Stat 534 Homework 3

Kenny Flagg

February 3, 2017

1. *On the last homework there was some confusion about two problems. I took off some points for one of those but am now giving you a chance to get them back.*

   (a) *It was pointed out in class that, conditional on n events, event locations are uniformly distributed for a homogeneous Poisson process. Show this result for a 1-d process. Hint: Consider a one-dimensional process on a transect of length L, $(0, L]$. Given that one event has occurred on the interval $(0, L]$ what is the probability that it occurred in the subinterval $(0, s]$ for $s < L$? Show that this is the cdf of a* $\mathrm{Unif}(0, L)$ *distribution.*

   First, note that for a homogeneous Poisson process in one dimension with intensity λ, the number of events in an interval follows a Poisson distribution with mean λ times the length of the interval, and the numbers of events in disjoint intervals are independent. If we have a homogeneous Poisson process on $(0, L]$ with intensity λ and one event occurs at location S, then, for $0 < s < L$,

$$\begin{aligned}
P(S \leq s) &= P(1 \text{ event in } (0, s] | 1 \text{ event in } (0, L]) \\
&= \frac{P(1 \text{ event in } (0, s] \text{ and } 1 \text{ event in } (0, L])}{P(1 \text{ event in } (0, L])} \\
&= \frac{P(1 \text{ event in } (0, s])P(0 \text{ events in } (s, L])}{P(1 \text{ event in } (0, L])} \\
&= \frac{e^{-\lambda s}\left(\lambda s\right)^1/1! \times e^{-\lambda(L-s)}\left(\lambda(L-s)\right)^0/0!}{e^{-\lambda L}\left(\lambda L\right)^1/1!} \\
&= e^{\lambda(L-s)}\frac{s}{L}e^{-\lambda(L-s)} \\
&= \frac{s}{L}
\end{aligned}$$

   is the cdf of S, so $S \sim \mathrm{Unif}(0, L)$.

(b) *Suppose we have a realization of a spatial point process consisting of N event locations $\{s_1, s_2, \ldots, s_N\}$. Let H_i denote the distance between the ith event and the nearest neighboring event. The cumulative distribution function of H (the nearest event-event distance) is the G function. (This problem will be continued on the next homework assignment). Derive the G function if the point process is CSR; i.e. what is $G(h) = P(H \leq h)$.*

The G function is

$$\begin{aligned} G(h) &= P(H \leq h) \\ &= P(\text{at least 1 event in a circle of radius } h) \\ &= 1 - P(\text{0 events in a circle of radius } h) \\ &= 1 - \frac{e^{-\lambda\pi h^2}\left(\lambda\pi h^2\right)^0}{0!} \\ &= 1 - e^{-\lambda\pi h^2}, \quad h > 0. \end{aligned}$$

The pdf of H is

$$g(h) = \frac{dG(h)}{dh} = 2\lambda\pi h e^{-\lambda\pi h^2}, \quad h > 0.$$

2. *We looked at one simple method of using nearest neighbor distances to assess a null hypothesis of CSR. The method was based on using Monte Carlo tests to evaluate the deviation of the mean distance from that expected under CSR. We will look at another possible approach in this problem, one that theoretically would allow us to use a test based on normal theory. A question on Homework 2 asked you to find the probability density function of H, the distance between an event and the nearest neighboring event. If you worked this problem correctly you got*

$$g(h) = 2\lambda\pi h \exp\left(-\lambda\pi h^2\right)$$

where $\lambda > 0$. This is a Weibull distribution parametrized as

$$g(h) = \frac{\beta}{\theta^\beta} h^{\beta-1} \exp\left(-\frac{h}{\theta}\right)^\beta$$

and with parameters $\beta = 2$ and $\theta = (\lambda\pi)^{-1/2}$. We will be working with a homogeneous Poisson process with intensity $\lambda = 30$.

