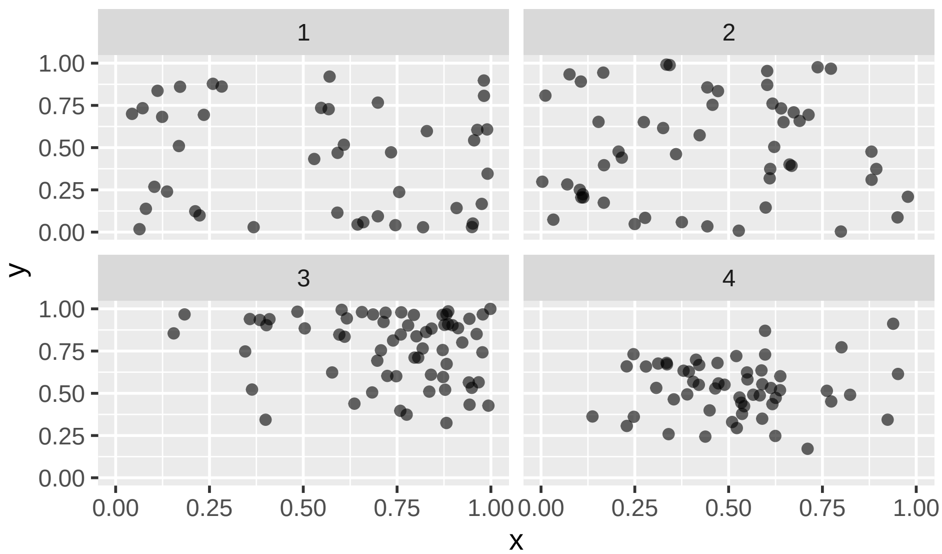
PP Hypothesis Tests

Hypothesis Tests for CSR

- With a spatial point process, *a single realization from the underlying point process is observed.*

- Hypothesis tests and Monte Carlo simulation procedures allow testing for departures from Complete Spatial Randomness (CSR).

- Devise and implement a Monte Carlo-based test for CSR. First detail your test statistic and implementation, then assess the results with the four data sets from Lecture 21.



G and F Functions

- One way to describe a spatial point process, is to consider the probability of being a certain distance from a point or similarly, the number of points expected in a distance from a point.

- *Define $Num(\boldsymbol{s}, d; \boldsymbol{S})$ as the number of points from $\boldsymbol{S}$ in a circle of diameter d from point $\boldsymbol{s}$.*

- Then we can compute the probability that more than one point exists with a distance d of an observed point.

$$E_{\boldsymbol{S}}\left(\sum_{\boldsymbol{s_i} \in \boldsymbol{S}} 1(Num(\boldsymbol{s}_i), d; \boldsymbol{S}) > 0\right) = \lambda |\boldsymbol{D}| P(Num_d(\boldsymbol{s}, d; \boldsymbol{S}) > 0)$$

- The term $P(Num_d(\boldsymbol{s}, d; \boldsymbol{S}) > 0$ is referred to as $G(d)$. $G(d)$ increases in d.

- *The $G(d)$ function is simiilar to a CDF of the **nearest neighbor** distance*

- A similar statistic is the $F(d)$ function. Whereas $G(d)$ is centered at the observed $\boldsymbol{s}_i$, $F(d)$ is defined at any arbitrary point. *Hence this is a CDF for empty space.*

- **QUESTION:** Assume a CSR Poisson process with constant intensity λ, calculate $F(d)$ or $G(d)$ (they are the same).

- Discuss how to create an empirical estimate of $\hat{G}(d)$, given a realization of a point process.

- With bounded area, edge correction procedures are necessary. *Hence, consider*

$$\hat{G}(d) = \frac{\sum_i 1(d_i \leq d < b_i)}{\sum_i 1(d < b_i)},$$

  where b_i is the distance from $\boldsymbol{s}_i$ to the edge and d_i is the distance to the nearest neigbor of $\boldsymbol{s}_i$.

- The empirical estimates of G or H can be compared with G or F using a QQ-plot.

