Intensity estimation for Poisson processes

Ludwik Czeslaw Drazek
Student number 200750924

Supervised by Dr Jochen Voß

Submitted in accordance with the requirements for the
module MATH5871M: Dissertation in Statistics
as part of the degree of

Master of Science in Statistics

The University of Leeds, School of Mathematics

September 2013

The candidate confirms that the work submitted is his/her own and that appropriate credit has been given where reference has been made to the work of others.

Abstract

This work investigates the modelling of data by a non-homogeneous Poisson process. The mathematical theory behind the Poisson distribution is introduced, this leads to the homogeneous Poisson process. The non-homogeneous Poisson process is developed as a generalisation of the homogeneous case. The theory behind the estimation of the non-homogeneous intensity function is developed. Throughout, R is used as the statistical software to graphically and numerically described the data and as the programming language to estimate the intensity functions. Several classes of intensity functions are considered and the parameters are found by maximum likelihood estimation. The resulting models are found to fit the data fairly well.

Acknowledgments

I wish to acknowledge the invaluable help from my supervisor Dr Jochen Voß, who's knowledge and experience far exceeds mine, corrected my many R and LaTeXproblems and provided most of the theoretical guidance.

I thank the EPSRC for providing me with the financial aid to study the MSc in Statistics.

Contents

1 Introduction **1**

1.1 Notation 2

2 The Poisson Distribution **5**

2.1 Introduction 5

2.2 Estimate for the Parameter (Intensity) 5

2.3 Confidence Interval for the Intensity 6

2.3.1 Lower and upper confidence bounds for the intensity 7

2.3.2 Numerical methods for confidence intervals 9

2.3.3 Varying Rate example 17

3 Point Processes **21**

3.1 Duality 21

3.2 Memoryless property 21

3.3 Moment generating functions 22

4 Homogeneous Poisson Process **25**

4.1 Equivalent definitions 25

4.2 Equivalence of the definitions 26

4.3 Length of observation interval 28

5 Non-Homogeneous Poisson Process **31**

5.1 Motivation 31

5.2 Maximum Likelihood Estimation of the intensity 32

5.3 Double checking the theory 33

5.4 Fitting an intensity function 33

5.4.1 Numerical Integration in R 34

5.4.2 Numerical Optimisation in R 34

6 Results **37**

7 Further work **63**

8 Program code **65**

List of Figures

2.1 Confidence intervals for the mean of a Poisson distribution, from a single sample. The horizontal axis gives the value of the sample, the vertical axis gives the corresponding confidence interval. 12

2.2 Confidence intervals with a confidence coefficient of 99% are wide and all except one overlap from one time period to the next. 13

2.3 Confidence intervals with a confidence coefficient of 95% are narrower but still fairly wide. 14

2.4 Confidence intervals with a confidence coefficient of 50% are much narrower and the appearance of the graph is very different from the previous one. 15

2.5 Confidence intervals with a confidence coefficient of 10% are very narrow and the intensity function looks almost constant from interval to interval, except for one high outlier. 16

2.6 Confidence intervals as the poisson mean varies as a Sine wave. 19

6.1 Confidence intervals with a daily grouping. No obvious linear trend or any other pattern. Could be a homogeneous Poisson process. 39

6.2 Confidence intervals with a 12 hour grouping. A lot more variability. A suggestion of periodicity at the lower end ? 40

6.3 Confidence intervals with a 8 hour grouping. The data sees to be splitting itself into two large groups with outliers at the top. 41

6.4 Confidence intervals with a 6 hour grouping. More clear evidence of data splitting into two groups. 42

6.5 Confidence intervals with a 4 hour grouping. Even more evidence of two groups with a hint that the top group may be periodic. 43

6.6 Confidence intervals with a 3 hour grouping. The upper group is looking more and more periodic. 44

6.7 Confidence intervals with a 2 hour grouping. The two group periodic pattern is still there with some intermediate points. 45

6.8 Confidence intervals with a 1 hour grouping. It is now clear that we have periodic peaks. 46

6.9 Confidence intervals with a 1 day grouping and a homogeneous rate fitted in red. n=0. 48

6.10 Confidence intervals with a 1 hour grouping and a homogeneous rate fitted in red. n=0. 49

6.11 Confidence intervals with a 1 day grouping and an intensity function with one Fourier level fitted in red. The one day grouping is not fine enough to show any periodicity. n=1. 50

6.12 Confidence intervals with a 12 hour grouping. We are starting to see the peaks and troughs in the data matching the peaks and troughs in the model. n=1. . . . 51

6.13 Confidence intervals with a 8 hour grouping. It is now clear that the peaks and troughs in the data matching the peaks and troughs in the model. n=1. 52

6.14 Confidence intervals with a 6 hour grouping. We can still see the peaks and troughs in the data matching the peaks and troughs in the model although the random variability in the data is starting to mask it. n=1. 53

6.15 Confidence intervals with a 12 hour grouping. The graph is very cramped, it is difficult to see the fitted model clearly. n=2. 55

6.16 Confidence intervals with a 12 hour grouping. We are starting to see the peaks and troughs in the data matching the peaks and troughs in the model. n=2. . . . 56

6.17 Confidence intervals with a 8 hour grouping. It is now clear that the main peaks and troughs in the data matching the peaks and troughs in the model. The secondary troughs are a better fit than the secondary peaks. Note the two high outliers corresponding to the secondary peaks! n=2. 57

6.18 Confidence intervals with a 6 hour grouping. The main peaks match best the secondary peaks correspond to three high outliers. The troughs fitting is average. n=2 58

6.19 Confidence intervals with a 4 hour grouping. Both the main and secondary peaks and troughs correspond to the data. n=2. 59

6.20 Confidence intervals with a 4 hour grouping. The main peaks and troughs correspond to the data. The secondary troughs are also a fairly good fit. The secondary peaks are a worse fit. It is very difficult to see how the tertiary peaks and troughs match the data. n=3. 60

6.21 Confidence intervals with a 3 hour grouping. With the data rectangles more extreme is is easier to see that the main peaks and troughs correspond to the data. The secondary troughs are a fairly good fit. The secondary peaks are a worse fit. It is very difficult to see how the tertiary peaks and troughs match the data. n=3. 61

List of Tables

6.1 Table of estimated parameter values for the exponential Fourier model. 54

Chapter 1

Introduction

This work is concerned with fitting a non-homogeneous Poisson process to data. The non-homogeneous process is a generalisation of the homogeneous process, which itself uses the Poisson distribution as part of it's definition. The homogeneous Poisson process is fundamental. It is very simple, in one sense defines randomness (to be explained later), it is easy to manipulate mathematically and various extensions give rise to other stochastic processes which are more realistic.

When modelling data with most mathematical and statistical models there are usually unknown parameters in these models which need to be estimated. This estimation usually involves the data. The Poisson distribution has a single parameter. For the simple homogeneous Poisson process the unknown parameter is the rate of occurrence, this is a single real number. There are several common accepted methods of estimating these parameters: method of moments (Dudewicz and Mishra, 1988, p. 362-7), least squares and the method of maximum likelihood is very popular due to favourable theoretical properties (Dudewicz and Mishra, 1988, p. 347-362).

Modelling the non-homogeneous Poisson process is more complex. Not only are there unknown parameters which need estimating, we also need to decide on the functional form of the intensity function. In this work four classes of functions were considered. These were:

1. Polynomials of the form $\lambda(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$ with the restriction that $\lambda(x) \geq 0 \quad \forall x$.
2. Fourier(periodic) series $\lambda(x) = a_0 + \sum_{j=1}^{n} b_j \sin(2\pi \frac{x}{T} f_j) + c_j \cos(2\pi \frac{x}{T} f_j)$ with the restriction that $\lambda(x) \geq 0 \quad \forall x$.
3. Exponentials of polynomials $\lambda(x) = \exp(a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n)$
4. Exponentials Fourier series $\lambda(x) = \exp\Big(a_0 + \sum_{j=1}^{n} b_j \sin(2\pi \frac{x}{T} f_j) + c_j \cos(2\pi \frac{x}{T} f_j)\Big)$

Each function has unknown parameters. Maximum Likelihood Estimation (MLE) is used to estimate the parameters.

This work contains chapters on the Poisson distribution where the distribution is defined, the theory about confidence intervals is introduced. R programs which calculate the confidence intervals based on an observation from a Poisson distribution are included.

A chapter on point processes, which includes theory on general point process as well as results on the Poisson process in particular.

A chapter on the homogeneous Poisson process showing how four definitions of it are equivalent.

A chapter on the non-homogeneous Poisson process focusing on the theoretical MLE of the intensity function and how it can be implemented in practice using the R language.

A final chapter on the results as we fit the various models to actual data.

1.1 Notation

This is the default notation that will be used throughout this work. Exceptions to this notation will be clearly marked and explained in the text.

t, s will refer to time. By default $s < t$

log will represent the natural logarithm to base e

$\{N(t), t \geq 0\}$ is a counting process. Usually starting at time $t = 0$, so $N(0) = 0$

λ will in general refer to the parameter of both the Poisson processes and the various distributions such as the exponential, gamma and Poisson.

μ will be reseserved to represent the mean or expected value of some random process.

The Poisson distribution with parameterλ will be written as $Pois(\lambda)$ where,

$$P(X = k) = \frac{e^{-\lambda}\lambda^k}{k!}, k = 0, 1... \tag{1.1}$$

The Exponential distribution with parameter λ will be written as $M(\lambda)$ where,

$$f(t) = \lambda e^{-\lambda t}, t \geq 0 \tag{1.2}$$

The Gamma distribution with parameters n and λ will be written as $\Gamma(n, \lambda)$ where,

$$f(t) = \frac{t^{n-1}\lambda^n e^{-\lambda t}}{(n-1)!}, t \geq 0 \tag{1.3}$$

Chapter 2

The Poisson Distribution

2.1 Introduction

This chapter describes the Poisson distribution where it is defined, the theory about confidence intervals is introduced. R programs which calculate the confidence intervals based on an observation from a Poisson distribution are included.

The Poisson distribution is attributed to the French mathematician Simeon Denis Poisson (1781-1840) who introduced it in 1837.

The Poisson distribution is a discrete distribution defined on non-negative integers, it has a single parameter say λ, and will be written as $Pois(\lambda)$ where,

$$P(X = k) = \frac{e^{-\lambda}\lambda^k}{k!}, k = 0, 1... \tag{2.1}$$

It can easily be shown (Dudewicz and Mishra, 1988, p. 221-222) that the expected value of X, $E(X) = \lambda$ and the variance $var(X) = \lambda$. Since $E(X) = \lambda, \lambda$ the parameter is also a measure of the intensity or magnitude of the value of the r.v. X.

2.2 Estimate for the Parameter (Intensity)

Suppose we have a single observation from the Poisson distribution, say m and we want to estimate the value of the intensity λ based on the value of this observation. One method of doing this is maximum likelihood. We find the value of λ which maximises the likelihood of the observation occurring.

Although the Poisson distribution is discrete from the point of view of the random variable, it is both continuous and differentiable with respect to the parameter λ when the random variable is kept constant, so it is possible to use calculus to maximise the likelihood.

Suppose we observe no event, then $(m = 0)$ the likelihood of this happening is using 2.1 $P(X = 0) = e^{-\lambda}$.

This equation has no turning point and is maximised on the boundary when $\lambda = 0$.

When $(m > 0)$ equation 2.1 is differentiable as a function of λ and standard calculus can be used to find the maximum likelihood estimate of λ, this turns out to be m itself (Beaumont, 1980, p. 89-90). This estimate of λ turns out to be an unbiased estimate of λ.

2.3 Confidence Interval for the Intensity

Instead of just having a single estimate of λ it is more useful to have an interval estimate for λ. This interval will be defined by a lower bound(limit) and an upper bound(limit).

One way of finding these lower and upper bounds is to consider a duality with hypotheses tests (Dudewicz and Mishra, 1988, p. 549-552) contains a very good abstract description based on a generic hypotheses test and it's equivalence to interval estimation. Based on the observed data "m" we accept the null hypothesis that $H_0 : \lambda = \lambda_0$ when m falls within certain bounds with probability $(1 - \alpha)$, we call this the confidence coefficient. And we reject the null hypotheses when m falls outside these limits with probability α.