(a) *What are the mean and variance of $\overline{H} = (1/30)\sum H_i$ when $\lambda = 30$, i.e. both the sample size and the intensity equal 30?*

Under CSR,

$$E(H_i) = \theta\Gamma\left(1 + \frac{1}{\beta}\right) = \frac{1}{\sqrt{30\pi}}\Gamma\left(\frac{3}{2}\right) = 0.0912871$$

and

$$\begin{aligned} Var(H_i) &= \theta^2 \left(\Gamma\left(1 = \frac{2}{\beta}\right) - \Gamma\left(1 + \frac{1}{\beta}\right)^2 \right) \\ &= \left(\frac{1}{\sqrt{30\pi}}\right)^2 \left(\Gamma(2) - \Gamma\left(\frac{3}{2}\right)^2 \right) \\ &= 0.002277. \end{aligned}$$

If the events are independent (as they are under CSR) then the H_i are also independent. So

$$E\left(\overline{H}\right) = \frac{1}{n}\sum_i E(H_i) = \frac{n \times 0.0912871}{n} = 0.0912871$$

and

$$Var\left(\overline{H}\right) = \frac{1}{n^2}\sum_i Var(H_i) = \frac{n \times 0.002277}{n^2} = \frac{0.002277}{30} = 0.0000759.$$

(b) *What is the approximate sampling distribution of*

$$\frac{\overline{H} - E\left[\overline{H}\right]}{\sqrt{Var\left[\overline{H}\right]}}$$

under CSR and how do you know this?

For a large sample this is approximately standard normal because $\overline{H}$ is a sample mean so the Central Limit Theorem applies.

(c) *Simulate 1000 realizations of complete spatial randomness in the unit square with 30 events in each realization. For each realization*

i. *Calculate the distance between each event and its nearest-neighboring event (H_i for the ith event in the realization)*

ii. *Calculate and store the mean distance.*

iii. *Calculate and store the values of*

$$Z = \frac{\overline{H} - E\left[\overline{H}\right]}{\sqrt{Var\left[\overline{H}\right]}}$$

using the mean and variance from part (a) above.

```
library(spatstat, quietly = TRUE)

# Simulation parameters.
lambda <- 30
n_sim <- 1000

# Simulate.
```

```
sims <- runifpoint(n = lambda, nsim = n_sim)
results <- data.frame(t(sapply(sims, function(x){

  # Nearest neighbor distances.
  H <- nndist(x)

  # Mean nearest neighbor distance.
  Hbar <- mean(H)

  # Z-score.
  Z <- (Hbar - 0.0912871) / sqrt(0.0000759)

  return(c(Hbar = Hbar, Z = Z))
})))
```

(d) *Compute the mean and standard deviation of the 1000 simulated $\overline{H}$ values and compare them to what would be expected under CSR. Are they higher or lower than expected? What could explain this result?*

```
mean(results$Hbar)
```

```
[1] 0.09858615
```

```
sd(results$Hbar)
```

```
[1] 0.01035259
```

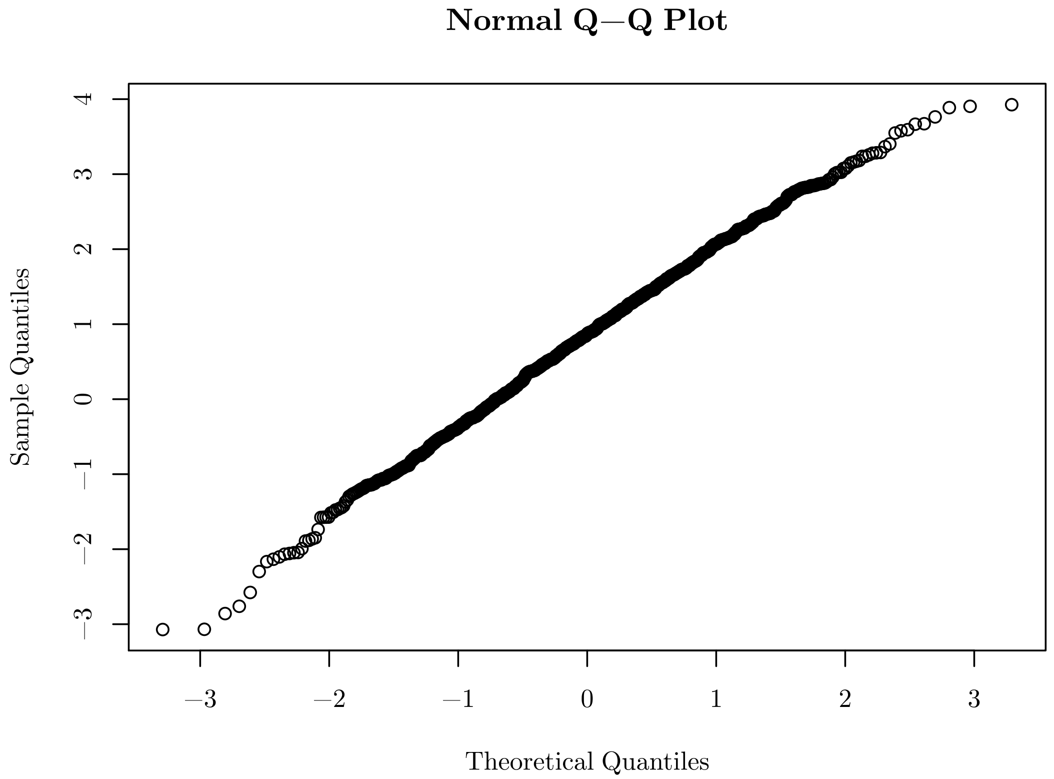
```
var(results$Hbar)
```

```
[1] 0.0001071761
```

