

Problem 6.5 posed in Bullmer's *Principles of Statistics*, Dover, 1978 has to do with the spacing of random points.

If skylarks' nests are distributed at a random over a large area of ground, show that the square of the distance from a nest to its nearest neighbor follows an exponential distribution. (This fact has been used to test whether there is any territorial spacing of nests.)

Here is the derivation. It is like the derivation of an exponential variable.

Let's find the probability  $P(x)$  that there will be no nests within a distance  $x$  of some fixed point. Suppose that there are  $\lambda$  nests per unit area. The probability that there will be a nest in an infinitesimal annulus between distances  $x$  and  $x + \Delta x$  is  $2\pi\lambda x\Delta x$  which is the first order approximation of the area  $\pi[(x + \Delta x)^2 - x^2] = 2\pi x\Delta x + \pi(\Delta x)^2$  because the area of the circle of radius  $x$  is  $\pi x^2$ . Let  $X$  be the distance to the nearest nest. Assuming that the probability of more than one nest in the  $x$  to  $x + \Delta x$  annulus is negligible, then the probability that no nest occurs is  $1 - 2\pi\lambda x\Delta x$ . We also assume that the chances of a nest occurring in any annulus is independent of the event that a nest occurs in any other annulus. The probability that there are no nests within  $x + \Delta x$  is by independence the probability that there are no nests within  $x$  times the probability that no nest occurs in the annulus between  $x$  and  $x + \Delta x$ .

$$P(x + \Delta x) = P(x)(1 - 2\pi\lambda x \Delta x)$$

$$\frac{P(x + \Delta x) - P(x)}{\Delta x} = -2\pi\lambda x P(x).$$

Letting  $\Delta x \rightarrow 0$ , we arrive at the differential equation

$$\frac{dP(x)}{dx} = -2\pi\lambda x P(x).$$

Using the initial condition  $P(0) = 1$ , we arrive at the solution

$$P(x) = -\exp(\pi\lambda x^2).$$

Now, the cdf  $F(t) = 1 - P(t)$  from which it follows that  $X^2$  is an exponential random variable with rate constant  $\pi\lambda$ .

To explore this fact, we consider  $n = 500$  uniformly distributed points on the unit square, and look at the distribution nearest point squared distances. The distribution is very nearly exponential. We compute the squared distance to the closest neighbor from the  $i$ th point and call it  $e_i$ . The expectation of the exponential variable  $Y$  is  $E(Y) = 1/\lambda$ . We use the estimator

$$\hat{\lambda} = \frac{n}{\sum_{i=1}^n e_i}.$$

Note that  $\hat{\lambda}$  is an estimator for  $\pi\lambda$  above. The histogram for squared distances had bars of width  $w = .0005$ . To compute the expected frequency  $f_i$  and expected proportion  $p_i$  in any bar, we compute the proportion of data to fall in the bar using  $\hat{\lambda}$ . Thus, for the  $i$ th bar

$$p_i = \int_{(i-1)w}^{iw} \hat{\lambda} e^{-\hat{\lambda}t} dt = \text{pexp}(iw, \hat{\lambda}) - \text{pexp}((i-1)w, \hat{\lambda}).$$

Since  $1 - \text{pexp}(9w, \hat{\lambda}) < .001$  we didn't adjust for the tail. The expected frequencies are thus  $f_i = nq_i$ .

---

**R Session:**

---

R version 2.10.1 (2009-12-14)  
Copyright (C) 2009 The R Foundation for Statistical Computing  
ISBN 3-900051-07-0