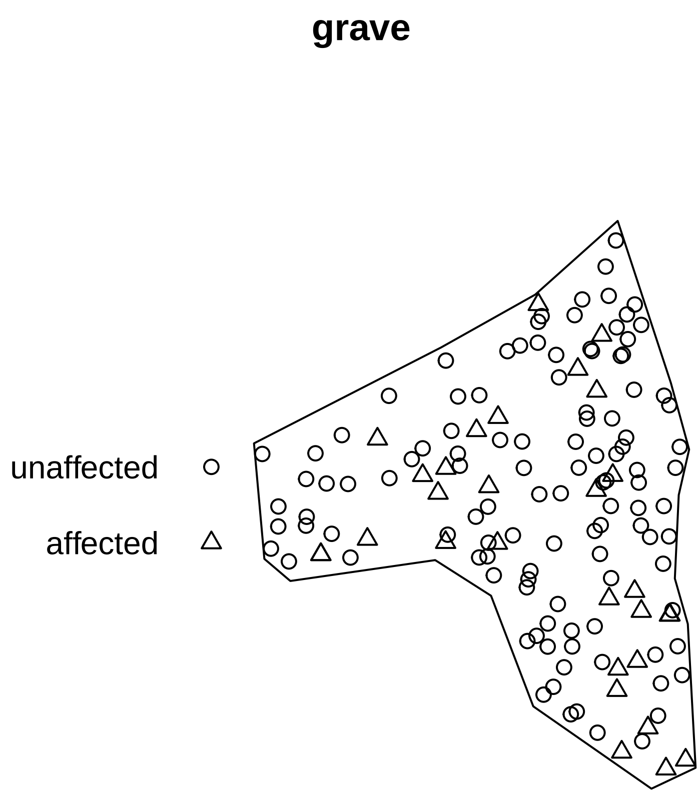
- **Discuss:** What would be the implications of shorter tails or longer tails than expected under CSR?
  shorter tail = clustering/attraction, longer tail = inhibition/repulsion

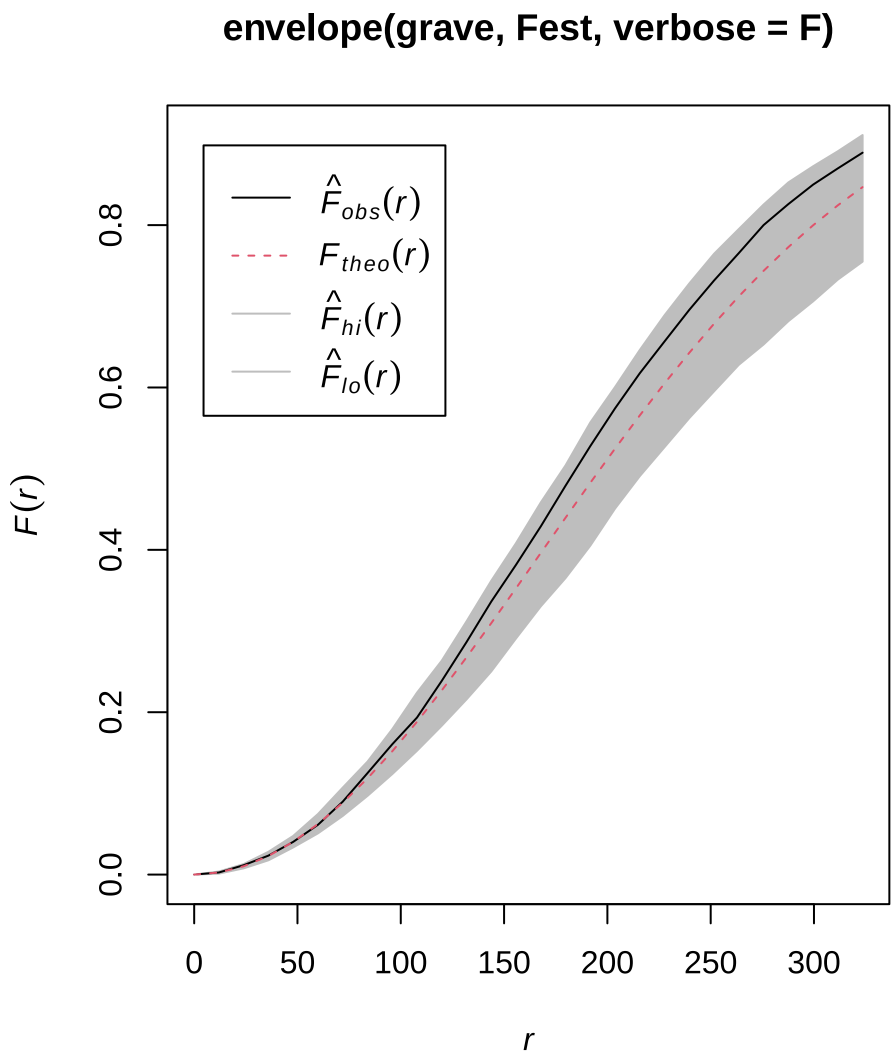
- *Describe a natural process that might cluster and another than might repel*