Suppose, we have a simple null hypotheses $H_0 : \lambda = \lambda_0$ and a composite alternative $H_1 : \lambda \neq \lambda_0$ we reject H_0 if m falls in a critical region C_λ , let D_λ be it's complement.

If $P_\lambda(m \in C_\lambda) = \alpha$ then $P_\lambda(m \in D_\lambda) = 1 - \alpha$.

The significance level of the hypotheses test is $\alpha, (1 - \alpha)$ is the confidence coefficient.

Let $S(m) = \{\lambda^* : m \in D_{\lambda^*}\}$.

In other words $S(m)$ is the collection of all λ^*'s for which the observation m does not belong in the rejection region of H_0.

So, $S(m)$ is a set of λ^*'s which can reasonably give rise to the observation m. Thus given m, or if m is observed then $S(m)$ is the set of λ's which can reasonably give rise to m.

We thus call $S(m)$ a confidence set for λ with confidence coefficient $(1 - \alpha)$.

An informal way of explaining how to obtain these lower and upper bounds is: (Daly et al., 1995, p. 269-271) consider an upper bound for λ, if the true value λ_u is very high and above its upper bound (where the upper bound is determined from the data m) then there should be a very low probability of obtaining the actual observed value of m. So if the true value of λ is λ_u and the probability of obtaining a value as low as m or lower is $< \alpha/2$. The actual upper limit for λ is the value where the probability of getting a value as low as m or lower is equal to $\alpha/2$.

This motivates the definition of function f below.

Consider a lower bound for λ, if the true value λ_l is very low and below its lower bound (where the lower bound is determined from the data m) then there should be a very low probability of obtaining the actual observed value of m. So if the true value of λ is λ_l and the probability of obtaining a value as high as m or higher is $< \alpha/2$. The actual lower limit for λ is the value where the probability of getting a value as high as m or higher is equal to $\alpha/2$. This motivates the definition of function g below.

An alternative and equivalent explanation is: Based on the observed data m, suppose we have lower and upper bounds $L(m), U(m)$ and these define an $(1-\alpha)\%$ symmetrical confidence interval for the unknown parameter λ. When λ is at the boundary points the probability of getting the original data m or more extreme is actually equal to $\alpha/2$. So if $\lambda = L(m)$ then $P(X \geq m) = \alpha/2$. If $\lambda < L(m)$ then $P(X \geq m) < \alpha/2$. So if $\lambda = U(m)$ then $P(X \leq m) = \alpha/2$. If $\lambda > U(m)$ then $P(X \leq m) < \alpha/2$.

2.3.1 Lower and upper confidence bounds for the intensity

Definition 1. *Let* $0 < \alpha \leq 1$ *and* $X \sim Pois(\lambda)$. *Define functions* f *and* g *by*

$$f(\lambda) = P_\lambda(X \leq m) = \sum_{k=0}^{m} e^{-\lambda} \frac{\lambda^k}{k!}$$

and

$$g(\lambda) = P_\lambda(X \geq m) = \sum_{k=m}^{\infty} e^{-\lambda} \frac{\lambda^k}{k!}$$

for all $\lambda > 0$. *For a symmetrical confidence interval we define* U *or* $U(m)$ *by* $f(U(m)) = \frac{\alpha}{2}$ *and L or L(m) by* $g(L(m)) = \frac{\alpha}{2}$.

The definition of U is equivalent to:

$$P_{U(m)}(X \leq m) = \sum_{k=0}^{m} e^{-U(m)} \frac{(U(m))^k}{k!} = \frac{\alpha}{2}.$$

The definition of L is equivalent to:

$$P_{L(m)}(X \geq m) = \sum_{k=m}^{\infty} e^{-L(m)} \frac{(L(m))^k}{k!} = \frac{\alpha}{2}.$$

Suppose we want a $(1-\alpha)\%$ confidence interval(symmetrical if possible) for a parameter λ of a Poisson distribution.

Proposition 1. *If* $X \sim pois(\lambda)$ *then, as* λ *increases,* $P_\lambda(X \leq a)$ *decreases and* $\lim_{\lambda \to \infty} P_\lambda(X \leq a) = 0$ *for all* a.

Proof. We have

$$P_\lambda(X \leq a) = \sum_{k=o}^{a} e^{-\lambda}\frac{\lambda^k}{k!} = e^{-\lambda} + \lambda e^{-\lambda} + \frac{\lambda^2}{2!}e^{-\lambda} + ... + \frac{\lambda^a}{a!}e^{-\lambda}.$$

When $\lambda = 0, P_\lambda(X \leq a) = 1$ for all $a \in \{0, 1, 2....\}$. $P_\lambda(X \leq a)$ is differentiable in λ so differentiating it:

$$\begin{aligned}\frac{dP_\lambda(X \leq a)}{d\lambda} &= -e^{-\lambda} \\ &\quad + (e^{-\lambda} - \lambda e^{-\lambda}) \\ &\quad + (\lambda e^{-\lambda} - \frac{\lambda^2}{2!}e^{-\lambda}) \\ &\quad + (\frac{\lambda^2}{2!}e^{-\lambda} - \frac{\lambda^3}{3!}e^{-\lambda}) \\ &\vdots \\ &\quad + (\frac{\lambda^{(a-2)}}{(a-2)!}e^{-\lambda} - \frac{\lambda^{(a-1)}}{(a-1)!}e^{-\lambda}) \\ &\quad + (\frac{\lambda^{(a-1)}}{(a-1)!}e^{-\lambda} - \frac{\lambda^a}{a!}e^{-\lambda}).\end{aligned} \tag{2.2}$$

These terms cancel out pairwise term by term until only the final term is left.

$$\frac{dP_\lambda(X \leq a)}{d\lambda} = -\frac{\lambda^a}{a!}e^{-\lambda} < 0$$

So $P_\lambda(X \leq a)$ is monotonically decreasing as λ increases.
Also, we have since exponentials dominate finite powers as $\lambda \to \infty$.

$$\lim_{\lambda\to\infty} P_\lambda(X \leq a) = \lim_{\lambda\to\infty} \frac{\sum_{k=0}^{a} \frac{\lambda^k}{k!}}{e^\lambda} = 0.$$

□

Proposition 2. *$U(m)$ is increasing as m increases.*

Proof. Let $m_1 > m_2$. $U(m_1)$ is defined by:

$$\begin{aligned}&\sum_{k=0}^{m_1} e^{-U(m_1)}\frac{(U(m_1))^k}{k!} = \frac{\alpha}{2} \\ &\Rightarrow \sum_{k=0}^{m_2} e^{-U(m_1)}\frac{(U(m_1))^k}{k!} + \sum_{k=m_2+1}^{m_1} e^{-U(m_1)}\frac{(U(m_1))^k}{k!} = \frac{\alpha}{2}\end{aligned} \tag{2.3}$$

The second summation in the above equation is positive. So the first summation is:

$$\sum_{k=0}^{m_2} e^{-U(m_1)}\frac{(U(m_1))^k}{k!} < \frac{\alpha}{2}$$

¿From definition 1 we know that:

$$\sum_{k=0}^{m_2} e^{-U(m_2)} \frac{(U(m_2))^k}{k!} = \frac{\alpha}{2}$$

Both the above equations represent cumulative Poisson probabilities up to m_2. From proposition 1, we know that these cumulative probabilities decrease as the parameter increases. Since the first summation is $< \alpha/2$. The parameter $U(m_1)$ must be $> U(m_2)$. Thus $U(m)$ increases as m increases. □

It can be shown that as $m \to \infty, U(m) \to \infty$.

2.3.2 Numerical methods for confidence intervals

Suppose we now observe n objects.
When $n = 0$ we have a special case. $P(N = (n = 0)) = e^{-\lambda}$
The MLE of λ is zero. And the lower confidence limit, $L(0) = 0$ since λ cannot be negative, the lower limit is equal to the estimate, to keep the confidence coefficient at $\alpha\%$ we define a one sided confidence interval. To find $U(0)$ we use α rather than $\frac{\alpha}{2}$.
For $U(0)$, we want to find U such that $P(N \leq 0) = P(N = 0) = e^{-U} = \alpha$.
If $\alpha = 0.05$ we get $U(0) = 2.996$ as the solution.
A 95% CI is thus [0, 2.996]
An $(1 - \alpha)\%$ CI is thus $[0, -log(\alpha)]$

When $n = 1$, we need to find $L(1)$ such that $P(N \geq 1) = 1 - P(N = 0) = 1 - e^{-L} = \frac{\alpha}{2}$.
Solution is $L = -log(1 - \alpha/2)$
$\alpha/2 = 0.025$ gives $L = 0.0253$

For the upper limit we want U such that $P(N \leq 1) = e^{-U} + Ue^{-U} = \alpha/2$
Solving this analytically is difficult, using numerical methods gives the answer $U = 5.572$ when $\alpha/2 = 0.025$
A 95% CI is thus $[0.025, 5.572]$

For values of $n \geq 2$ we need to use numerical methods. R is the statistical software package used. the two R routines being used are `ppois` and `uniroot`.

For the lower limit, given an observed value of n, we want to find a value L such that $P(N \geq n) = P(N > (n - 1)) = \alpha/2$. This is equivalent to finding a root of the equation $P(N > (n - 1)) - \alpha/2 = 0$. Due to the way 'R' defines the function `ppois`.
To use uniroot, we define a function fl,

```
fl <- function(U,m,a)  ppois((m-1),U, lower.tail=FALSE) - a/2
```

uniroot finds a numerical root of an equation however it expects to be given two estimates of the root, one an under-estimate and an over-estimate. For the lower confidence limit, the under-estimate is zero and the over-estimate is the actual MLE of the parameter.

For the upper limit , given an observed value of n, we want to find a value U such that $P(N \leq n) = \alpha/2$. In R this condition is `ppois(n,U) = α/2`.

Thus we want to find the zero of the equation ppois(n,U) - $\alpha/2$=0.
To use uniroot, we define a function fu,

```
fu <- function(U,m,a)  ppois(m,U) - a/2
```

To call uniroot in a specific case, for example we observe a value of n and the confidence coefficient is 95% we need to specify lower and upper estimates for U. The lower limit is the MLE estimate, which is n itself. The upper limit can be found by using Chebyshev's inequality (Ross, 2000, p. 74). Chebyshev's inequality is a genral inequality. Being a very general inequality It should give a larger value than the one we are seeking and will be used as the upper bound in the estimate of U in the root finding routine uniroot. The mathematical derivation of this upper bound using Chebyshev's is below.

Chebyshev's inequality states that for any random variable X with mean μ and variance σ^2, then for any $k > 0$,

$$P\{|X - \mu| \geq k\} \leq \frac{\sigma^2}{k^2}.$$

If $X \sim pois(\lambda)$, then $\mu = E(X) = \lambda$ and $\sigma^2 = var(X) = \lambda$. Substituting these into Chebyshev's inequality we get.

$$P\{|X - \lambda| \geq k\} \leq \frac{\lambda}{k^2}.$$

The condition $|X - \lambda| \geq k$ is equivalent to: $X \in (-\infty, (\lambda - k)] \cup [(\lambda + k), \infty)$.
These are two disjoint events. Thus:

$$\begin{aligned}
&P(|X - \lambda| \geq k) = P(X \leq (\lambda - k)) + P(X \geq (\lambda + k)) \\
&P(|X - \lambda| \geq k) - P(X \leq (\lambda - k)) = P(X \geq (\lambda + k)) > 0 \\
&P(|X - \lambda| \geq k) - P(X \leq (\lambda - k)) > 0 \\
&P(|X - \lambda| \geq k) > P(X \leq (\lambda - k)) \\
&P(X \leq (\lambda - k)) < P(|X - \lambda| \geq k) \leq \frac{\lambda}{k^2} \\
&P(X \leq (\lambda - k)) < \frac{\lambda}{k^2}
\end{aligned} \tag{2.4}$$

We are free to choose k, so let $k = (\lambda - m) \Longleftrightarrow (\lambda - k) = m, \forall_\lambda$ with $\lambda > m$.

$$\Rightarrow P(X \leq m) < \frac{\lambda}{(\lambda - m)^2}$$

If we solve $\frac{\lambda}{(\lambda - m)^2} = \frac{\alpha}{2}$ The resulting λ can be used as the upper over-estimate for $U(m)$ that is supplied to uniroot. This is quadratic in λ the solution is $\lambda = m + \frac{1+\sqrt{2m\alpha+1}}{\alpha}$.