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

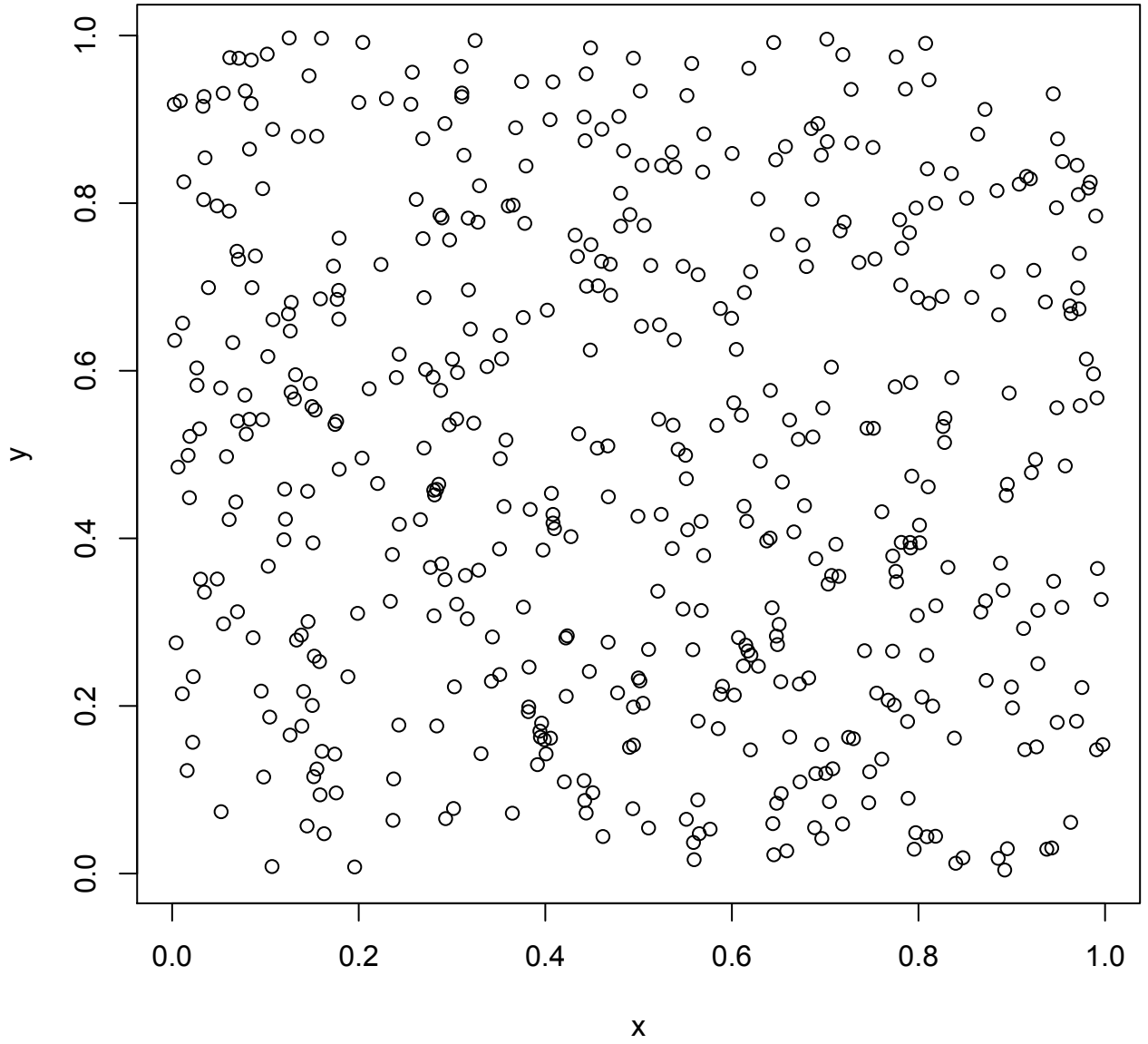
[R.app GUI 1.31 (5538) powerpc-apple-darwin8.11.1]

[Workspace restored from /Users/andrejstreibergs/.RData]