spatstat

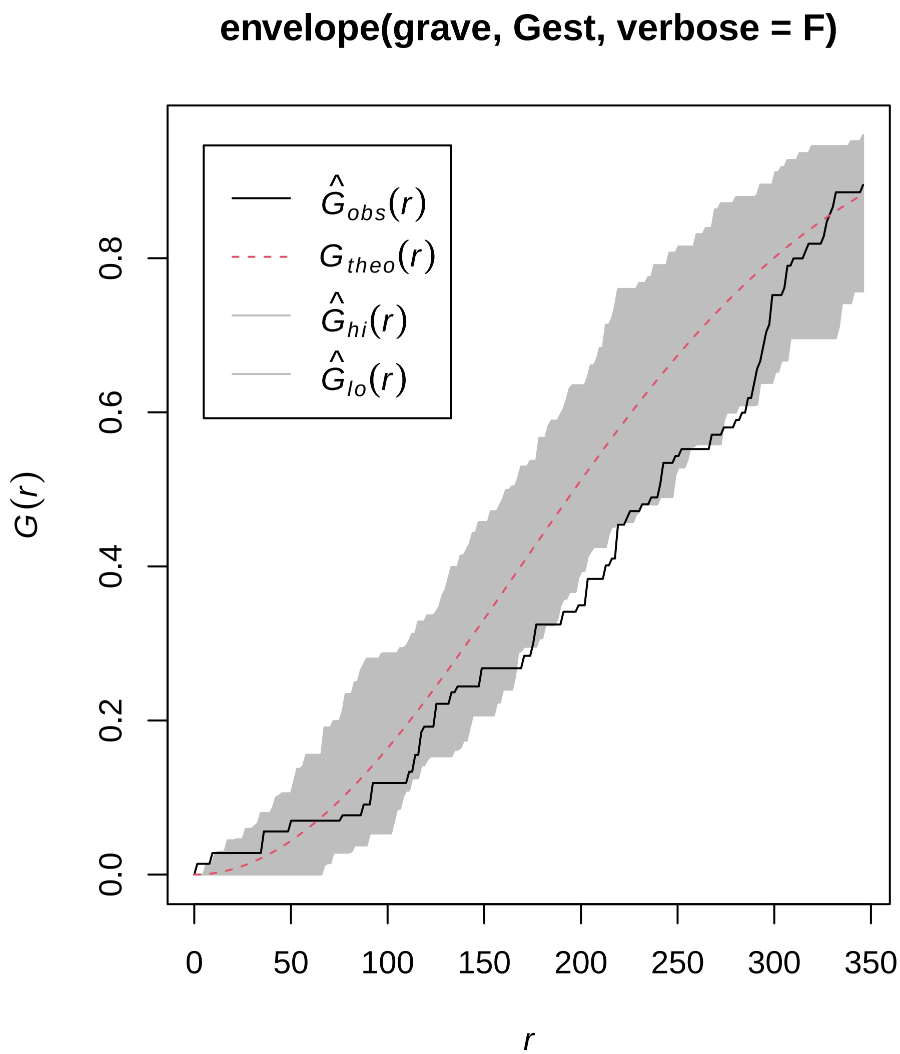
Consider a dataset with medieval grave site information.

```
## Marked planar point pattern:  143 points
## Average intensity 5.70489e-06 points per square unit
##
## Coordinates are integers
## i.e. rounded to the nearest unit
##
## Multitype:
##            frequency proportion   intensity
## unaffected       113  0.7902098 4.50806e-06
## affected          30  0.2097902 1.19683e-06
##
## Window: polygonal boundary
## single connected closed polygon with 16 vertices
## enclosing rectangle: [4376.579, 10511.88] x [2809.612, 10702.971] units
##                      (6135 x 7893 units)
## Window area = 25066200 square units
## Fraction of frame area: 0.518
```


grave
unaffected
affected



envelope(grave, Fest, verbose = F)
$\hat{F}_{obs}(r)$
$F_{theo}(r)$
$\hat{F}_{hi}(r)$
$\hat{F}_{lo}(r)$
0.0
0.2
0.4
0.6
0.8
$F(r)$
0
50
100
150
200
250
300
r

