num  172 174 194 195 221 227 321 414 416 443 ...
$ marks1  : chr  "Loud Music/Party" "Loud Music/Party"...
> mapview(noise_sf2[, ], zcol = "marks1")
```

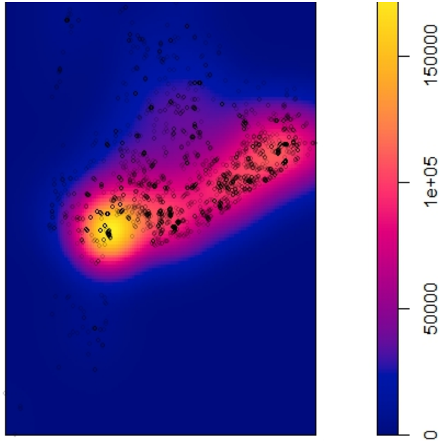

noise_sf2[,] - marks1
Loud Music/Party
Loud Talking
Wayne
PATERSON
YONKERS
New Rochelle
Montville
Parsippany-Troy Hills
Clifton
Glen Cove
Cliffside
Great Neck
East Orange
Hicksville
NEWARK
Levittown
Summit
Floral Park
Elmont
Hempstead
Union
ELIZABETH
Bayonne
Scotch Plains
Freeport
Plainfield
Woodbridge
Perth Amboy
Lower New York Bay
South River
Hazlet
10 km
noise_sf2[,] - marks1

It appears that the majority of incidents involving of Loud Music/Party and Loud Talking noise are not reported at the same location. Upon closer examination of the map, it is evident that many of these occurrences are again concentrated along the street network, but in proximity to parks.



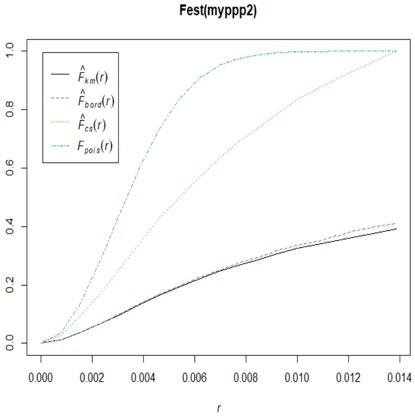
Repeating the analysis of the spatial intensity of the point pattern, it has been discovered that the area with the expected high frequency of events not only encompasses the previously identified region, but has also expanded significantly and is more horizontally stretched. Also, the legend shows that the rate of events is much higher than for the helicopters noise case.

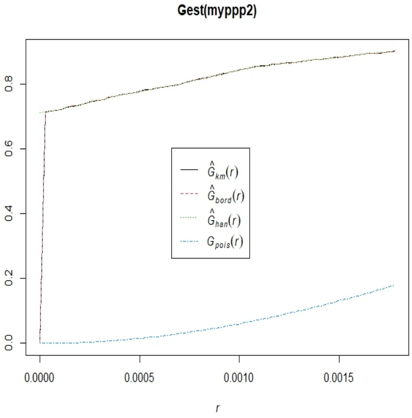
```
> range_lat <- range(MyData1_Park$Latitude)
> range_lat
[1] 40.51553 40.90963
> range_lon <- range(MyData1_Park$Longitude)
> range_lon
[1] -74.24226 -73.71336
> myppp2 <- ppp(MyData1_Park$Latitude,MyData1_Park$Longitude,
+     range_lat, range_lon )
> den2 <- density(myppp2, sigma = 0.03)
> plot(den2)
> plot(myppp2, add = TRUE, cex = 0.5)
```


150000
1e+05
50000
0

Finally, as expected, the both F and G functions indicate a clustering pattern of the incidents:

```
> plot(Fest(myppp2))
> plot(Gest(myppp2))
```





Analysis of types of complains

In this part we briefly analyse spatial distributions of noise incidents due to their types ("Noise - Helicopter" and "Noise - Park").

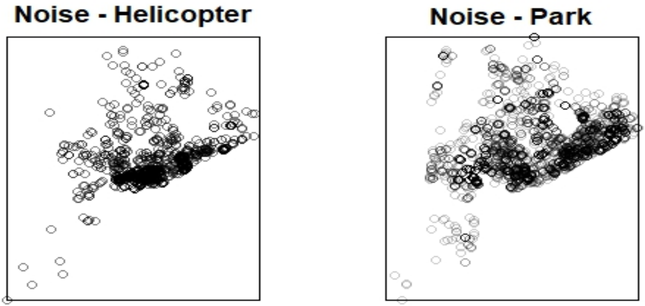
First we create a pp object with spatial locations and marks that correspond to the incident types:

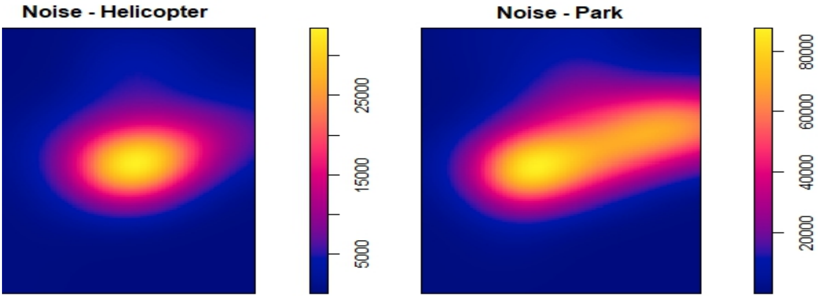
```
> range_lat <- range(MyData1$Latitude)
> range_lat
[1] 40.50127 40.90963
> range_lon <- range(MyData1$Longitude)
> range_lon
[1] -74.24226 -73.71336
> myppp3 <- ppp(MyData1$Latitude,MyData1$Longitude,
+ range_lat,range_lon, marks = factor(MyData1$Complaint.Type))
```

Then we split and plot complaints locations and their estimated intensities by their types:

```
> plot(split(myppp3))
> plot(density(split(myppp3)))
```

These plots are similar to those we produced earlier, but they offer better comparison and more informative insights when viewed side-by-side.





Finally, we estimate the cross-type pair correlation functions between two different event types. The obtained plot indicates the spatial association of these two types of events at all distances, which may not necessarily be attributed to their association, as they more likely occur in the same areas with large number of people. Therefore, this information requires further detailed analysis.

```
> p <- pcfcross(myppp3,"Noise - Helicopter","Noise - Park")
> plot(p)
```