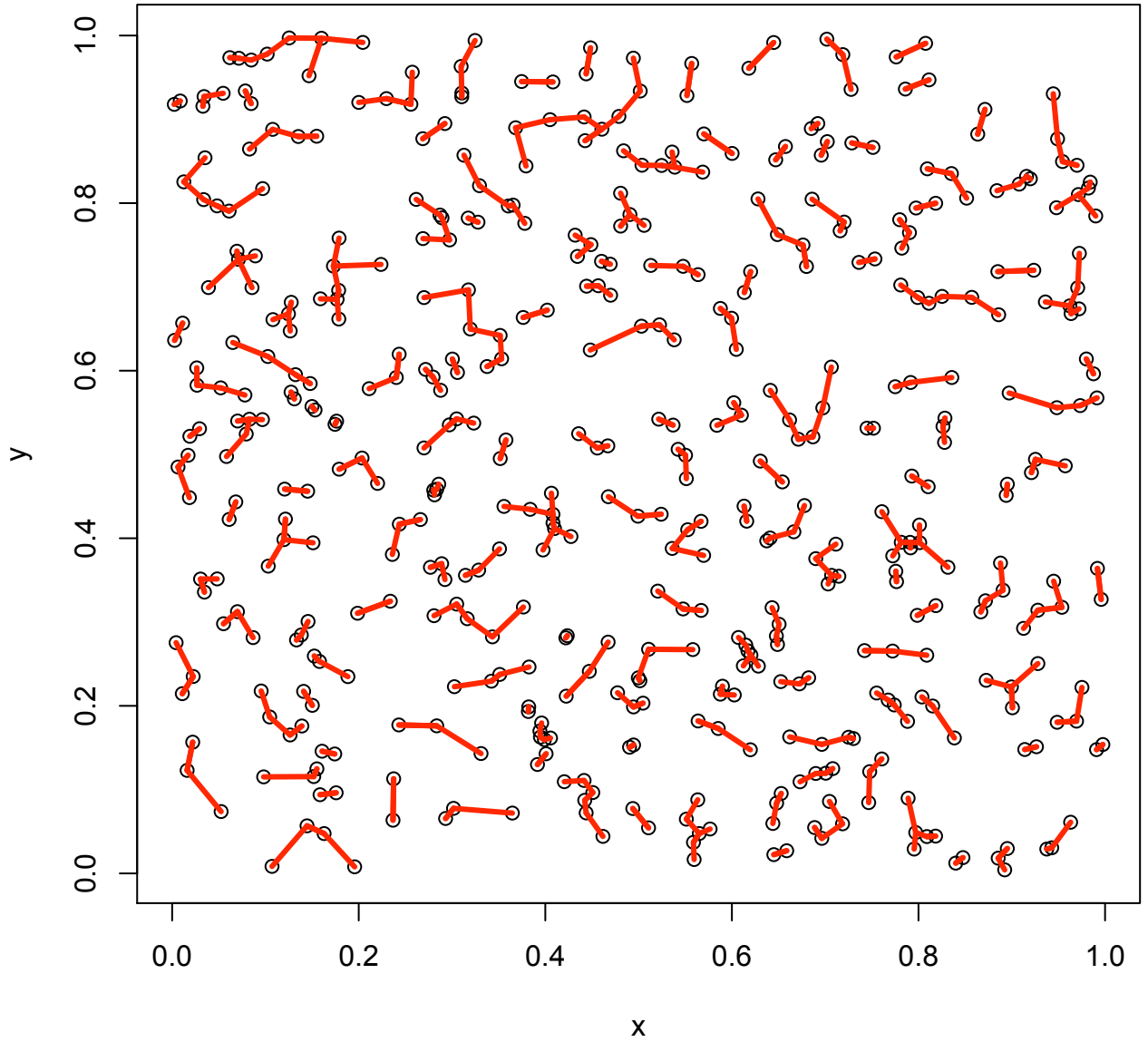
```
> ##### GENERATE AND PLOT 500 RANDOM POINTS #####
>
> n <- 500
> x<- runif(n)
> y <- runif(n)
> plot(x,y)
> plot(x,y,main=paste(n," Uniformly Distributed Points"))
> # M3074Skylark1.pdf

> ##### FIND CLOSEST POINTS AND CONNECT WITH EDGE #####
> # order() identifies location of ith smallest
>
> plot(x,y,main=paste(n," Unif. Points & Closest Neighbors"))
> for( i in 1:n){
+       l <- order( (x-x[i])^2 + (y-y[i])^2)[2]
+       lines(x[c(i,l)],y[c(i,l)],col=2,lwd=3)
+     }
> # M3074Skylark2.pdf
```