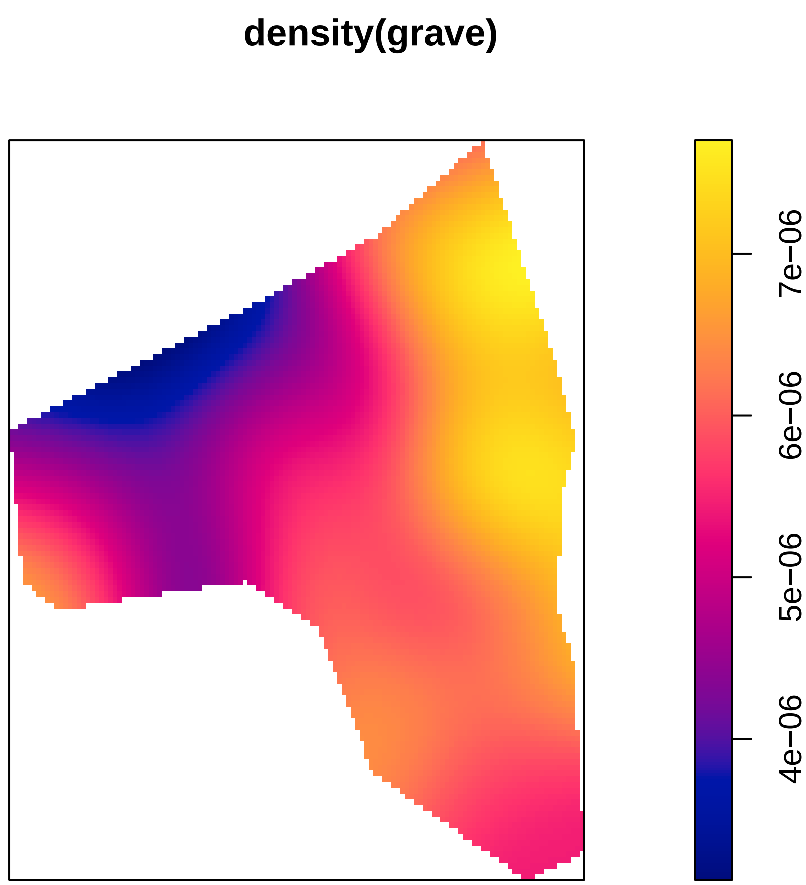

envelope(grave, Gest, verbose = F)
$\hat{G}_{obs}(r)$
$G_{theo}(r)$
$\hat{G}_{hi}(r)$
$\hat{G}_{lo}(r)$
$G(r)$
0.0
0.2
0.4
0.6
0.8
0
50
100
150
200
250
300
350
r

Estimating the intensity Function

- With CSR, the intensity function is trivial

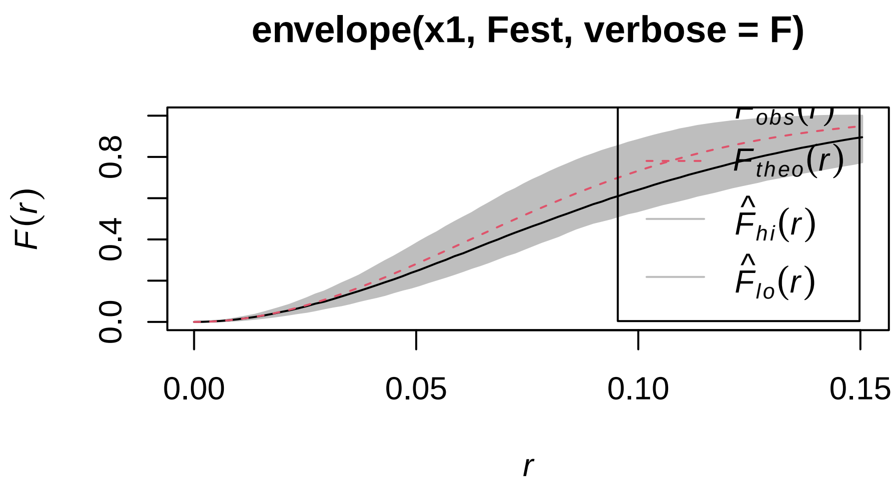
- **Discuss:** given a realization of a point process, how could an intensity function be estimated?

