Lab 3 Point Pattern Analysis-Hypothesis Testing

Due September 23, 2014

1. **Statistical modeling of point pattern analysis**

   1.1 Simulating CSR

   “spatstat” provides runifpoint function to generate uniform random points

```
r1<-runifpoint(65)  #Generate 65 uniform random points
plot(r1, pch="+", main="65 points under CSR")
#Quadrat test
quadrat.test(r1,3,3)#Is this CSR
quadrat.test(r1,4,4)#Is this CSR?
#one more random points
r2<-runifpoint(65)
#Quadrat test
quadrat.test(r2,3,3)
quadrat.test(r2,4,4)
```

   1.2 Testing CSR using nearest neighbor distances and Monte Carlo simulation

   Upper and lower bound can be achieved from Monte Carlo simulation. The function below creates upper and lower bounds

```
r1.ghat<-Gest(r1)
r1.ghat$rs #Ghat value at each r
#the following plots are similar except for the x,y limit
par(mfrow=c(1,2))
plot(r1.ghat)
plot(r1.ghat$r, r1.ghat$rs, type="l", xlim=c(o,0.1))
r2.ghat<-Gest(r2)
plot(r2.ghat)
plot(r2.ghat$r, r2.ghat$rs, type="l", xlim=c(o,0.1))
par(mfrow=c(1,1))
#Monte Carlo simulation
#Repeat to generate random points and to take Ghat values 100 times
hold<-matrix(0, nrow=100, ncol=length(r1.ghat$r))
dim(hold)

for (i in 1: 100){
   rp<-runifpoint(65)
```

```
    rp.ghat<-Gest(rp, r1.ghat$r)
    rp.ghat.rs<-rp.ghat$rs
    hold[i,]<-rp.ghat.rs
}

r1.ghat$r[100]              #100th distance point
summary(hold[,100])         #summary of Ghat at the 100th distance point
max(hold[,100])             #upper bound of 100 simulations at 100th point
min(hold[,100])             #lower bound of 100 simulations at 100th point
apply(hold,1,max)[100]
#get the upper bound and lower bound at every point
ubnd<-apply(hold,2,max)
lbnd<-apply(hold,2,min)

#plot the results with JP data
plot(jp.ghat, rs~r, xlim=c(0, max(r1.ghat$r)))
lines(r1.ghat$r,ubnd, lty=2, col=2)
lines(r1.ghat$r,lbnd, lty=2,col=2)
```

Let's make a function for getting upper and lower bounds through simulation

```
ghat.env<-function(n, s, r, win=owin(c(0,1),c(0,1))){
    hold<-matrix(0,s, length(r))
    for (i in 1:s){
            hold[i,]<-Gest(runifpoint(n,win=win), r=r)$rs
    }
    mn<-apply(hold, 2, mean)
    Up<-apply(hold,2,max)
    Down<-apply(hold,2,min)
    return(data.frame(mn,Up, Down))
}

jp.ghat<-Gest(jp)
jp.win<-window(jp)
plot(jp.ghat, rs~r, main="G estimates")
jp.genv<-ghat.env(n=jp$n, s=100, r=jp.ghat$r, win=jp.win)
#upper and lower envelopes
lines(jp.ghat$r, jp.genv$Up,lty=5, col=2)
lines(jp.ghat$r,jp.genv$Down,lty=5, col=2)
```

Similarly, the upper and lower bounds for F function can be derived from Monte Carlo simulation. The following function produced corresponding upper and lower bounds.

```
fhat.env<-function(n, s, r, win=owin(c(0,1),c(0,1))){
    hold<-matrix(0,s, length(r))
    for (i in 1:s){
        hold[i,]<-Fest(runifpoint(n,win=win), r=r)$rs
    }
    mn<-apply(hold, 2, mean)
    Up<-apply(hold,2,max)
    Down<-apply(hold,2,min)
    return(data.frame(mn,Up, Down))
}

jp.fhat<-Fest(jp)
jp.win<-window(jp)
plot(jp.fhat, rs~r, main="F estimates")
jp.fenv<-fhat.env(n=jp$n, s=100, r=jp.fhat$r, win=jp.win)
#upper and lower envelopes
lines(jp.fhat$r, jp.fenv$Up,lty=5, col=2)
lines(jp.fhat$r,jp.fenv$Down,lty=5, col=2)
```

Assignment I

(a) create a plot for F hat with simulating bounds for Japanese pine sapling and provide it in your report

(b) Based on the Ghat and Fhat with the simulation results, interpret the point pattern of Japanese pine sapling

1.3 Testing CSR using K function and Monte Carlo simulation

Similarly, the bounds for K function can be created from simulation

Assignment II

(a) Create a function to get upper and lower bounds for K function as above and include the R code in your report [hint: Kest(runifpoint(n, win=win), r=r)$border

(b) Create a plot for K hat with simulating bounds for Japanese pine sapling and provide it in your report

(c) Based on the simulation results, interpret the point pattern of Japanese pine sapling

2. Conduct point pattern analysis with another dataset

Based on the previous practice, analyze spatial point pattern of following datasets

```
data(redwood) #load the dataset
?redwood #information for the dataset
summary(redwood)

regular<-rsyst(nx=10) #generate regular pattern
summary(regular)
```

Assignment III

(a) For each of the above dataset, conduct point pattern analysis including quadrat test, kernel estimation, nearest neighbor distance, point-to-event distance, K function, and test for CSR with simulations (for nearest point-to-event distance, event-to-event distance, K function)
(b) Report the results of the analysis including all test results and plots
(c) Based on the results, draw conclusions about point patterns of the two dataset (aspects of first and second order) and explain reasons for the conclusion
(d) Please include all the R codes at the end of your report.