### 500 Uniformly Distributed Points



### 500 Unif. Points & Closest Neighbors



```

> ##### EXTRACT SQUARED DISTANCE TO CLOSEST POINT #####
> # distribution of squared distances
> ex <- numeric(n)
> r <- numeric(n)
> for( i in 1:n){
+ r <- (x-x[i])^2 + (y-y[i])^2
+ ex[i] <- min(r[-i])}

> ##### EXTRACT COUNTS, BREAKS #####
> h <- hist(ex)
> h
$breaks
 [1] 0.0000 0.0005 0.0010 0.0015 0.0020 0.0025 0.0030
 [8] 0.0035 0.0040 0.0045

$counts
 [1] 284 104 65 25 12 5 1 3 1

$intensities
 [1] 1136.000 416.000 260.000 100.000 48.000 20.000
 [7] 4.000 12.000 4.000

$density
 [1] 1136.000 416.000 260.000 100.000 48.000 20.000
 [7] 4.000 12.000 4.000

$mids
 [1] 0.00025 0.00075 0.00125 0.00175 0.00225 0.00275
 [7] 0.00325 0.00375 0.00425

$xname
 [1] "ex"

$equidist
 [1] TRUE

attr("class")
 [1] "histogram"

> ##### COMPUTE LAMBDA HAT AND DENSITY FUNCTION #####
> nt <- length(h$counts)
> sc <- sum(h$counts); sc
 [1] 500
> fc <- h$counts/sc
> lambda <- sc/sum(ex);lambda
 [1] 1564.226

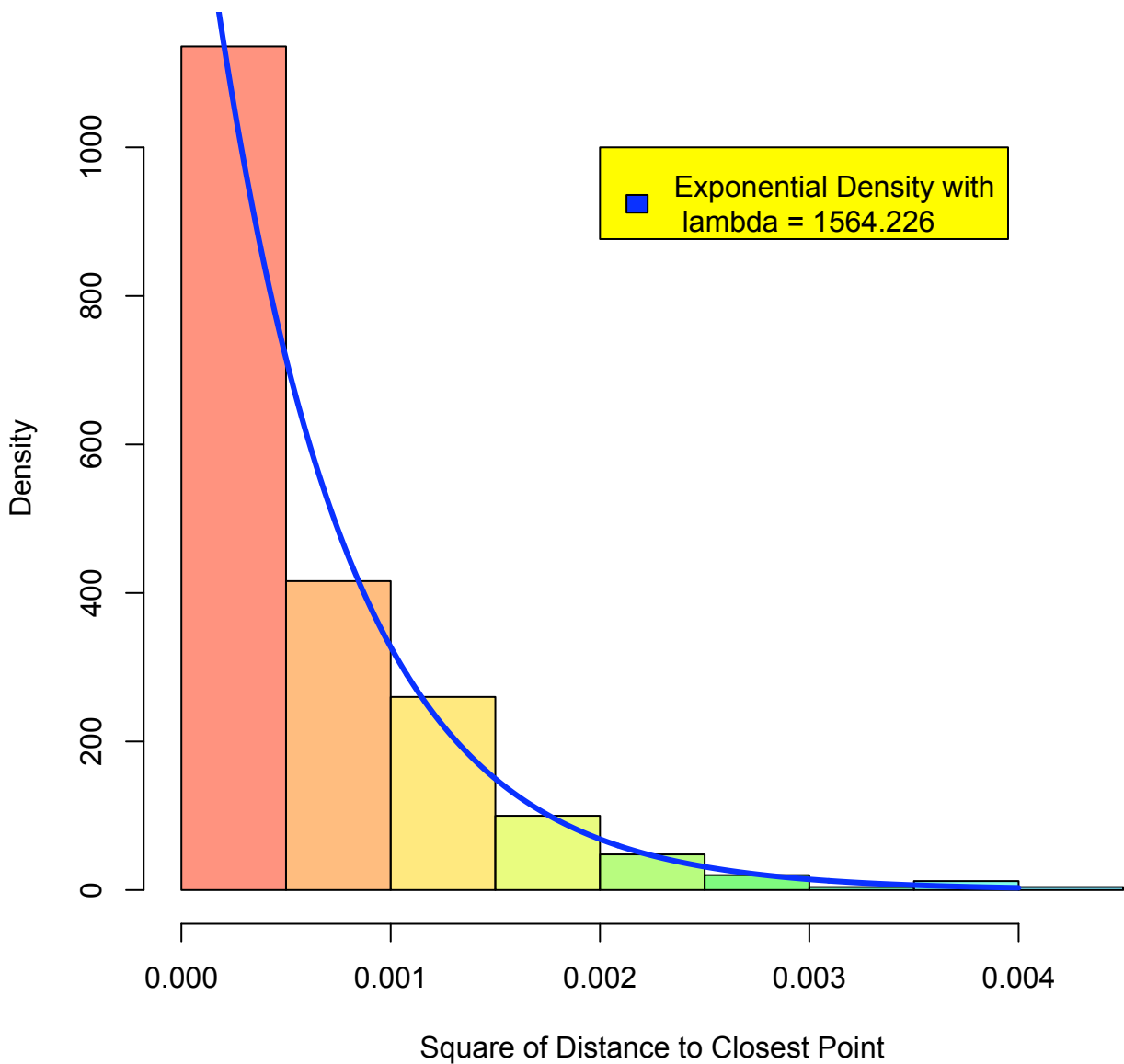
```

```

> ##### PLOT HISTOGRAM AND DENSITY CURVE #####
> hist(ex, main = paste("Hist. of ", n, "Closest Point Distances^2"),
+ xlab = "Square of Distance to Closest Point", freq = F,
+ col = rainbow(15,alpha=.5))
> lines(step, dexp(step, lambda), col = 4, lwd = 3)
> legend(.002, 1000, legend = paste("Exponential Density with\n",
+ "lambda =", round(lambda,3), ""), fill=4, bg="yellow")
> # M3074Skylark4.pdf