Now using the `plot(density(.))` function, plot and interpret the empirical intensity for the grave dataset along with the four synthetic examples.

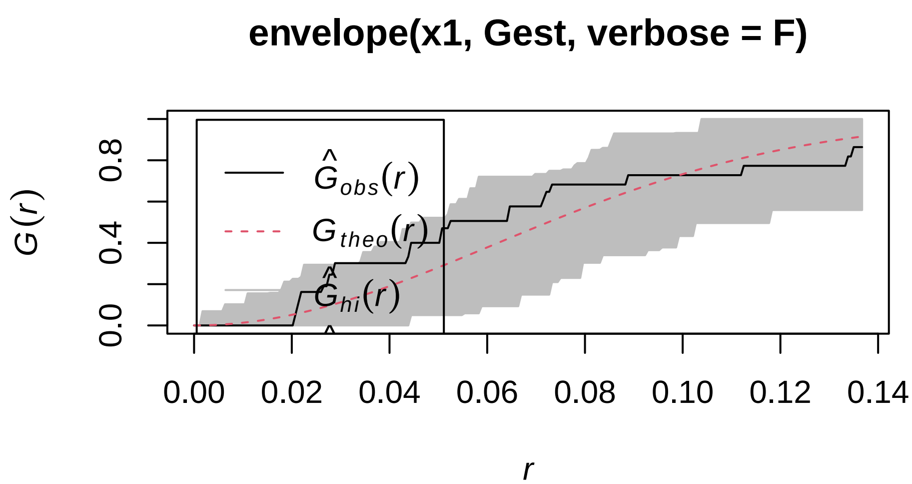


Now let's return to the four datasets we looked at earlier. You can use `envelope` but first need to create `ppp` objects.

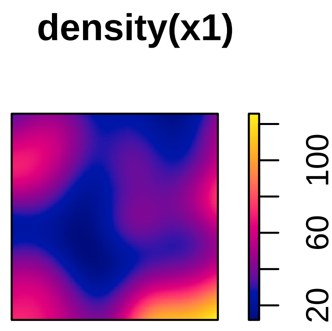
```
x1 <- ppp(x[1:n[1]], y[1:n[1]])
plot(envelope(x1, Fest, verbose = F))
```



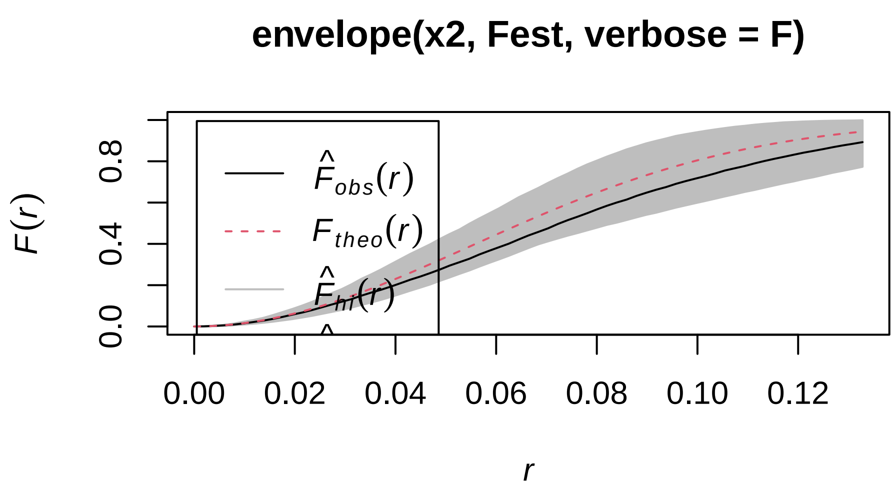

```
plot(envelope(x1, Gest, verbose = F))
```