The following 'R' code routines calculate the upper and lower limits for λ, given an observation m. The original functions fl ans fu were modified to work with the routines GetConfidenceInterval.

```
GetConfidenceInterval <- function(x, alpha=0.05) {
        if (x == 0) {
                L <- 0
                U <- - log(alpha)
        } else {
                fl <- function(lambda) {
                        ppois(x, lambda) - alpha/2
                }
                lambda.upper <- x + (1 + sqrt(2*x*alpha + 1)) / alpha
                U <- uniroot(fl, c(x, lambda.upper))$root
                fu <- function(lambda) {
                        ppois(x - 1, lambda, lower.tail=FALSE) - alpha/2
                }
                L <- uniroot(fu, c(0, x))$root
        }
        return(c(L,U))
}
```

To display the confidence intervals for various observations for example 0 to 100, I used the following 'R' code.

```
m=100
temp <-c()
LU = matrix(nrow=(m+1), ncol=2, byrow=TRUE)
for (k in 0:m){
   temp=GetConfidenceInterval(k)
   LU[(k+1),1]=temp[1]
   LU[(k+1),2]=temp[2]
}
x<-seq(0,m)
plot(x,x,type="l",xlab="Observed value",
ylab="Upper and Lower confidence bounds")
```

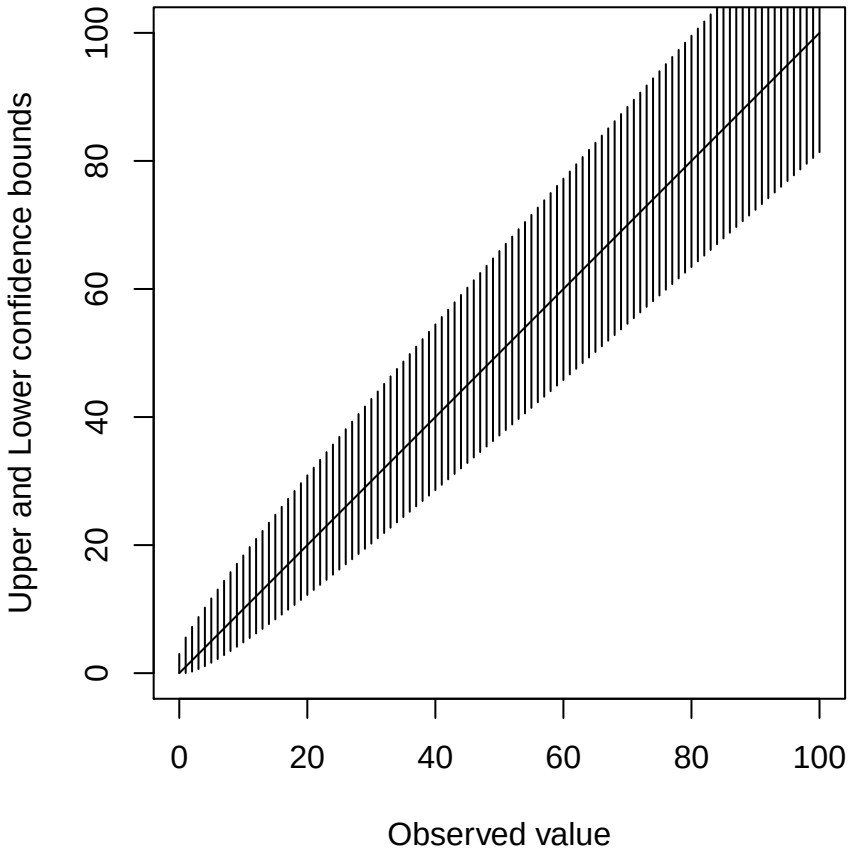

Figure 2.1: Confidence intervals for the mean of a Poisson distribution, from a single sample. The horizontal axis gives the value of the sample, the vertical axis gives the corresponding confidence interval.

```
for (k in 1:(m+1)){
    segments(x0=x[k],y0=LU[k,1],x1=x[k],y1=LU[k,2])
}
```

A sample plot is shown in figure 2.1. This graph very clear, the lower and upper confidence bounds are on either side of the estimate and the range increases (confidence interval length) increases with the observed value.

However an exploratory analysis was done on a sample of real data and is shown graphically, in figures 2.2, 2.3, 2.4, 2.5 This shows how the confidence intervals appear as we decrease the confidence coefficient. This indicates that the graphs appearance changes as we alter the confidence coefficient. Thus to discover the essential features of the data clearly there is a need to experiment with different values of the confidence coefficient, this facility is provided in the R routines that were written here.

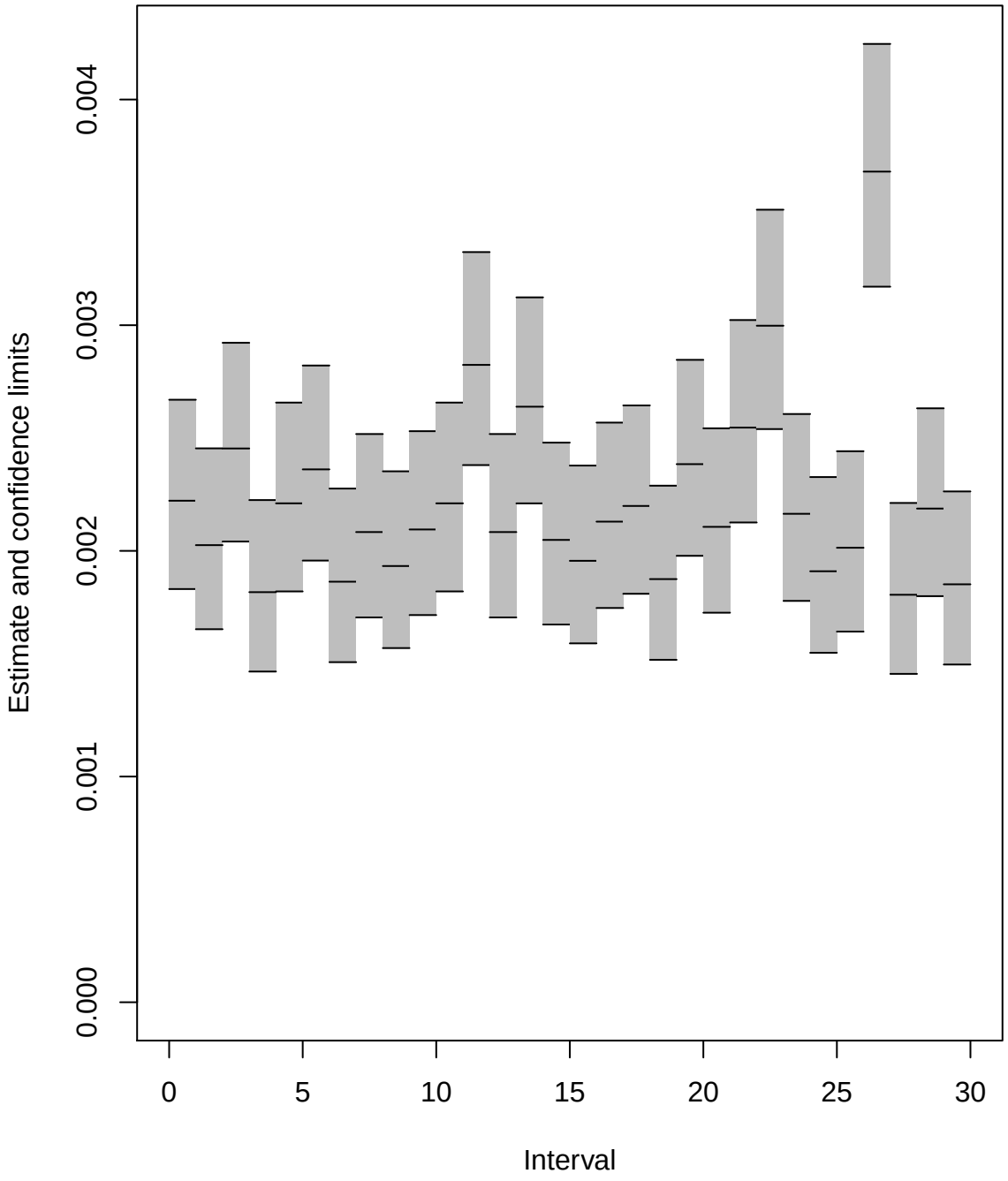

Figure 2.2: Confidence intervals with a confidence coefficient of 99% are wide and all except one overlap from one time period to the next.

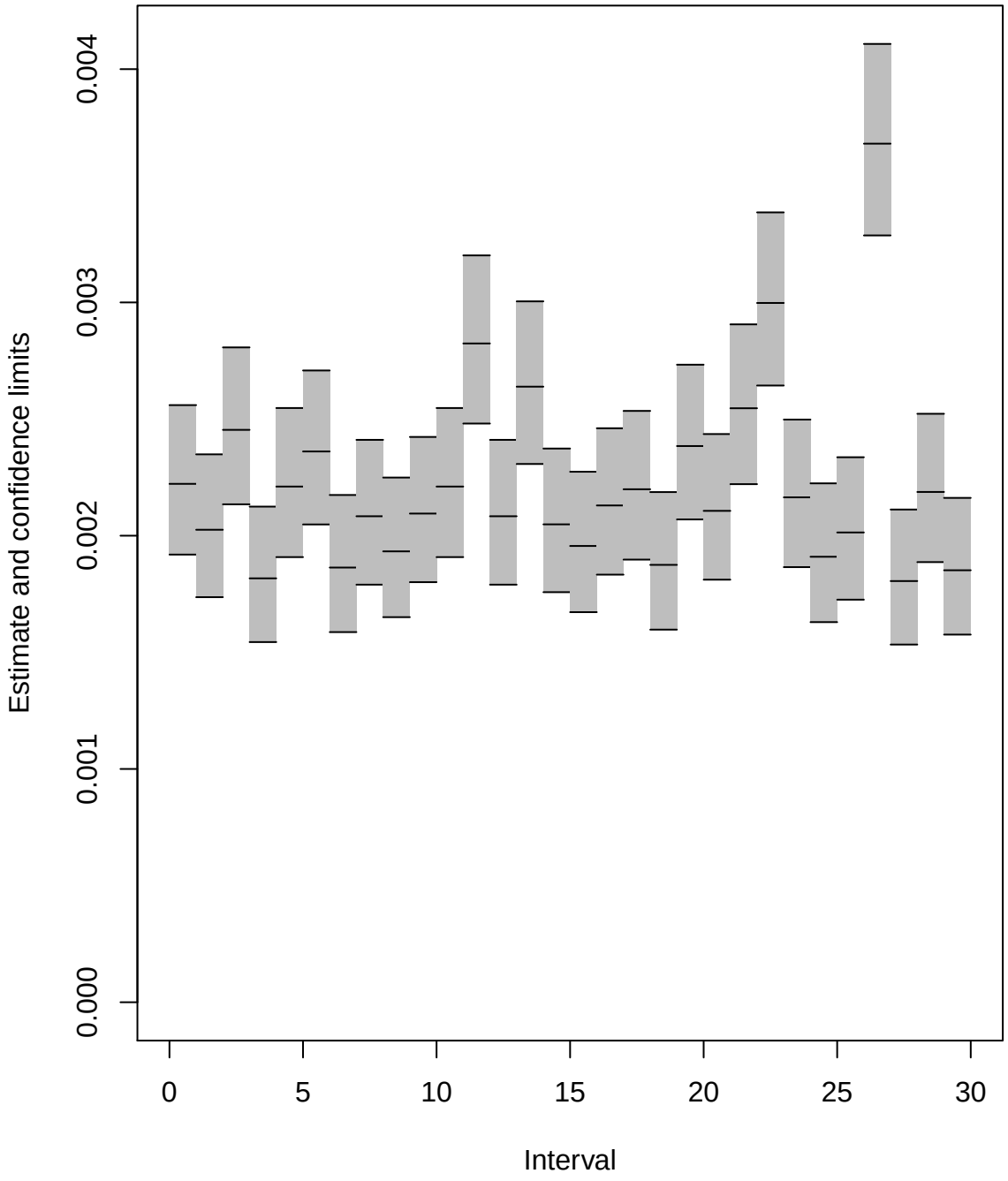

Figure 2.3: Confidence intervals with a confidence coefficient of 95% are narrower but still fairly wide.