The mean and variance of the simulated $\overline{H}$ are both larger than what would be expected under CSR. This could be due to edge effects where points near the boundary have larger nearest neighbor distances than points in the interior of the region, resulting in a skewed distribution of the H_i values and therefore inflating the mean and variance of $\overline{H}$.

(e) *Produce a qqplot of the z scores. Comment.*

```
qqnorm(results$Z)
```



The points form a straight line so there is little doubt that the distribution of simulated $\overline{H}$ values is normal. However, the sample quantiles range from about −3 to about 4 while the theoretical quantiles go from −3 to 3, so it appears the distribution is not exactly *standard* normal.

(f) *Use the following formulas for the expected value and variance of H:*

$$E\left[\overline{H}\right] = 0.5\sqrt{A/n} + 0.051P/n + 0.041P/n^{3/2}$$

$$Var\left[\overline{H}\right] = 0.0703A/n^2 + 0.037P\sqrt{A/n^5}$$

where A is the area and P is the perimeter of the spatial domain (the unit square). Compare the mean and standard deviation from these formulas to those you computed from the simulations above. Does this modification seem to help?

$$E\left(\overline{H}\right) = 0.5\sqrt{\frac{1}{30}} + 0.051\frac{4}{30} + 0.041\frac{4}{30^{3/2}} = 0.0990852$$

$$Var\left(\overline{H}\right) = 0.0703\frac{1}{30^2} + 0.037 \times 4\sqrt{\frac{1}{30^5}} = 0.0001081$$

Yes, the mean and variance computed using these formulas are closer to the simulated mean and variance.

(g) *The above procedure is called the Clark-Evans test. Use it to test the null hypothesis of CSR for the cells and redwood data sets. Interpret the results of each test. Also, compute approximate large sample 95% confidence intervals for the mean distance and interpret.*

Cells

```
data(cells)
cells # Checking that the window is the unit square.

Planar point pattern: 42 points
window: rectangle = [0, 1] x [0, 1] units

Hbar <- mean(nndist(cells))
Hbar

[1] 0.1289729
```

The observed point pattern has 42 trees with a sample mean nearest neighbor distance of $\overline{H} = 0.128973$. Under the null hypothesis of complete spatial randomness, we expect $\overline{H}$ to follow a normal distribution with mean

$$E\left(\overline{H}\right) = 0.5\sqrt{\frac{1}{42}} + 0.051\frac{4}{42} + 0.041\frac{4}{42^{3/2}} = 0.0826113$$

and variance

$$Var\left(\overline{H}\right) = 0.0703\frac{1}{42^2} + 0.037 \times 4\sqrt{\frac{1}{42^5}} = 0.0000528.$$

```
2 * pnorm(0.128973, 0.0826113, sqrt(0.0000528), lower.tail = FALSE)

[1] 1.767195e-10
```

This results in a p-value < 0.0001, very strong evidence that the locations of the cells are not completely spatially random. A 95% confidence interval for the mean nearest neighbor distance is 0.115 to 0.143; this interval suggests that cells are farther from their nearest neighbors than expected under CSR, implying that the cells have a regular pattern.