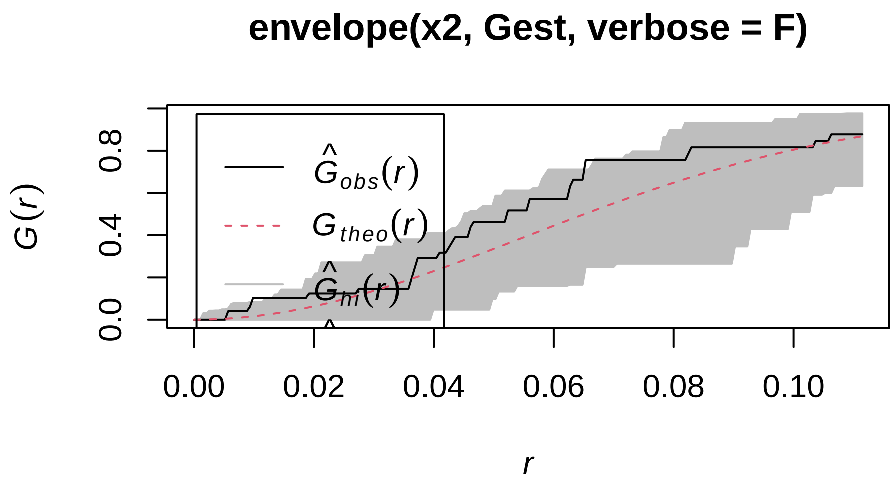

```
plot(density(x1))
```



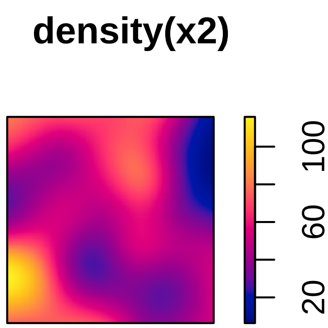

```r
x2 <- ppp(x[(n[1] + 1):(n[1] + n[2])], y[(n[1] + 1):(n[1] + n[2])])
plot(envelope(x2, Fest, verbose = F))
```



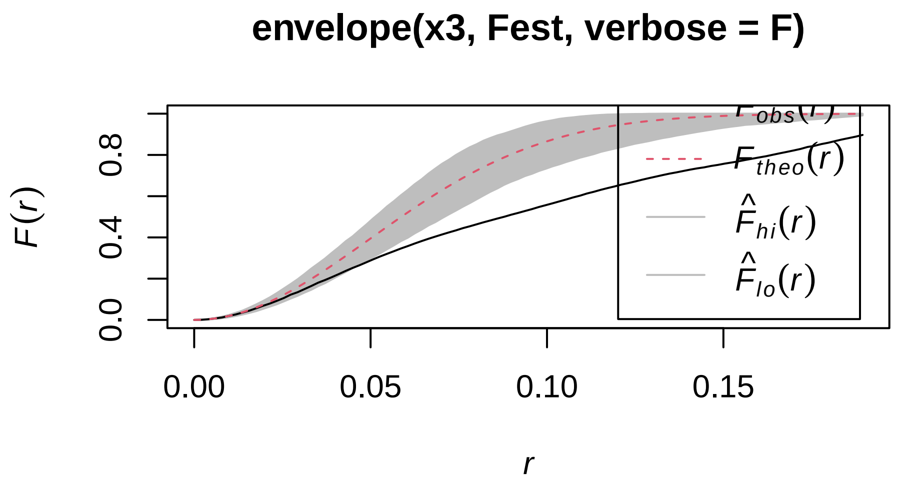

```r
plot(envelope(x2, Gest, verbose = F))
```