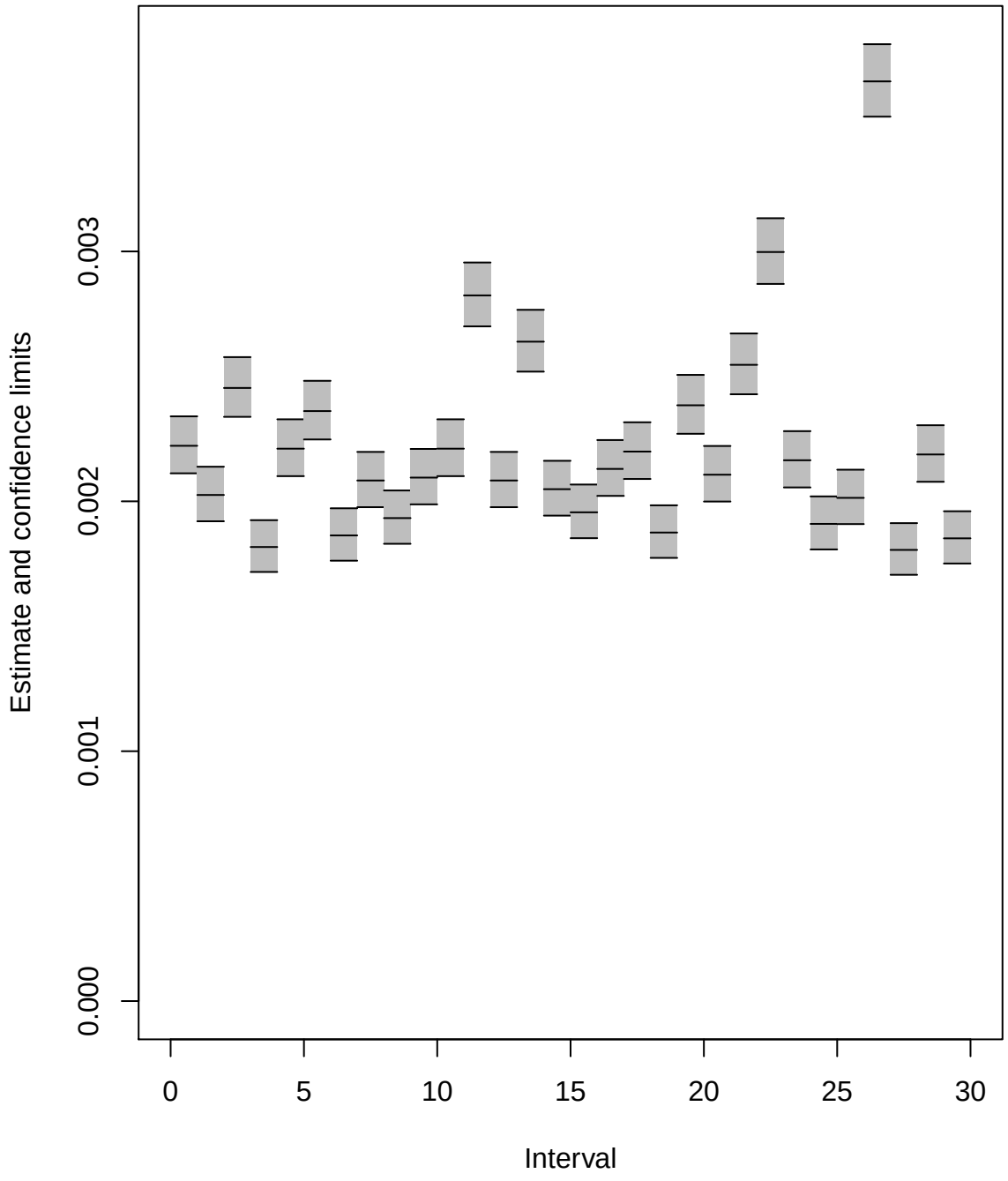

Figure 2.4: Confidence intervals with a confidence coefficient of 50% are much narrower and the appearance of the graph is very different from the previous one.

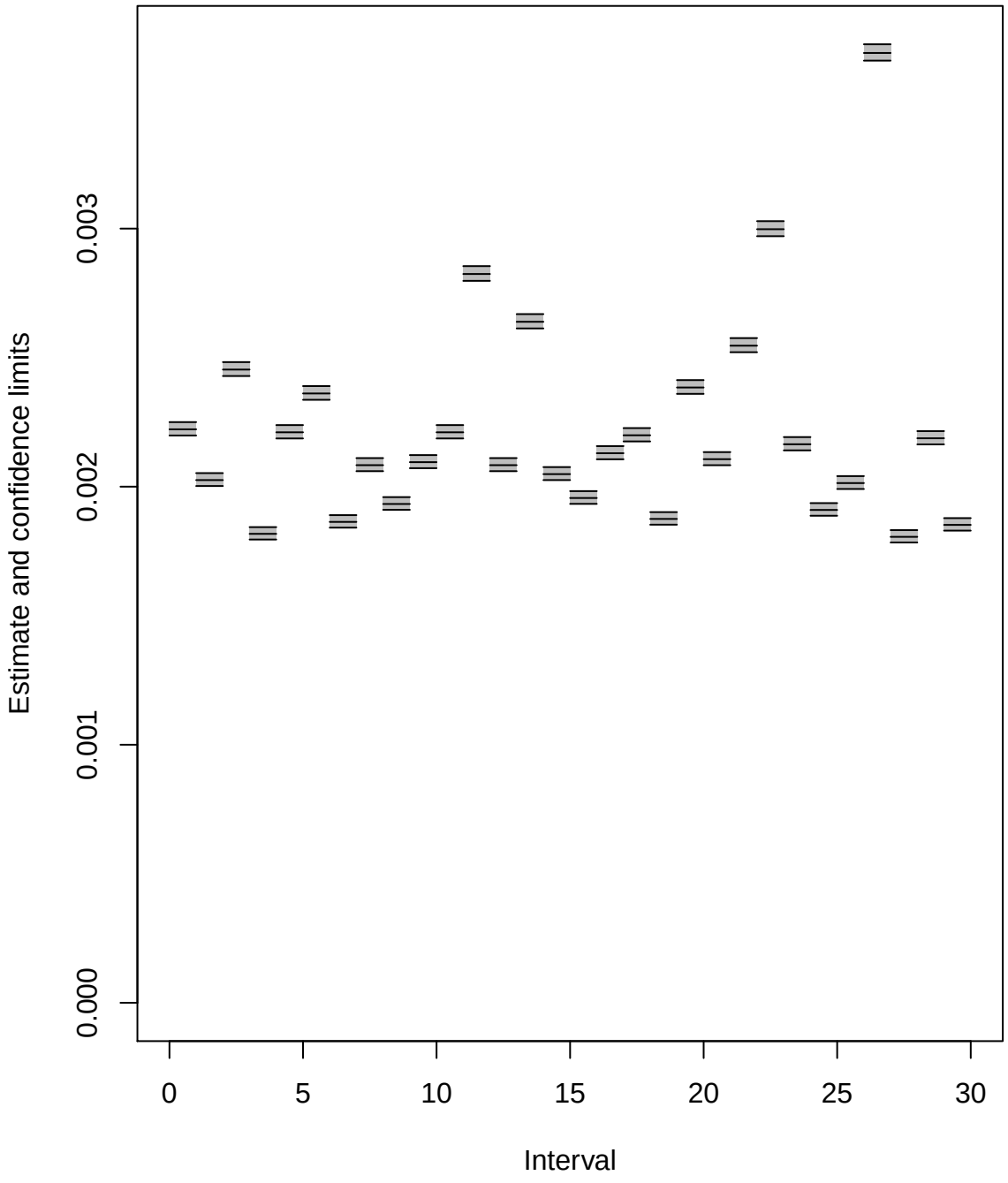

Figure 2.5: Confidence intervals with a confidence coefficient of 10% are very narrow and the intensity function looks almost constant from interval to interval, except for one high outlier.

2.3.3 Varying Rate example

Suppose that the rate of occurrence is not constant but varies through time. In this example we assume that the rate varies according to a full cycle of a Sine function. The first step is to generate/simulate data. The R function `GeneratePoissonSample` uses rejection sampling to simulate data from a Poisson distribution where the rate changes. We assume a rate of 100 per unit time, the variable `N` is the number of occurrences in a time interval length `(b-a)`. The variable `X` generates the times of the occurrences. It is well known that if we know that a fixed number of events have occurred in a given time, then the actual times of these events are uniformly distributed over that time interval (Ross, 2000, p. 270-272). The variable `X` contains the times. `Y` is a vector which enables us to use rejection sampling (Morgan, 1984, p. 98-107) (Ross, 2000, p 591-595) to either accept or reject the times in `X` according to the rate varying as a Sine function.

```
GeneratePoissonSample <- function(a, b) {
   N <- rpois(1, 100*(b-a))
   X <- runif(N, a, b)
   Y <- runif(N, 0, 2)
   return(sort(X[which(Y <= sin(2*pi*(X-a)/(b-a)) + 1)]))
}
```

The `R` function `IntervalCount` splits the interval `(b-a)` into `n` intervals and counts how many events fall into each interval. `Lud1` is passed in as the vector containing the times of events. The function outputs `Lud2` as the counts.

```
IntervalCount <- function(Lud1,n,a,b) {
   Lud2 <- c()
   if (n==1) {Lud2[1] = length(Lud1)}
   else {for (i in 1:n) {
      lower = a +(i-1)*(b-a)/n
      upper = lower + (b-a)/n
      Lud2[i]=0
      for (j in 1:length(Lud1)){
         if((Lud1[j]>=lower)&&(Lud1[j]<upper)){
            Lud2[i]=Lud2[i]+1}
              }
            }
      }
      return(Lud2)
   }
```

The function `ConfidenceIntervals2` takes the output from the function `IntervalCount` `"Lud2"` and returns the confidence intervals in `Lud3`.

```
ConfidenceIntervals2 <- function(lud2){
#Input a vector of counts
#Output the lower and upper confidence limits for these counts
   U <- c()
   L <- c()

   for (i in 1:length(lud2)){
      a=lud2[i]
      UL =GetConfidenceInterval(a)
      U[i] = UL[2]
      L[i] = UL[1] }
   return(c(L,U))
}
```

The above functions can be used to calculate and draw graphs showing the actual rate of occurrence, the estimated values and the corresponding confidence intervals. Below we have a sample R code and graph. This verifies that the software works as intended, the estimates of the Poisson rate and the confidence interval follow the rate that was used to generate them quite closely.

```
a <- 0
b <- 24
lud1 = GeneratePoissonSample(a, b)
t = seq(a, b, length.out=100)
lambda = 50 * (sin(2*pi*(t-a)/(b-a)) + 1)

#pdf("m5871100.pdf", width=4.5, height=4.5)
par(mai=c(0.8,0.8,0.1,0.1))

# n= Number of intervals
DoOnePlot = function(n) {
   lud2 = IntervalCount(lud1, n, a, b)
   lud3 = ConfidenceIntervals2(lud2)

   #scale
   est = lud2 * n/(b-a)
   LU = lud3 * n/(b-a)

   plot(t, lambda, xlim=c(a,b), ylim=c(0,max(LU)), type="l",
    xlab="interval", ylab="estimate and confidence limits")

   for (i in 1:n) {
     x0 = a + (b-a) * (i-1) / n
      x1 = a + (b-a) * i / n
      rect(x0, LU[i], x1, LU[i+n], col="gray", border=NA)
      segments(x0=x0, y0=est[i], x1=x1,y1=est[i])
      segments(x0=x0, y0=LU[i], x1=x1, y1=LU[i])
```