Redwood

```
data(redwood)
redwood

Planar point pattern: 62 points
window: rectangle = [0, 1] x [-1, 0] units

Hbar <- mean(nndist(redwood))
Hbar

[1] 0.03928432
```

The observed point pattern has 62 trees with a sample mean nearest neighbor distance of $\overline{H} = 0.0392843$. Under the null hypothesis of complete spatial randomness, we expect $\overline{H}$ to follow a normal distribution with mean

$$E\left(\overline{H}\right) = 0.5\sqrt{\frac{1}{62}} + 0.051\frac{4}{62} + 0.041\frac{4}{62^{3/2}} = 0.0671263$$

and variance

$$Var\left(\overline{H}\right) = 0.0703\frac{1}{62^2} + 0.037 \times 4\sqrt{\frac{1}{62^5}} = 0.0000232.$$

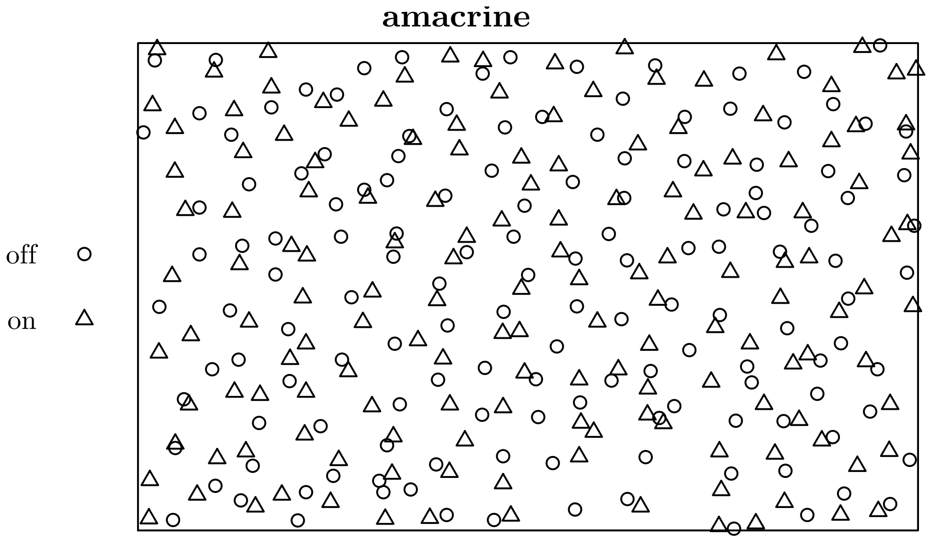
```
2 * pnorm(0.0392843, 0.0671263, sqrt(0.0000232))
```

```
[1] 7.453181e-09
```

This results in a p-value < 0.0001, very strong evidence that the locations of the trees are not completely spatially random. A 95% confidence interval for the mean nearest neighbor distance is 0.0298 to 0.0487; this interval suggests that trees are closer to their nearest neighbors than expected under CSR, implying that the trees are clustered.