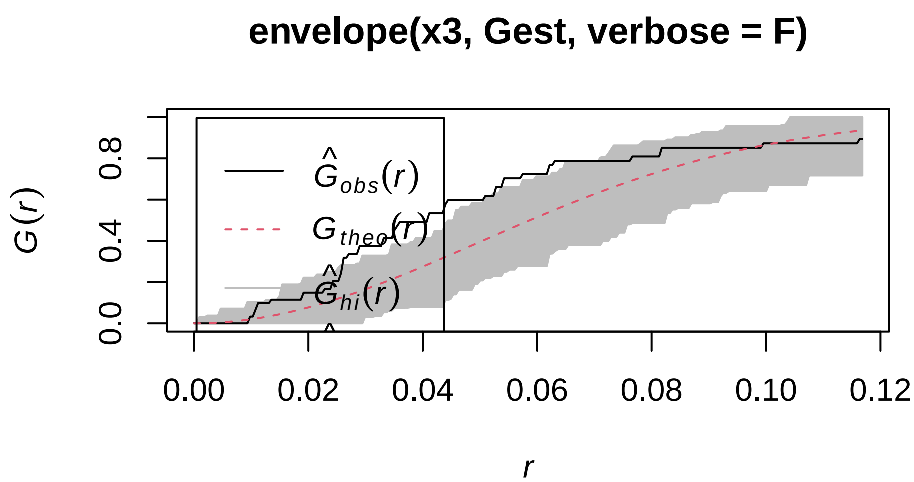

```r
plot(density(x2))
```



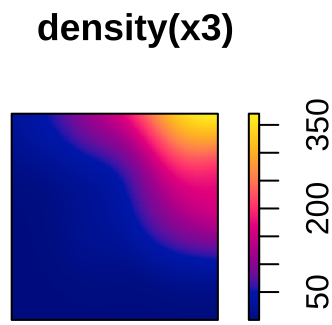

```r
x3 <- ppp(x[(n[1]+ n[2] + 1):(n[1] + n[2] + n[3])],
          y[(n[1]+ n[2] + 1):(n[1] + n[2] + n[3])])
plot(envelope(x3, Fest, verbose = F))
```



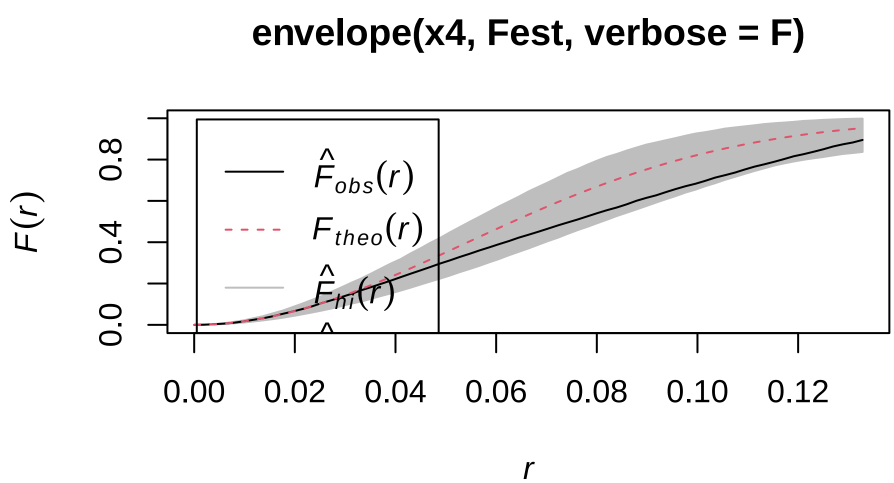

```r
plot(envelope(x3, Gest, verbose = F))
```



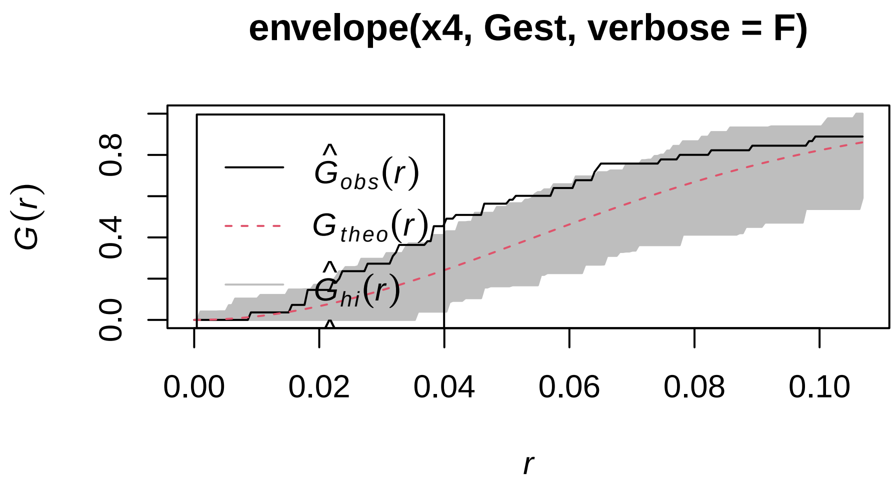

```r
plot(density(x3))
```




```
x4 <- ppp(x[(n[1]+ n[2] + n[3] +1):(n[1] + n[2] + n[3] + n[4])],
          y[(n[1]+ n[2] + n[3] + 1):(n[1] + n[2] + n[3] + n[4])])
plot(envelope(x4, Fest, verbose = F))
```




```
plot(envelope(x4, Gest, verbose = F))
```




```r
plot(density(x4))
```