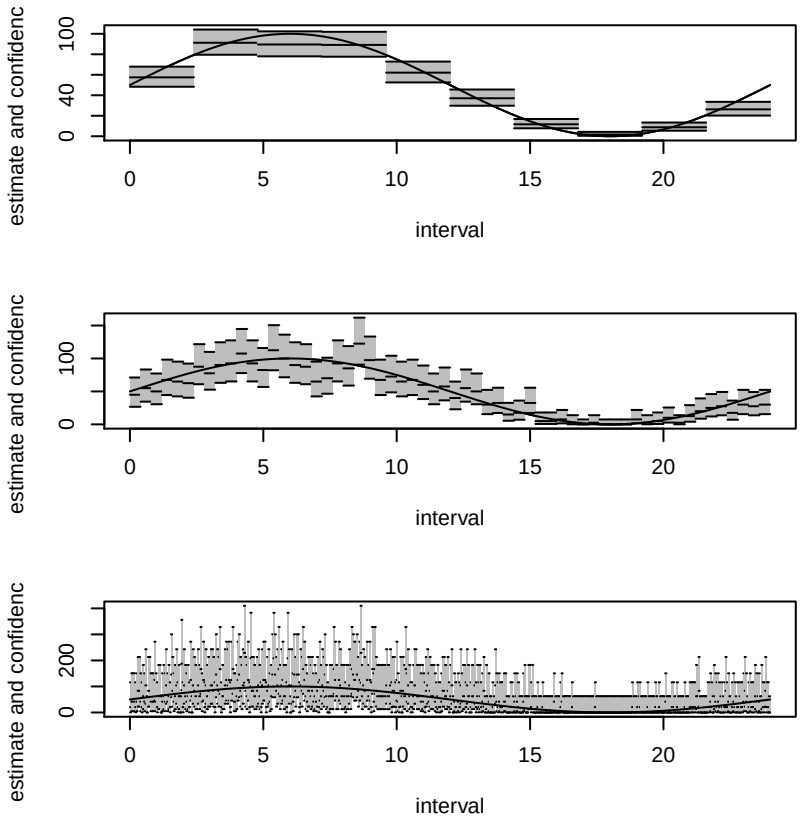

Figure 2.6: Confidence intervals as the poisson mean varies as a Sine wave.

```
      segments(x0=x0, y0=LU[i+n], x1=x1, y1=LU[i+n])
   }
   lines(t, lambda)
}

par(mfrow=c(3,1))
DoOnePlot(10)
DoOnePlot(60)
DoOnePlot(500)

#dev.off()
```

This example of the intensity λ varying as a sine wave is an example of a non-homogeneous Poisson process which will be considered on more detail in later chapters.

Chapter 3

Point Processes

This is a chapter on general point process and results specific to the Poisson process. We start with a discussion on duality between the waiting time until a fixed number of events occur and the number of events that happen in a fixed length of time. Then the memoryless property of the exponential distribution is proved, this is useful to tie up loose ends when considering the start of the Poisson process.

3.1 Duality

Suppose we have a process in time and we are interested in how many events have occurred after a fixed length of time. And the distribution of how long it takes for a fixed number of events to occur. As stated previously, let $N(t)$ be the distribution of the number of events that occur in an interval $(0, t]$. And W_n be the distribution of the waiting time until the nth event.
There is a relationship between $N(t)$ and W_n

$$P(W_n > t) = P(N(t) < n) \text{ and equivalently } P(W_n \leq t) = P(N(t) \geq n) \tag{3.1}$$

This can be explained as follows, if $W_n > t$ then we are still waiting for the nth event to occur when time is t. The nth event occurs after time t. So the number of events that have occurred up to time t must be less than n. And vice versa.

3.2 Memoryless property

The exponential distribution is the only continous distribution possesing the memoryless property. This is the property that the distribution of waiting time to an event from some arbitary point, is independent of how long it has being since the previous event happened. This is best explained by an example.

Suppose inter-arrival times follow an exponential distribution with rate λ. Let T be a random variable of actual inter-arrival times. Let as start observing the system at time $T = c$. Let X be the random variable measuring time to the first arrival, from the time we start observing the system.

Suppose $T \sim M(\lambda)$ We want the distribution of X.

$$\begin{aligned} P(X > x) &= P(T > x + c | T > c) \\ &= \frac{P([T > x + c] \cap [T > c])}{P(T > c)} \\ &= \frac{P(T > x + c)}{P(T > c)} \\ &= \frac{e^{-\lambda(x+c)}}{e^{-\lambda c}} \\ &= e^{-\lambda x}. \end{aligned} \tag{3.2}$$

Thus $X \sim M(\lambda)$. This means that the fact observing started at time $T = c$ has been forgotten and is not relevant.

3.3 Moment generating functions

Definition 2. *The moment generating function (mgf), $\phi_X(s)$ of a random variable X is defined as $\phi_X(s) = E(e^{sX})$.*

Example 1. *The mgf of a Poisson distribution $X \sim pois(\lambda)$ is:*

$$\begin{aligned} \phi(s) &= E(\mathrm{e}^{sX}) \\ &= \sum_{k=0}^{\infty} \mathrm{e}^{sk} P(X = k) \\ &= \sum_{k=0}^{\infty} \frac{\mathrm{e}^{sk} \mathrm{e}^{-\lambda} \lambda^k}{k!} \\ &= \mathrm{e}^{-\lambda} \sum_{k=0}^{\infty} \frac{(\lambda \mathrm{e}^s)^k}{k!} \\ &= \exp\big(\lambda(\mathrm{e}^s - 1)\big). \end{aligned} \tag{3.3}$$

Mgf's are important because each distribution has a unique mgf. Thus from the mgf we can identify the distribution (Francis, 1979, p. 314). And the mgf of a sum of independent random variables is the product of the mgf's of the constituent random variables.

Proof. Let $X_1, X_2, X_3 ....X_n$ be independent random variables with Mgfs $\phi_1(s), \phi_2(s), \phi_3(s) .....\phi_n(s)$.

Let $Y = X_1 + X_2 + X_3 + ... + X_n$.

$$\begin{aligned}\phi_Y(s) &= E(\exp(sY)) \\ &= E(\exp(sX_1 + sX_2 + sX_3 + ... + sX_n)) \\ &= E(\mathrm{e}^{sX_1}\mathrm{e}^{sX_2}\mathrm{e}^{sX_3}....\mathrm{e}^{sX_n}) \text{ by independence.} \\ &= E(\mathrm{e}^{sX_1})E(\mathrm{e}^{sX_2})E(\mathrm{e}^{sX_3})....E(\mathrm{e}^{sX_n}) \\ &= \phi_1(s)\phi_2(s)\phi_3(s)...\phi_n(s).\end{aligned} \tag{3.4}$$

If in addition $X_1, X_2....X_n$ are identically distributed then $\phi_Y(s) = (\phi_1(s))^n$. □

lemma 1. *Distribution of waiting time to nth event. Suppose the inter-arrival time between successive events are independent and follow an exponential distribution. The waiting time to the n_{th} event follows a Gamma distribution.*

Three methods showing this result are below. Each method starts with different assumptions.

Proof. Using mgfs.

The mgf of $M(\lambda)$ is $\frac{\lambda}{(\lambda - s)}$.

The mgf of $\Gamma(n, \lambda)$ is $(\frac{\lambda}{\lambda - s})^n$.

This shows that since the mgf of the Gamma is the mgf of the exponential raised to the power n. The Gamma distribution is the sum of n i.i.d exponential variates using equation 3.4. □

Proof. From Duality.

Using equation (3.1) and the assumption that the number of events in a fixed time follows a Poisson distribution.

$$\begin{aligned}P(W_n \leq t) &= P(N(t) \geq n) \\ &= \sum_{i=n}^{\infty} \mathrm{e}^{-\lambda t}\frac{(\lambda t)^i}{i!} \text{ Differentiating with respect to t gives the density function of } W_n. \\ f(t) &= -\sum_{i=n}^{\infty} \lambda \mathrm{e}^{-\lambda t}\frac{(\lambda t)^i}{i!} + \sum_{i=n}^{\infty} \lambda \mathrm{e}^{-\lambda t}\frac{(\lambda t)^{i-1}}{(i-1)!} \\ &= \lambda \mathrm{e}^{-\lambda t}\frac{(\lambda t)^{n-1}}{(n-1)!}. \\ &= \frac{t^{n-1}\lambda^n e^{-\lambda t}}{(n-1)!}\end{aligned} \tag{3.5}$$

This is the density of the Gamma distribution. □

Proof. Using independence in disjoint time periods.

This uses again the assumption that in a fixed time, number of events follow a Poisson distribution. We also assume that events happen independently in disjoint time periods.

$$\begin{aligned} P(t < W_n < t + dt) &= P(N(t) = n - 1,\ 1 \text{ event occurs in } (t, t + dt)) \\ &= P(N(t) = n - 1) P(\ 1 \text{ event occurs in } (t, t + dt)) \\ &= \mathrm{e}^{-\lambda t} \frac{(\lambda t)^{n-1}}{(n-1)!} \lambda dt \text{ dividing by dt.} \\ f(t) &= \mathrm{e}^{-\lambda t} \frac{(\lambda t)^{n-1}}{(n-1)!} \lambda \\ &= \frac{t^{n-1} \lambda^n e^{-\lambda t}}{(n-1)!} \end{aligned} \tag{3.6}$$

This again is the density of the Gamma distribution. □

Chapter 4

Homogeneous Poisson Process

The homogeneous Poisson process is a fundamental stochastic process which is simple, easy to understand and possesses desirable mathematical and theoretical properties making it easy to handle. It can be easily extended to more complicated and realistic situations Kingman (1993).

This chapter contains four definitions of the homogeneous Poisson process and proves that they are equivalent.

4.1 Equivalent definitions

The Poisson process was developed over time by different people with differing opinions on how to define it. One definition can lead to a conclusion which someone else decides to include as a definition. Some definitions are shorter than others, some are more more plausible to beginners. Other definitions can be abstracted and used in a more general context.

In this section we will discuss four definitions of the homogeneous Poisson process and show that they are all equivalent to each other. This discussion will be in one dimension, usually time. This will be extended to higher dimensions later.

Definition 3 (Statement A)**.** $N : [0, \infty) \to \mathbb{N}$ *is a homogeneous Poisson process with intensity* $\lambda : [0, \infty) \to [0, \infty)$ *if:*

(i) *N(0) = 0*

(ii) *For any times* $0 \leq t_0 < t_1 < t_2 < t_3 ..... < t_{n-1} < t_n < \infty$ *The random variables* $N(t_k) - N(t_{k-1}), k = 1, 2, 3...n$ *are mutually independent.*

(iii) *For any pair of times* $0 \leq t_1 < t_2$*, the random variable* $N(t_2) - N(t_1)$ *is Poisson distributed with parameter* $\lambda(t_2 - t_1)$*. This implies that the process is stationary.*

Definition 4 (Statement B)**.** $N : [0, \infty) \to \mathbb{N}$ *is a homogeneous Poisson process with intensity* $\lambda : [0, \infty) \to [0, \infty)$ *if:*

(i) N(0) = 0

(ii) The process has stationary and independent increments.

(iii) For 'small' $h, P(N(h) = 1) = \lambda h + o(h)$

(iv) $P(N(h) \geq 2) = o(h)$.

Definition 5 (Statement C)**.** $N : [0, \infty) \to \mathbb{N}$ *is a homogeneous Poisson process with intensity* $\lambda : [0, \infty) \to [0, \infty)$ *if: The waiting time between n events,* W_n *follows a Gamma* $\Gamma(n, \lambda)$ *distribution.*

Definition 6 (Statement D)**.** $N : [0, \infty) \to \mathbb{N}$ *is a homogeneous Poisson process with intensity* $\lambda : [0, \infty) \to [0, \infty)$ *if: The waiting time between successive events, or arrivals follows an exponential* $M(\lambda)$ *distribution.*

4.2 Equivalence of the definitions

To show all four definitions are equivalent I will show that. $B \Leftrightarrow A \Leftrightarrow C \Leftrightarrow D$

$B \Rightarrow A$. This is easy. Many standard books such as (Ross, 1996, p. 60-63) contain a derivation. Using definition B, a series of differential equations is set up for $P(N(t) = n), n = 0, 1, 2....$ This system is solved for $n = 0, 1$ And mathematical induction is used to prove that $P(N(t) = n)$ is equal to a Poisson distribution with parameter λt.