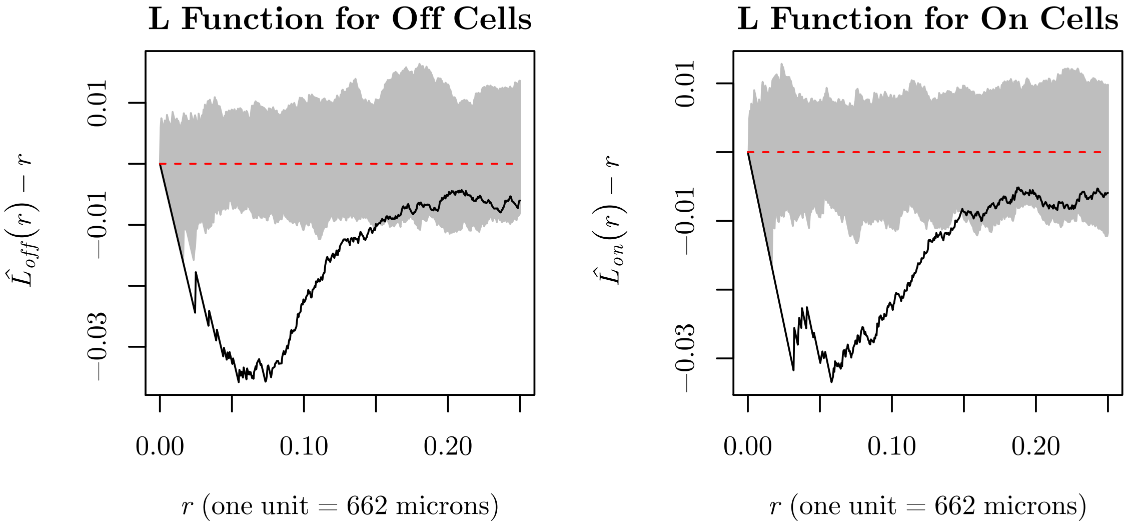
3. *I am sending you a copy of a paper by Peter Diggle on the use of K and cross K functions in the analysis of spatial point patterns. The data he is referring to are in the amacrine data set in the spatstat library in R. Read the paper and reproduce the analysis. The data are in* `spatstat` *(use the command* `data(amacrine)`*. You do not have to carry out the significance tests he refers to but I would like for you to take the same approach I took on the analysis of the* `betacells` *data set we discussed in class. Write up a summary of your analysis. Pay attention to the distinction between the independence and random labeling hypotheses.*

```
data(amacrine)
par(mar = c(0, 0, 0.5, 0))
plot(amacrine)
```



As Diggle did, I will assess the support for two hypotheses, H1: the on cells and off cells result from independent processes, and H2: the on cells and off cells result from a single process and are differentiated by random labeling. I begin by using the L function for each cell type to examine the second-order structure of the observed patterns.

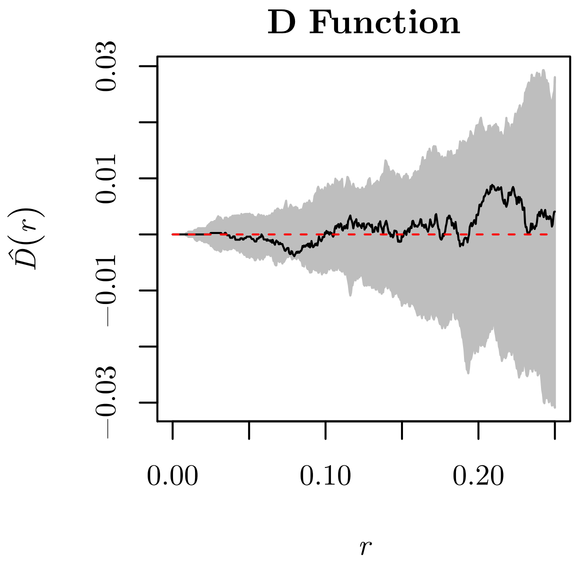
```
amacrine_split <- split(amacrine)
par(mfrow = c(1, 2), mar = c(4, 5, 2, 1))
plot(envelope(amacrine_split$off, fun = Lest, verbose = FALSE), .-r~r,
     legend = FALSE, ylab = expression(hat(L)[off](r)-r),
     main = 'L Function for Off Cells')
plot(envelope(amacrine_split$on, fun = Lest, verbose = FALSE), .-r~r,
     legend = FALSE, ylab = expression(hat(L)[on](r)-r),
     main = 'L Function for On Cells')
```



The L functions are generally similar, both taking values less than r and falling outside the pointwise confidence band for r less than 0.15 or so. This indicates regularity for both cell types. I will use the D function to assess equality of the L functions more formally.