```

### Hist. of 500 Closest Point Distances<sup>2</sup>



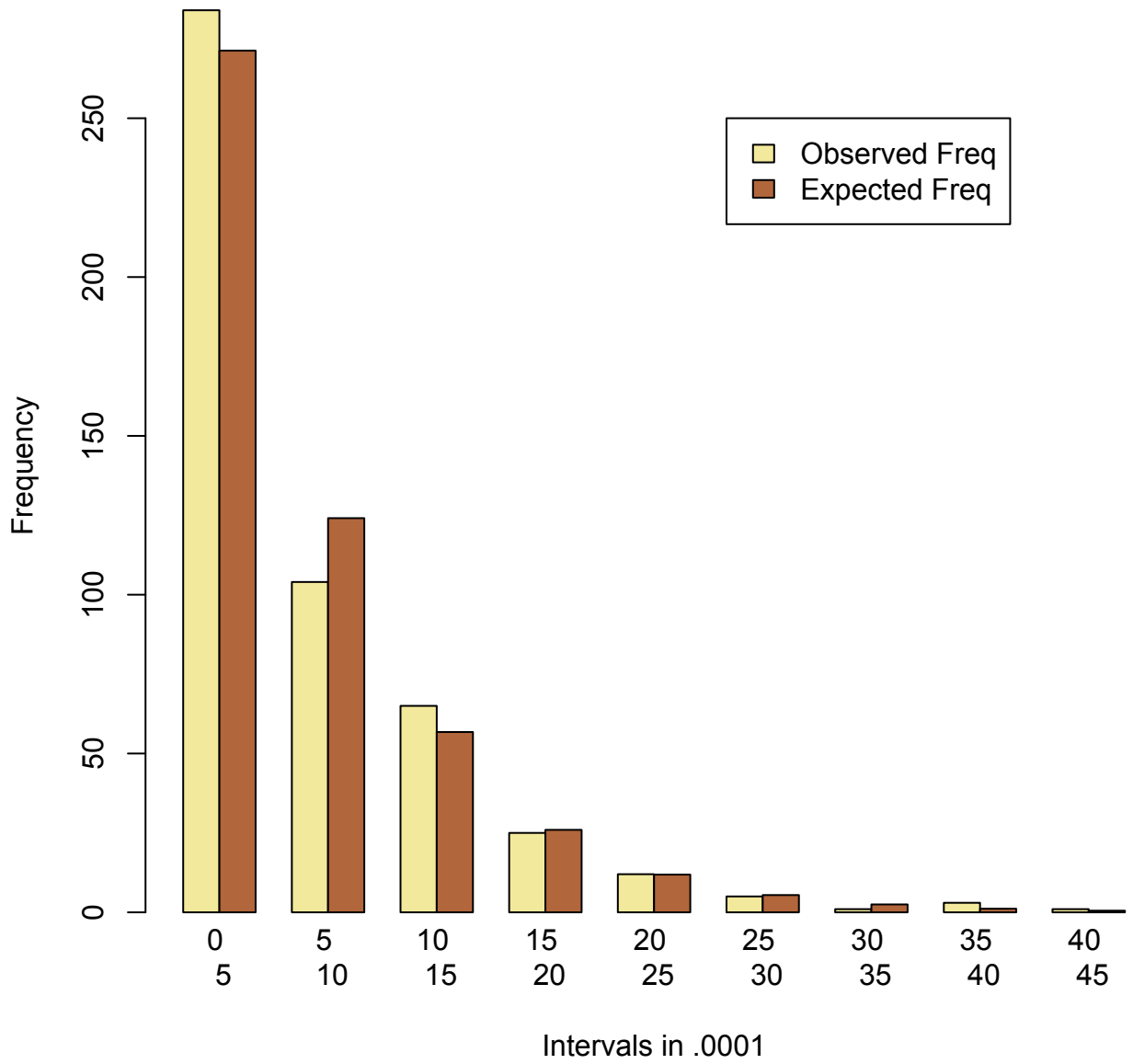
```

> ##### FIND EXPECTED FREQ AND MAKE COMPARATIVE PLOT #####
>
> h$breaks
[1] 0.0000 0.0005 0.0010 0.0015 0.0020 0.0025 0.0030
[8] 0.0035 0.0040 0.0045
> colnames(t)<-c("Time -","Interval","Obs.Freq", "Exp.Freq")

> dx <- h$breaks[2]; dx
[1] 5e-04
> ep <- pexp((1:9)*dx,lambda)-pexp((0:8)*dx,lambda); ep
[1] 0.542561597 0.248188510 0.113530956 0.051933419
[5] 0.023756340 0.010867062 0.004971012 0.002273932
[9] 0.001040184
> ec <- sc*ep
> t <- cbind(h$breaks[-(nt+1)],h$breaks[-1],h$counts,round(ec,3))
> colnames(t)<-c("Time -","Interval","Obs.Freq", "Exp.Freq")
> t
      Time - Interval Obs.Freq Exp.Freq
[1,] 0.0000   0.0005      284  271.281
[2,] 0.0005   0.0010      104  124.094
[3,] 0.0010   0.0015       65   56.765
[4,] 0.0015   0.0020       25   25.967
[5,] 0.0020   0.0025       12   11.878
[6,] 0.0025   0.0030        5    5.434
[7,] 0.0030   0.0035        1    2.486
[8,] 0.0035   0.0040        3    1.137
[9,] 0.0040   0.0045        1    0.520
>
>
> xn <- paste(seq(0,40,5),"\n",seq(5,45,5));xn
> barplot(rbind(h$counts, ec), beside=T, col=c("khaki","sienna"),
+ names = xn, xlab = "Intervals in .0001", ylab = "Frequency")
> title(paste("Histogram of", n, "Closest Distances^2"))
> legend(16, 250, c("Observed Freq", "Expected Freq"),
+ fill = c("khaki", "sienna"), bg = "white")
> # M3074Skylark5.pdf

```

### Histogram of 500 Closest Distances<sup>2</sup>





```
> ##### CHI-SQUARED TEST TO SEE IF EXPONENTIAL HOLDS #####
> #
> # Small counts means bins should be combined, but we don't bother.
> # Anyway, test shows no strong evidence to reject dist is exponential.
```

```
> chisq.test(h$counts,ep)
```

Pearson's Chi-squared test

```
data: h$counts and ep
X-squared = 63, df = 56, p-value = 0.2425
```

Warning message:

```
In chisq.test(h$counts, ep) : Chi-squared approximation may be incorrect
>
```