$A \Rightarrow B$. Parts (i) and (ii) agree. The number of events follows a Poisson distribution;

$$P(N(t) = n) = \frac{e^{-\lambda t}(\lambda t)^n}{n!}, n = 0, 1...$$

When $n = 1$ this becomes;

$$P(N(t) = 1) = e^{-\lambda t}(\lambda t)$$

Using the power series expansion for the exponential function this becomes;

$$\begin{aligned} P(N(t) = 1) &= (1 - \lambda t + \frac{(\lambda t)^2}{2!} - \frac{(\lambda t)^3}{3!} + o(\lambda t))(\lambda t) \\ &= \lambda t + o(\lambda t) \\ P(N(t) \geq 2) &= o(\lambda t) \end{aligned} \tag{4.1}$$

If t is small and $\lambda t = h$. The result follows.

$A \Rightarrow C$. We assume that the number of events in a fixed length of time follows a Poisson distribution and want to show that the waiting time distribution to the nth event follows a Gamma distribution.

Using the duality equality $W_n > t \Leftrightarrow N(t) < n$
Let F be the cdf of W_n and f the pdf of W_n.

$$\begin{aligned}
P(W_n > t) &= P(N(t) < n) \\
1 - F &= \sum_{k=0}^{n-1} P(N(t) = k) \\
-f &= \frac{d}{dt}[\sum_{k=0}^{n-1} P(N(t) = k)] \\
&= \frac{d}{dt}([P(N(t) = 0) + \sum_{k=1}^{n-1} P(N(t) = k)] \\
&= \frac{d}{dt}[e^{-\lambda t} + \sum_{k=1}^{n-1} \frac{e^{-\lambda t}\lambda^k t^k}{k!}] \\
&= -\lambda e^{-\lambda t} + \sum_{k=1}^{n-1} \frac{e^{-\lambda t}\lambda^k t^{k-1}}{(k-1!)} - \sum_{k=1}^{n-1} \frac{e^{-\lambda t}\lambda^{k+1} t^k}{k!} \\
&= -\lambda e^{-\lambda t} + \lambda e^{-\lambda t} + \sum_{k=2}^{n-1} \frac{e^{-\lambda t}\lambda^k t^{k-1}}{(k-1!)} - \sum_{k=2}^{n} \frac{e^{-\lambda t}\lambda^k t^{k-1}}{(k-1)!} \\
f &= \frac{e^{-\lambda t}\lambda^n t^{n-1}}{(n-1)!}
\end{aligned} \tag{4.2}$$

Thus the waiting time pdf to the nth event is the $\Gamma(n, \lambda)$ distribution.

$C \Rightarrow A$
Here we assume that the waiting time to the nth event follows the Gamma distribution and want to show that the number of events in a fixed length of time follows the Poisson distribution. This result from calculus will be needed very soon.

$$\begin{aligned}
\frac{d}{dx}((\lambda x)^n e^{-\lambda x}) &= \lambda[n(\lambda x)^{n-1} e^{-\lambda x} - (\lambda x)^n e^{-\lambda x}] \\
\int \lambda[n(\lambda x)^{n-1} e^{-\lambda x} - (\lambda x)^n e^{-\lambda x}]dx &= (\lambda x)^n e^{-\lambda x}
\end{aligned} \tag{4.3}$$

Using the duality equality $W_n \leq t \Leftrightarrow N(t) \geq n$

$$
\begin{aligned}
P(N(t)=n) &= P(N(t)\geq n) - P(N(t)\geq (n+1)) \\
&= P(W_n \leq t) - P(W_{n+1} \leq t) \\
&= \int_0^t \frac{\lambda(\lambda x)^{n-1}e^{-\lambda x}}{(n-1)!} - \frac{\lambda(\lambda x)^n e^{-\lambda x}}{n!}dx \\
&= \frac{1}{n!}\int_0^t \lambda[n(\lambda x)^{n-1}e^{-\lambda x} - (\lambda x)^n e^{-\lambda x}]dx \qquad (4.4)
\end{aligned}
$$

using 4.3

$$
\begin{aligned}
&= [\frac{(\lambda x)^n e^{-\lambda x}}{n!}]_0^t \\
&= \frac{(\lambda t)^n e^{-\lambda t}}{n!}
\end{aligned}
$$

This is the Poisson distribution for number of events with rate λt.

$C \Rightarrow D$

The Gamma pdf is $f(t) = \frac{t^{n-1}\lambda^n e^{-\lambda t}}{(n-1)!}, t \geq 0$

Letting n=1 we get the exponential pdf $f(t) = \lambda e^{-\lambda t}, t \geq 0$

$D \Rightarrow C$
If the exponential is the waiting time between successive events then the Gamma is the waiting time to the nth event using the previous section with the proof using mgf's.

In addition we have a direct $A \Rightarrow D$
If no events occur in time t. Then we are stilll waiting for the first event.
$P(W_1 > t) = P(N(t) < 1) = P(N(t) = 0) = e^{-\lambda t}$
Exponential waiting time to first event follows!

4.3 Length of observation interval

Our objective is to find a confidence interval for λ the intensity of a Poisson process, when λ is the intensity then the number of events occurring per unit time follows a Poisson distribution with parameter λ. If however, we are observing occurrences of some variable of interest over a time interval $[a, b]$, then the length of the interval is $(b - a)$ units of time. N, the number of observations follows a Poisson distribution with parameter $\lambda(b - a)$.

$$N \sim \text{Pois}(\lambda(b - a)).$$

$[L(n), U(n)]$ from definition 1 above will give the confidence interval for $\lambda(b - a)$ where n is the actual observed value of N. To get a confidence interval for λ we need to divide the previous

confidence interval for $\lambda(b-a)$ by $(b-a)$.

$$[\frac{L(n)}{(b-a)}, \frac{U(n)}{b-a}]$$

Will give the confidence interval for λ.

Chapter 5

Non-Homogeneous Poisson Process

The non-homogeneous Poisson process is defined. The principle of Maximum Likelihood Estimation is used to find the formula which needs to be maximised in order to fit an intensity function. Four classes of possible intensity functions are introduced and we consider the various R options such as numerical integration and optimisation which are needed to fit an intensity function to the given data.

5.1 Motivation

The non-homogeneous Poisson process is a generalisation of the homogeneous Poisson process which was defined in several different ways which were then shown to equivalent. And in a similar way it can be defined by different but equivalent definitions. The definition below is easy to understand, it's consequence is easier to generalise and thus becomes part of the definition in a more general setting. In one dimension, such as time we can have.

Definition 7. *This is taken from (Ross, 2000, p.284-6) A counting process* $\{N(t), t \geq 0\}$ *is a Non-homogeneous Poisson process with intensity function* $\lambda(t), t \geq 0$*, if*

1. *N(0)=0.*
2. $\{N(t), t \geq 0\}$ *has independent increments.*
3. $P(N(t+h) - N(t) \geq 2) = o(h)$
4. $P(N(t+h) - N(t) = 1) = \lambda(t)h + o(h)$

A consequence of the above is that $N(t) \sim \text{Pois}(\Lambda(t))$, where $\Lambda(t) = \int_0^t \lambda(x)dx$.

This can now be turned around and the consequence is used in the definition of a general Poisson process.

Definition 8. *A Poisson process on a set $D \subseteq \mathbb{R}^d$ with intensity function $\lambda : \mathbb{R}^d \to [0, \infty)$ is a random set $\Pi \subseteq D$ if:*

1. *If $A \subseteq D$ then $|\Pi \cap A| \sim \text{Pois}(\Lambda(A))$, where $|\Pi \cap A|$ is the number of points of Π in A and*
$$\Lambda(A) = \int_A \lambda(x)dx.$$
2. *If $A, B \subseteq$ are disjoint, then $|\Pi \cap A|$ and $|\Pi \cap B|$ are independent.*

5.2 Maximum Likelihood Estimation of the intensity

Suppose we have data that comes from a non-homogeneous Poisson process and we want to find the intensity function that gave rise to it. One way of doing it, is to use the principle of Maximum Likelihood Estimate(MLE) to find the intensity function λ that maximises the likelihood of the data occurring.

Suppose we have a sample of size $n, \pi = \{x_1, x_2...x_n\}$ from a Poisson process as described above. Then the likelihood $L(\lambda; \pi = \{x_1, x_2...x_n\}) = f(\pi = \{x_1, x_2...x_n\}; \lambda)$ is the probability of getting the sample $\pi = \{x_1, x_2...x_n\}$,

Let λ the intensity function be, $\lambda \colon [0, T] \to \mathbb{R}_{\geq 0}$.

$$\begin{aligned} \Lambda &= \int_0^T \lambda(x)dx, \\ \text{Probability of observing n points } &= e^{-\Lambda}\frac{\Lambda^n}{n!}, \\ \text{Probability density of observation } x_i &= \frac{\lambda(x_i)}{\Lambda}, \end{aligned} \tag{5.1}$$

Since the observations are independent of each other, $P(\pi = \{x_1, x_2...x_n\}) = \prod_{i=1}^n \frac{\lambda(x_i)}{\Lambda}$. Likelihood of getting the sample $\pi = \{x_1, x_2...x_n\} = P(n) \cdot P(\pi = \{x_1, x_2...x_n\}|n)$.

$$L(\lambda; \pi = \{x_1, x_2...x_n\}) = e^{-\Lambda}\frac{\Lambda^n}{n!}\prod_{i=1}^n \frac{\lambda(x_i)}{\Lambda}.$$

If the sample is ordered $x_1 < x_2 < x_3 < .... < x_n$. Then the likelihood of the ordered sample is the likelihood of the above unordered sample above multiplied by $n!$ (Ross, 1996, p. 66-7).

$$\text{Likelihood of ordered sample } L(\lambda) = e^{-\Lambda}\frac{\Lambda^n}{n!} \cdot \prod_{i=1}^n \frac{\lambda(x_i)}{\Lambda} \cdot n!.$$

$$L(\lambda) = e^{-\Lambda} \cdot \prod_{i=1}^n \lambda(x_i) = e^{-\int_0^T \lambda(x)dx} \cdot \prod_{i=1}^n \lambda(x_i).$$

$$\text{The Log-Likelihood: } l(\lambda) = \log(L(\lambda)) = -\int_0^T \lambda(x)dx + \sum_{i=1}^{n} \log \lambda(x_i).$$

There are two practical points about finding the MLE. One, our intention is to find the function $\lambda(t)$ which maximises the Likelihood $L(\lambda)$ but in practice it is usually easier to evaluate the Log-Likelihood $l(\lambda)$. Since the logarithm is a one to one function the values which maximise L are the same as the ones that maximise l. L and $\log(L)$ are maximised at the same point. Secondly, our intention is to maximise L and l but most computer routines are written to minimise functions. To maximise L or l is equivalent to minimising $-L$ or $-\,l$.

5.3 Double checking the theory

The above theory developed MLE for the non-homogeneous Poisson process. To try to verify the formula that was found for the MLE I will consider the degenerate case of the homogeneous Poisson process. The MLE will be found in two different ways. From first principles and from the Log-Likelihood formula above. If the two answers do not agree then we have problems and the above formula and theory are probably wrong. If however the two answers agree we have extra evidence(Not proof) that the formula and theory are correct.

Method 1 (Direct). Consider a homogeneous Poisson process with rate λ, we observe the process for a time T and count n observations. The likelihood of this occurring $L = \exp(-\lambda T)\frac{(\lambda T)^n}{n!}$. The Log-Likelihood $l = -\lambda T + n\log(\lambda T) - \log(n!)$. Differentiating l w.r.t. λ gives $\hat{\lambda} = \frac{n}{T}$

Method 2 (Using Non-Linear formula for MLE). Suppose we have a polynomial of degree zero $\lambda(x) = a_0$

$$\int_0^T a_0 dx = a_0 T$$

$$\sum_{i=1}^{n} \log \lambda(x_i) = \sum_{i=1}^{n} \log(a_0) = n\log(a_0).$$

$$l = n\log(a_0) - a_0 T$$

Differentiating l w.r.t a_0 gives $\hat{a_0} = \frac{n}{T}$.

a_0 is the rate. Both methods give identical results. This increases confidence that the theory developed in the previous section is in fact correct.