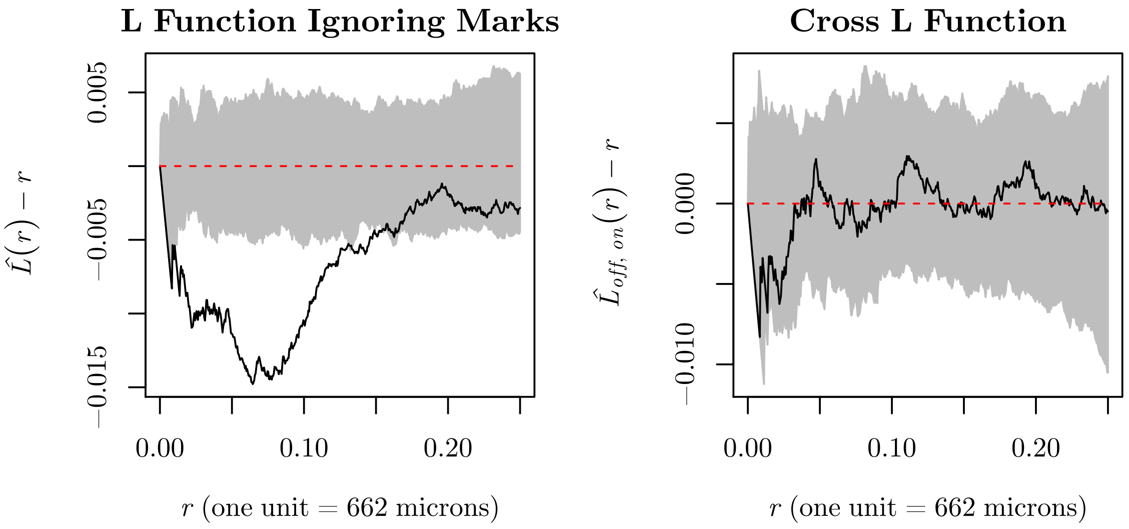
```
library(splancs, quietly = TRUE)
amacrine_box <- bboxx(rbind(Window(amacrine)$xrange, Window(amacrine)$xrange))
r <- Lest(amacrine)$r
k_off <- khat(as.points(amacrine_split$off), amacrine_box, r)
k_on <- khat(as.points(amacrine_split$on), amacrine_box, r)
D_hat <- k_off - k_on
Denv <- Kenv.label(as.points(amacrine_split$off), as.points(amacrine_split$on),
        amacrine_box, nsim = 99, r, quiet = TRUE)

# Use spatstat's plotting system for a consistent look.
Dfv <- fv(data.frame(r = r, Dhat = D_hat, lo = Denv$lower, hi = Denv$upper),
          valu = 'Dhat', yexp = expression(hat(D)(r)))
fvnames(Dfv, '.s') <- c('lo', 'hi')
par(mar = c(4, 5, 2, 1))
plot(Dfv, main = 'D Function', legend = FALSE)
segments(y0 = 0, x0 = min(r), x1 = max(r), col = 'red', lty = 2)
```



The D function stays near 0 for all r, so we have no reason to believe that the K (or L) functions differ between the cell types. Similarity of the L functions is consistent with both independence and random labeling, so I'll investigate further using the L function for all cells of both types and the cross L function.

