# K function

### K function

We previously looked at the $F(d)$ and $G(d)$ functions, which corresponded to *CDFs, as a function of distance, for open space and distance between points.*

Another interesting feature of a point process is the number of points in a specified area. Consider $E(Num(\boldsymbol{s}, d, \boldsymbol{S}))$, the expected number of points in $\delta_d \boldsymbol{s}$, a circle of radius $d$ centered at $\boldsymbol{s}$.

*Ripley's K or just the K function, considers the expected number of points within a distance d of an arbitrary point. Formally this is defined for CSR as*

$$K(d) = \frac{E(\text{number of points within d})}{\lambda}$$

*. In other words, this is scaled by $\lambda$*

With CSR, $K(d) = \frac{\lambda \pi d^2}{\lambda} = \lambda d^2$.

To estimate $K(d)$, *we use*

$$\hat{K}(d) = (\hat{\lambda})^{-1} \sum_i \sum_j 1(||\boldsymbol{s}_i - \boldsymbol{s}_j|| \leq d)/n$$

*where $\hat{\lambda} = n/|\mathcal{D}|$, note the typo in the book.*

The empirical $K$ statistic is compared with $\pi d^2$. *For $K > \pi d^2$, the series exhibits clustering, for $K < \pi d^2$ the process exhibits inhibition.*

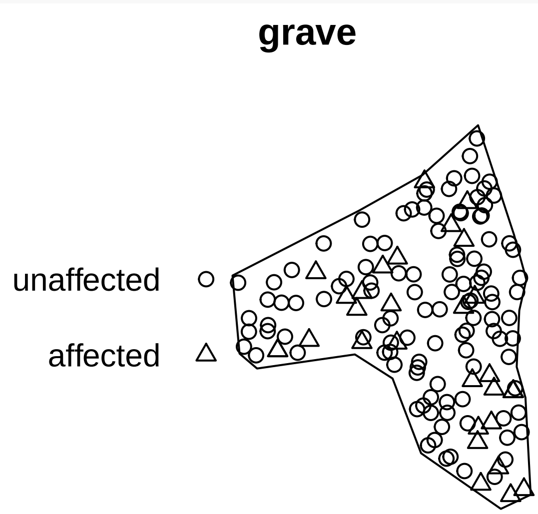
### More spatstat

Consider a dataset with medieval grave site information.

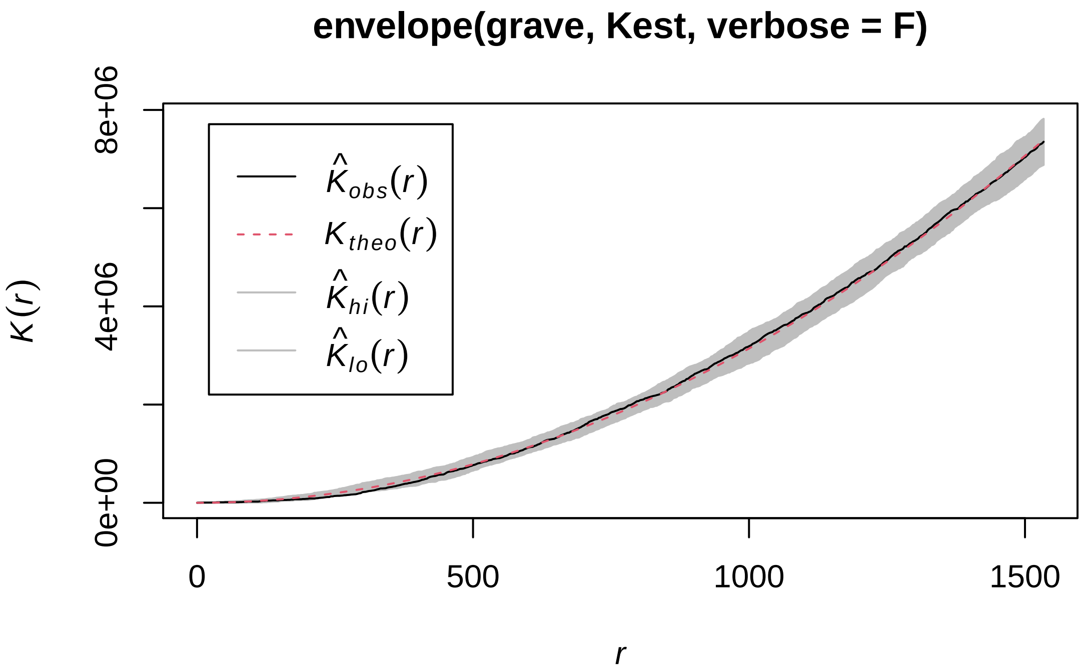
```r
data(grave)
summary(grave)
```

```
## 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
```