5.4 Fitting an intensity function

Using the principle of Maximum likelihood we can try fitting an intensity function to some real data. Trying to find a 'best' fitting intensity function $\lambda(t)$ is not practical unless we restrict our search to specified families of functions.

The familes that will now be considered are;

1. Polynomials of the form $\lambda(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n$ with the restriction that $\lambda(x) \geq 0 \quad \forall x$.
2. Fourier(periodic) series $\lambda(x) = a_0 + \sum_{j=1}^{n} b_j \sin(2\pi \frac{x}{T} f_j) + c_j \cos(2\pi \frac{x}{T} f_j)$ with the restriction that $\lambda(x) \geq 0 \quad \forall x$.
3. Exponentials of polynomials $\lambda(x) = \exp(a_0 + a_1x + a_2x^2 + \cdots + a_nx^n)$.
4. Exponentials Fourier series $\lambda(x) = \exp\Big(a_0 + \sum_{j=1}^{n} b_j \sin(2\pi \frac{x}{T} f_j) + c_j \cos(2\pi \frac{x}{T} f_j)\Big)$.

5.4.1 Numerical Integration in R

Our next objective is to find the intensity function $\lambda(t)$ such that the Log-Likelihood function: $l(\lambda) = \log(L(\lambda)) = -\int_0^T \lambda(x)dx + \sum_{i=1}^{n} \log \lambda(x_i)$ is maximised. The log-likelihood function is composed of two components, an integral and a summation. Considering the four families of functions we are to investigate only the first can be integrated explicitly. The others require numerical integration. Our programming language of choice is 'R'. In R, numerical integration of a function of one variable is performed by the `integrate` function.

The `integrate` function was verified by examples for two reasons, firstly to check that it was being used correctly and to double check that it actually worked!

The two examples used to verify `integrate` were $\int_0^{20} \mathrm{e}^{2+3x} dx$ and $\int_0^{\infty} \mathrm{e}^{-\frac{x^2}{2}}$.

```
g <- function(y,b) {exp(b[1]+b[2]*y+b[3]*y^2)}

L6=integrate(g,0,Inf,c(0,0,-0.5))
L6$value

[1] 1.253314
```

The example code above integrates the positive half of the standard normal distribution. As expected $\sqrt{\frac{\pi}{2}}$ = 1.253314 approx.

5.4.2 Numerical Optimisation in R

In R there are several numerical optimisation functions. Some of them appear to duplicate the work of others. The reasons behind this are that numerical optimisation is a complex subject and there is no one best method. The methods vary in what input they require. Some only require the function to be minimised, others want the gradient supplied as well. Where the gradient

is not supplied it is usually approximated by finite difference methods. Some of the methods are more efficient than others but that depends on the function being optimised. Some methods work where others fail. Optimisation functions built into R are:

`optim.`

Described as a general-purpose optimization based on NelderMead, quasi-Newton and conjugate-gradient algorithms. It includes an option for box-constrained optimization and simulated annealing.

This includes a choice of the following methods: ("Nelder-Mead", "BFGS", "CG", "L-BFGS-B", "SANN", "Brent").

Another is `nlm` described as carrying out a minimization of the function f using a Newton-type algorithm.

Another one is `nlinb` described as unconstrained and constrained optimization using PORT routines. The PORT documentation is at http://netlib.bell-labs.com/cm/cs/cstr/153.pdf.

Chapter 6

Results

The main data which was analysed consists of times measured in seconds of hits on a website over a thirty day period. It is our belief that the data does not follow a homogeneous Poisson process and the rate may vary by time of day and there also may be a trend present. The intention is to fit a non-homogeneous Poisson process model to the data. The data which is actually anaylsed here is in three data files. One file consists of the full data. Another is a systematic one tenth sample of the full data and the third is yet another one tenth sample of the second sampled file. The reason for using sampled data files is to allow faster program execution times in the event where program execution time on the full file is so long that it becomes impractical to run. Taking a 10% and a 1% sample of the data should not affect greatly the models that are fitted since periods of high activity and periods of low activity are being scaled evenly and the relative intensities in different time periods should not be affected. Although we must keep in mind the danger that we may be missing some vital feature of the data by throwing away 90% and 99% of the available information.

The initial investigation of the data was a graphical analysis, this was exploratory in nature. ”John Tukey ” (Tukey, 1977, p. 3) stated: ‘Exploratory data analysis can never be the whole story, but nothing else can serve as the foundation stone–as the first step.’

The data covered a thirty day period and the full data file had 57,601 data values in seconds. The task was to display graphically this data in a meaningful way. The method chosen was to partition the data into equal length time periods and count the number of events in each partition. We assumed that in each partition the count came from a homogeneous Poisson process. Given that the count follows a sample from a Poisson distribution, the parameter of the Poisson distribution is estimated together with the lower and upper confidence limits. These three estimates are based on a count which depends on the length of the time period of the partition. These are then ‘scaled’ or converted to a Poisson rate per unit time, see section 4.3.

Two obvious ways of controlling the way the data is displayed are changing the number of partitions and the confidence coefficient. On the horizontal axes we have the equally spaced partitions of time. The vertical axes represents the Poisson rate per unit time. Each partition is shown as a grey rectangle. The middle horizontal line through the box is the estimate of the Poisson rate in that partition, the top and bottom of the box are the upper and lower confidence

limits of the estimate. The data covers a thirty day period so it is desirable that the number of partitions should be a multiple of thirty. Thirty partitions groups the data by day, sixty groups it into 12 hour periods, ninety groups into 8 hour periods etc. The confidence coefficient controls the size or length of the confidence interval In figures 6.1 to 6.8 we display the data with a 50% confidence coefficient as the number of intervals increases and the size of the interval decreases. A periodic pattern emerges.

With the data grouped by day there is no obvious pattern. In fact the intensity function looks almost uniform there is no evidence of a trend. As the grouping is refined into smaller time periods a pattern emerges. Most of the data is homogeneous near the bottom with a periodic pattern emerging of evenly spaced peaks near the top. There are in fact thirty, evenly spaced peaks corresponding to the thirty days that the data covers. The magnitude of these peaks looks constant with no obvious trend. From this graphical analyses I can see a steady state periodic daily pattern with no obvious linear trend over the thirty day period.

The original intention was to fit four classes of functions to the data.

Polynomials of the form $\lambda(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n$ with the restriction that $\lambda(x) \geq 0 \quad \forall x$. However looking at the graphs it is obvious that polynomials will not be a suitable fit. From 6.1 it is possible to conclude that there is no trend, especially if we ignore the high outlier on the right end. A trend-less fit of $\lambda(x) = a_0$ is just a homogeneous intensity, these polynomials cannot model the periodic nature of the data. This model will no longer be considered.

Exponentials of polynomials $\lambda(x) = \exp(a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n)$ were tried on small test data and gave results. However there is no way to model the periodic nature of the data. And the exponential often went out of bounds and failed to give a result. This model will no longer be considered.

Fourier(periodic) series $\lambda(x) = a_0 + \sum_{j=1}^{n} b_j \sin(2\pi \frac{x}{T} f_j) + c_j \cos(2\pi \frac{x}{T} f_j)$ are periodic in nature and were investigated. However we still have to consider the restriction $\lambda(x) \geq 0$). This restriction however is not modelled by the built in `R` routines and needs 3rd party `R` routines to implement. This class of models will be considered later if time permits.

Exponentials of Fourier series $\lambda(x) = \exp\Big(a_0 + \sum_{j=1}^{n} b_j \sin(2\pi \frac{x}{T} f_j) + c_j \cos(2\pi \frac{x}{T} f_j)\Big)$ will ensure that the intensity function will always be non-negative. This model is periodic and is considered here.

The parameters that will be estimated by the program are $\{a_0, b_j, c_j\} \quad j = 1 \cdots n$. The parameters which are supplied by the user are $\{T, x, n, f_j\} \quad j = 1 \cdots n$. In a more complex program it may be desirable to use the program to also estimate f_j, however, here I estimated f_j by sight from the graphs 6.7 and 6.8. It is clear that there is a periodic daily pattern. In the program, x is the data in seconds, T is the number of seconds in a day, so x/T is in days and f_j is the frequency per day. The frequencies used were $f_1 = 1$, $f_2 = 2f_1$, $f_3 = 3f_1$ etc.

The optimization program was verified by running it using two different optimization routines. `optim` and `nlm` on the same data, both gave nearly identical results. I chose to use `optim`

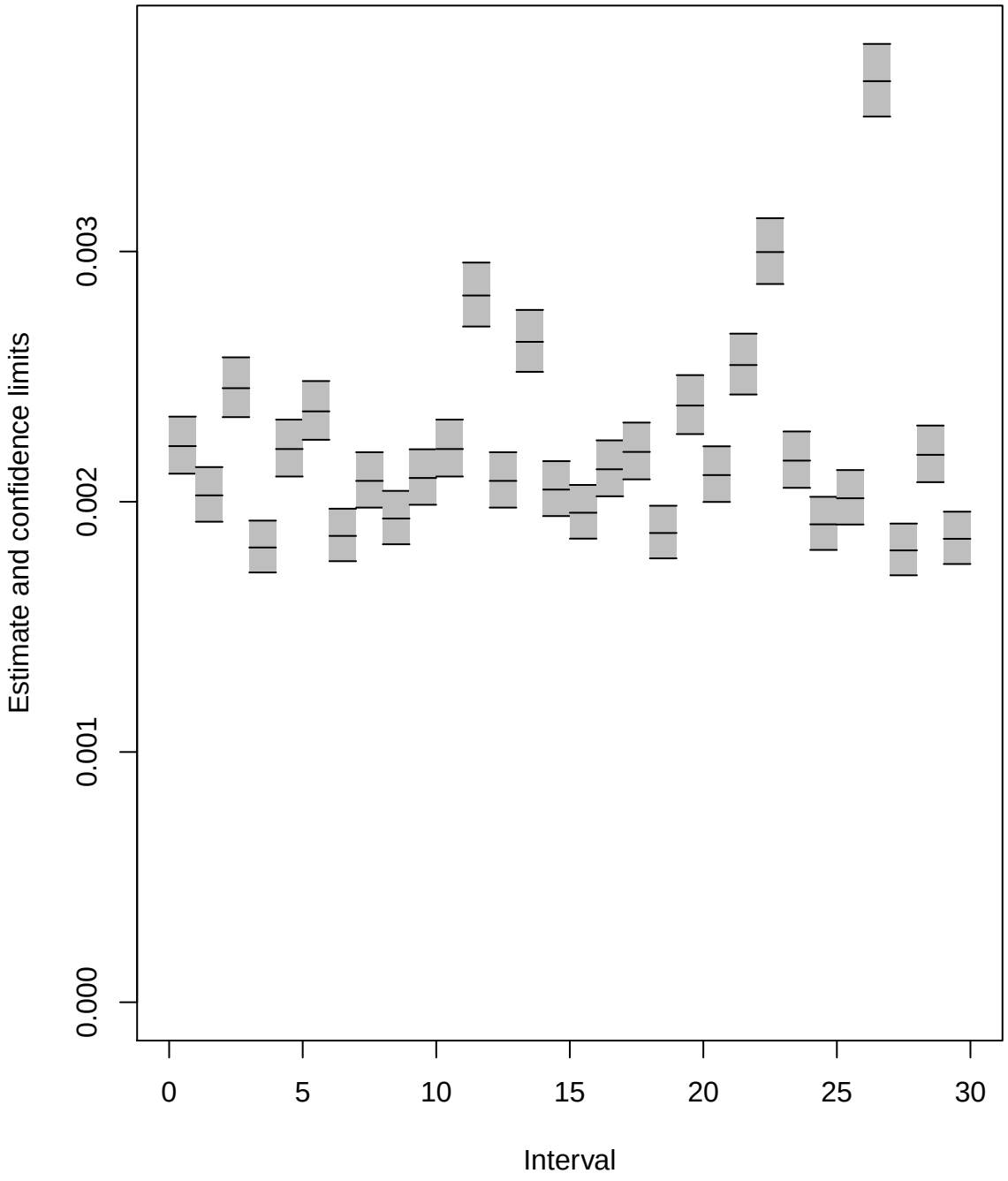

Figure 6.1: Confidence intervals with a daily grouping. No obvious linear trend or any other pattern. Could be a homogeneous Poisson process.