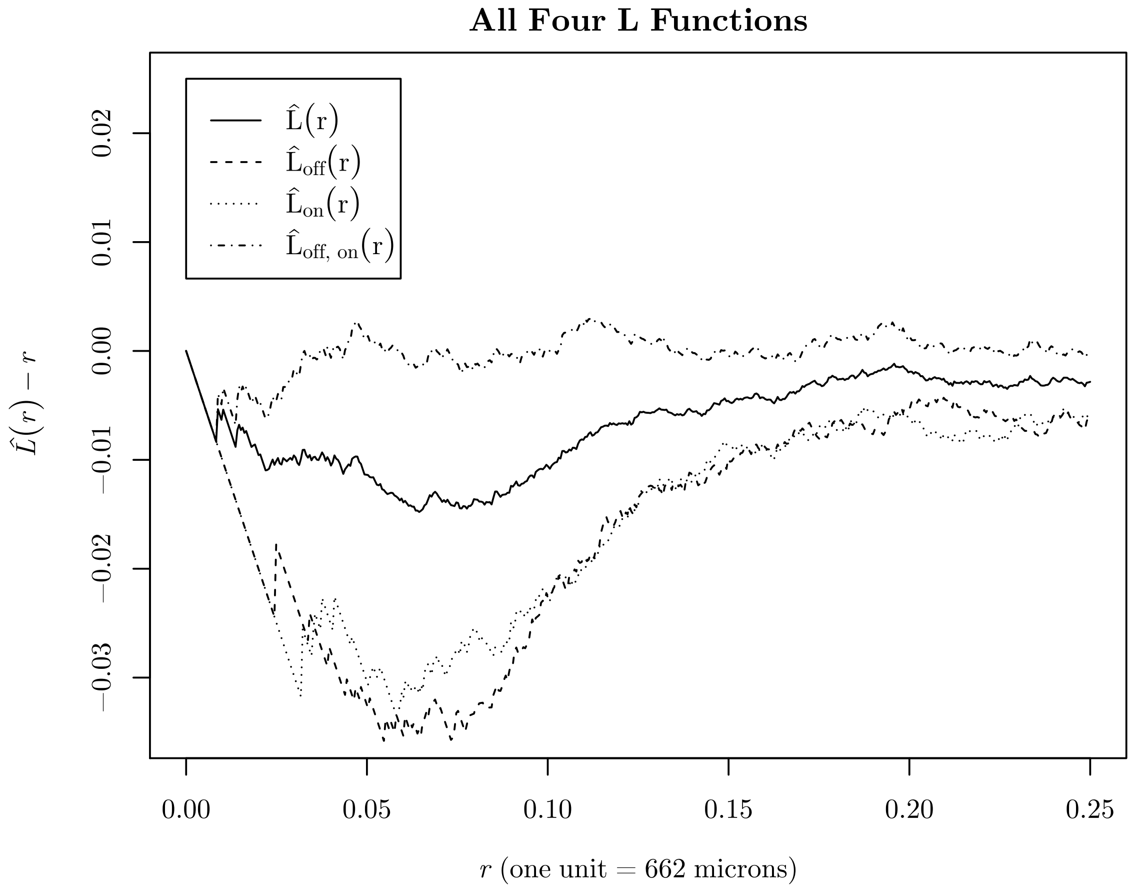
```
par(mfrow = c(1, 2), mar = c(4, 5, 2, 1))
plot(envelope(amacrine, fun = Lest, verbose = FALSE), .-r~r,
     legend = FALSE, ylab = expression(hat(L)(r)-r),
     main = 'L Function Ignoring Marks')
plot(envelope(amacrine, fun = Lcross, verbose = FALSE), .-r~r,
     legend = FALSE, ylab = expression(hat(L)[list(off,on)](r)-r),
     main = 'Cross L Function')
```



The L function ignoring cell type looks about the same as the L functions for the separate cell types, but it does not deviate quite as far from zero. As a single point pattern, the locations of the cells have a regular pattern on a scale up to about 0.10 to 0.15 units. The cross L function stays within the pointwise confidence bounds, providing no evidence of dependence between the two point patterns. However, the cross L function is clearly different from the L function of either cell type, which rules out random labeling.

I like Diggle's plot showing all four K functions, so I recreated it with the L functions. Random labeling requires all four of these functions to be equal, which is not the case here. The cross L function is not very different from r for all r, so there is no reason to believe the locations of cells of different types are not independent. The evidence is consistent with H1, that the on cells and off cells are placed in regular patterns by independent processes.

```
par(mar = c(4, 5, 2, 1))
plot(Lest(amacrine), iso -r ~ r, ylab = expression(hat(L)(r) - r),
     main = 'All Four L Functions',
     ylim = c(-0.035, 0.025), legend = FALSE, lty = 1)
plot(Lest(amacrine_split$off), iso - r ~ r, lty = 2, add = TRUE)
plot(Lest(amacrine_split$on), iso - r ~ r, lty = 3, add = TRUE)
plot(Lcross(amacrine), iso - r ~ r, lty = 4, add = TRUE)
legend(0, 0.025, lty = 1:4, legend = expression(hat(L)(r), hat(L)[off](r),
                                                 hat(L)[on](r), hat(L)[list(off,on)](r)))
```


All Four L Functions
$\hat{L}(r)$
$\hat{L}_{off}(r)$
$\hat{L}_{on}(r)$
$\hat{L}_{off,\,on}(r)$
$\hat{L}(r) - r$
r (one unit = 662 microns)