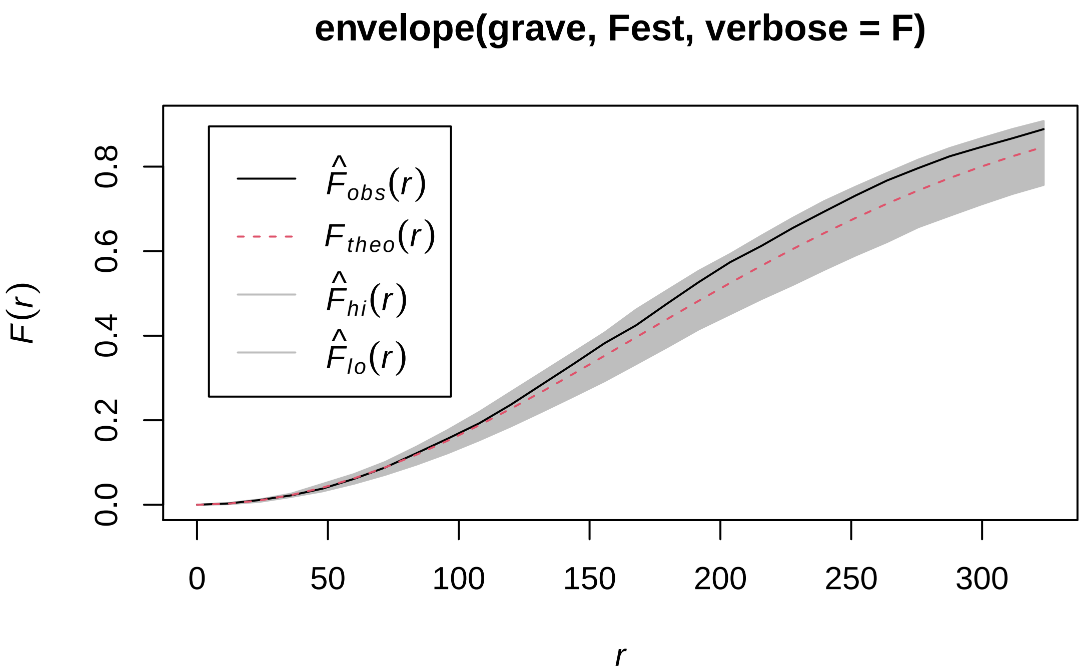
```r
plot(grave)
```



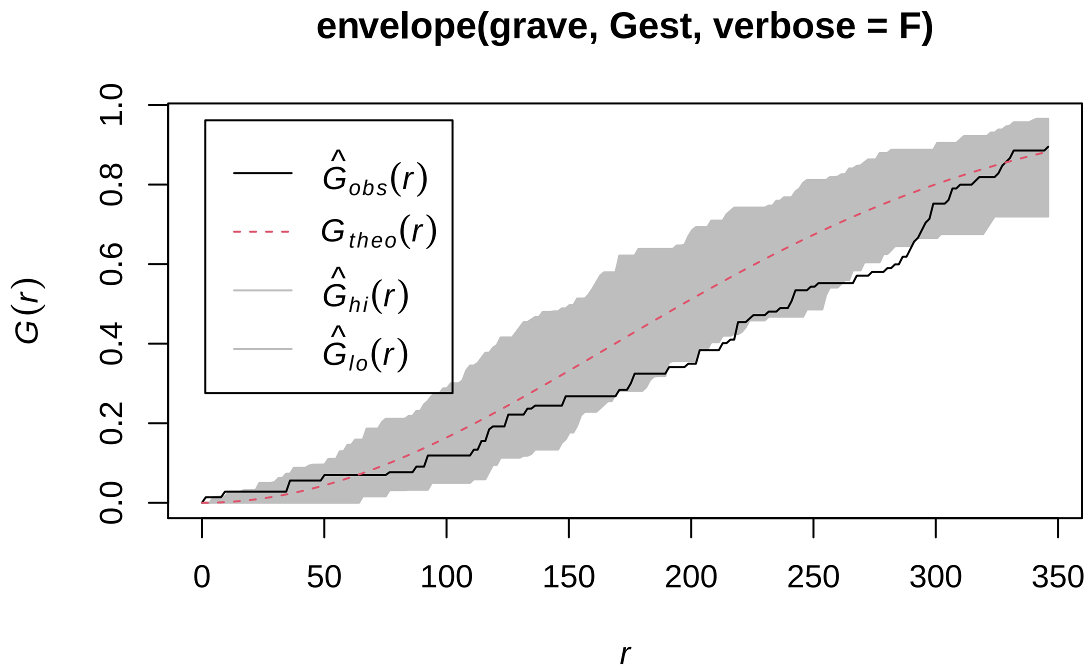

```r
plot(envelope(grave, Kest, verbose = F))
```




```r
plot(envelope(grave, Fest, verbose = F))
```




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




**Estimating the intensity Function**

- With CSR, the intensity function is trivial - *Just uniform with intensity* $\lambda$.

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

*One option would be to discretized histogram. Consider a fine grid, then let* $\lambda(\delta \boldsymbol{s}) = \int_{\delta \boldsymbol{s}} \lambda(\boldsymbol{s}) d\boldsymbol{s} \approx \lambda(\boldsymbol{s})|\delta \boldsymbol{s}|$, *where* $\lambda(\boldsymbol{s})$ *is constant over the grid square. Thus* $\lambda(grid) = \frac{Num(grid)}{Area(grid)}$

*An alternative is to use kernel density estimates. These are basically a smoothed version of a histogram using some sort of symmetric PDF.*

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