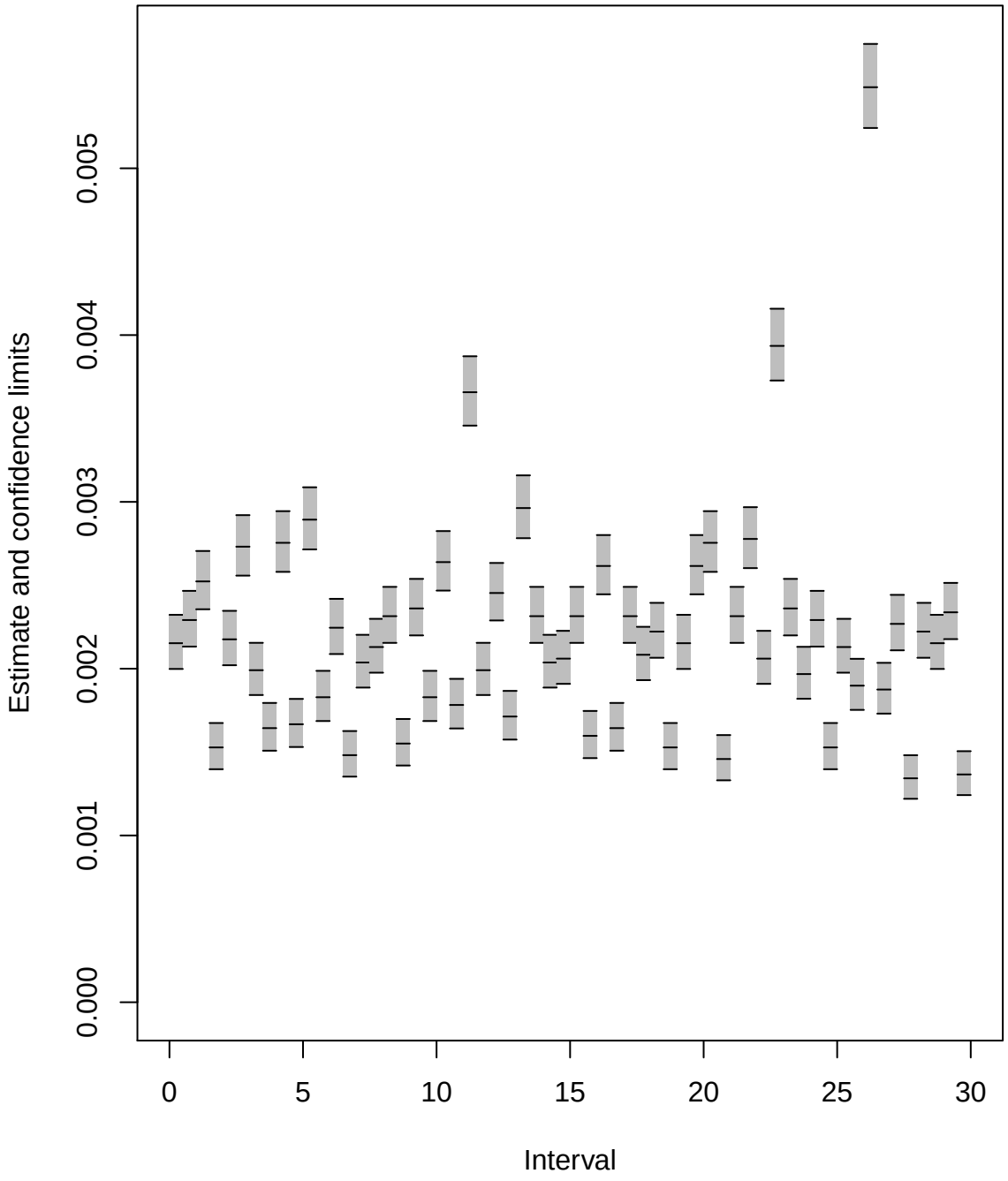

Figure 6.2: Confidence intervals with a 12 hour grouping. A lot more variability. A suggestion of periodicity at the lower end ?

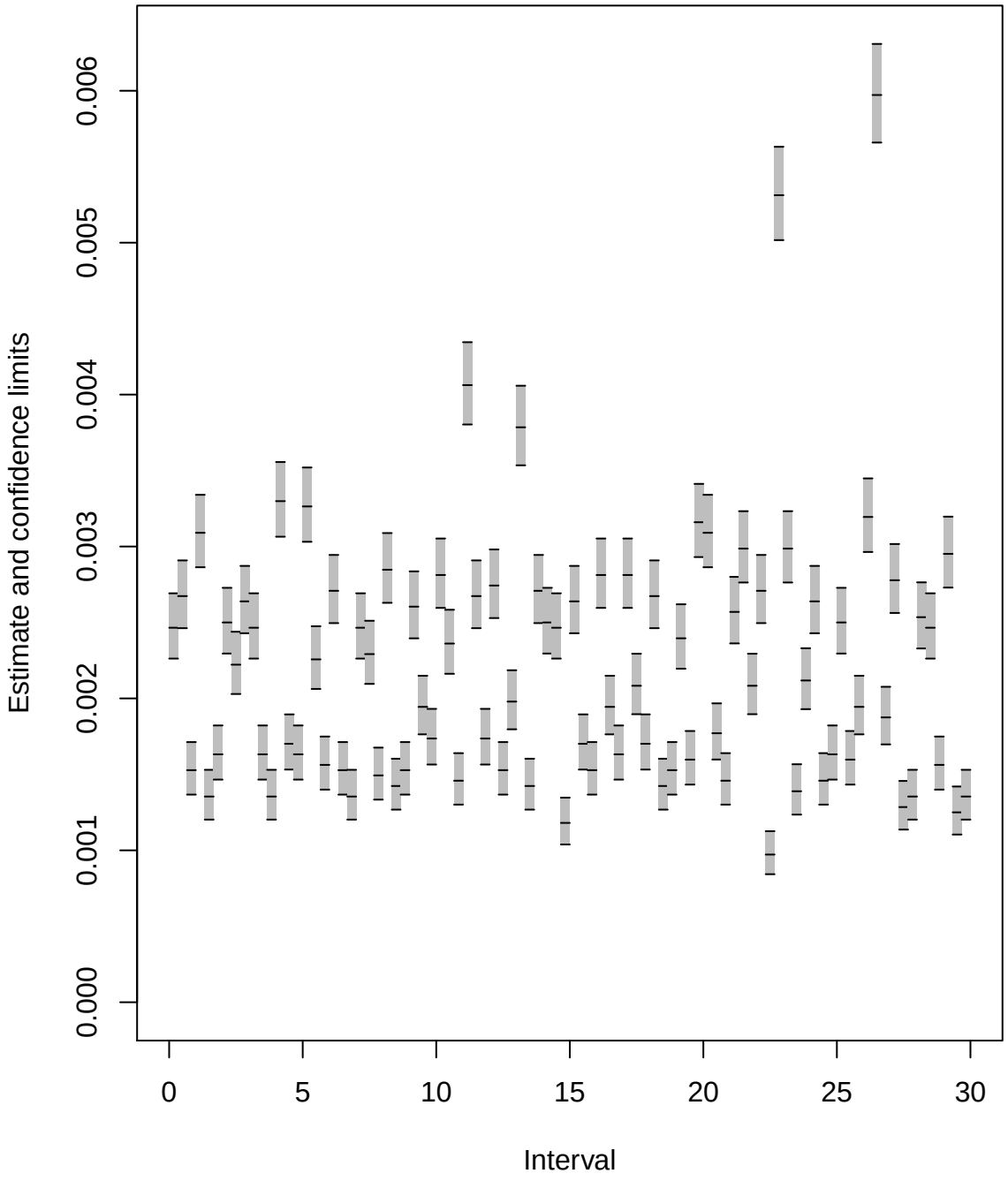

Figure 6.3: Confidence intervals with a 8 hour grouping. The data sees to be splitting itself into two large groups with outliers at the top.

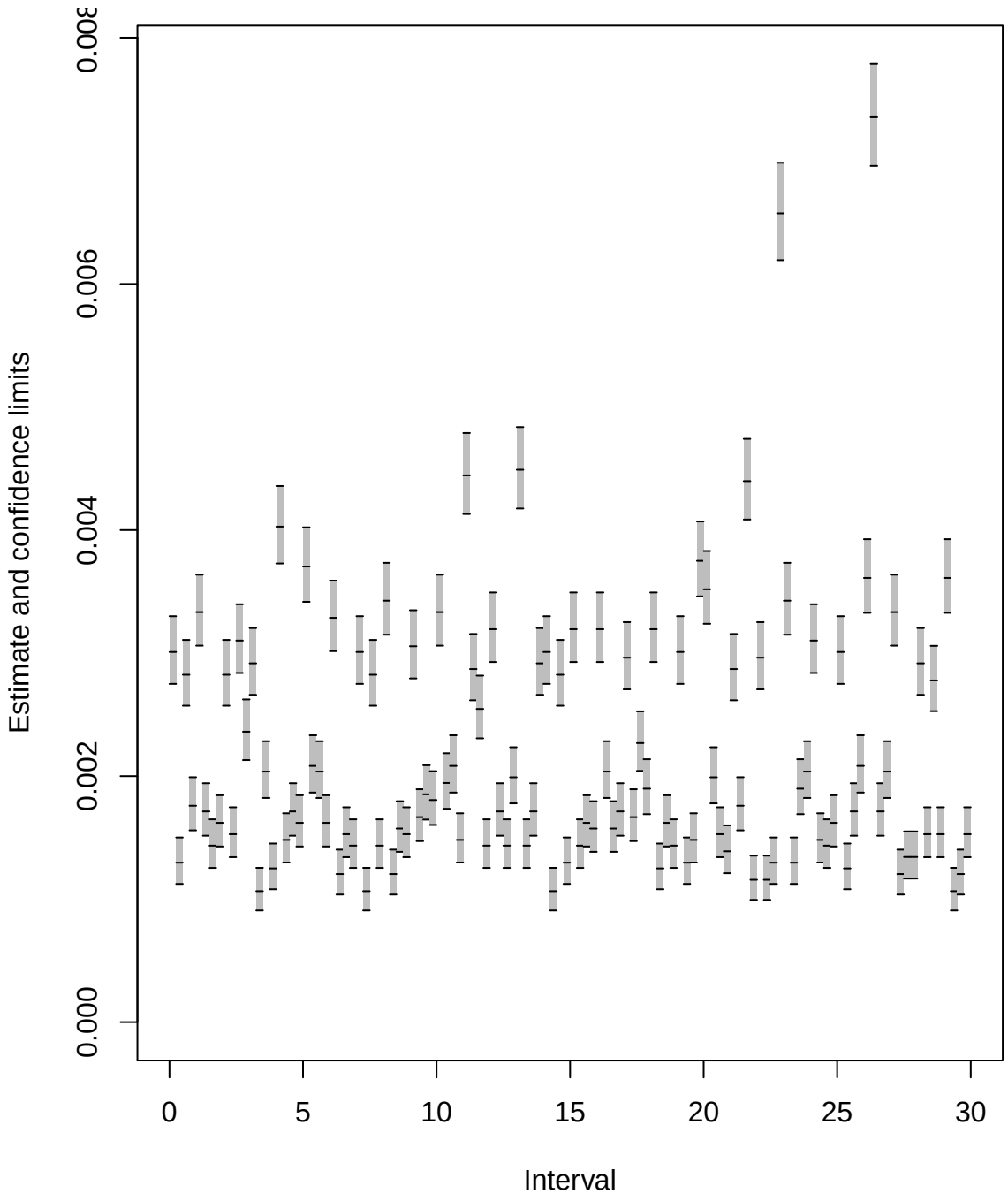

Figure 6.4: Confidence intervals with a 6 hour grouping. More clear evidence of data splitting into two groups.

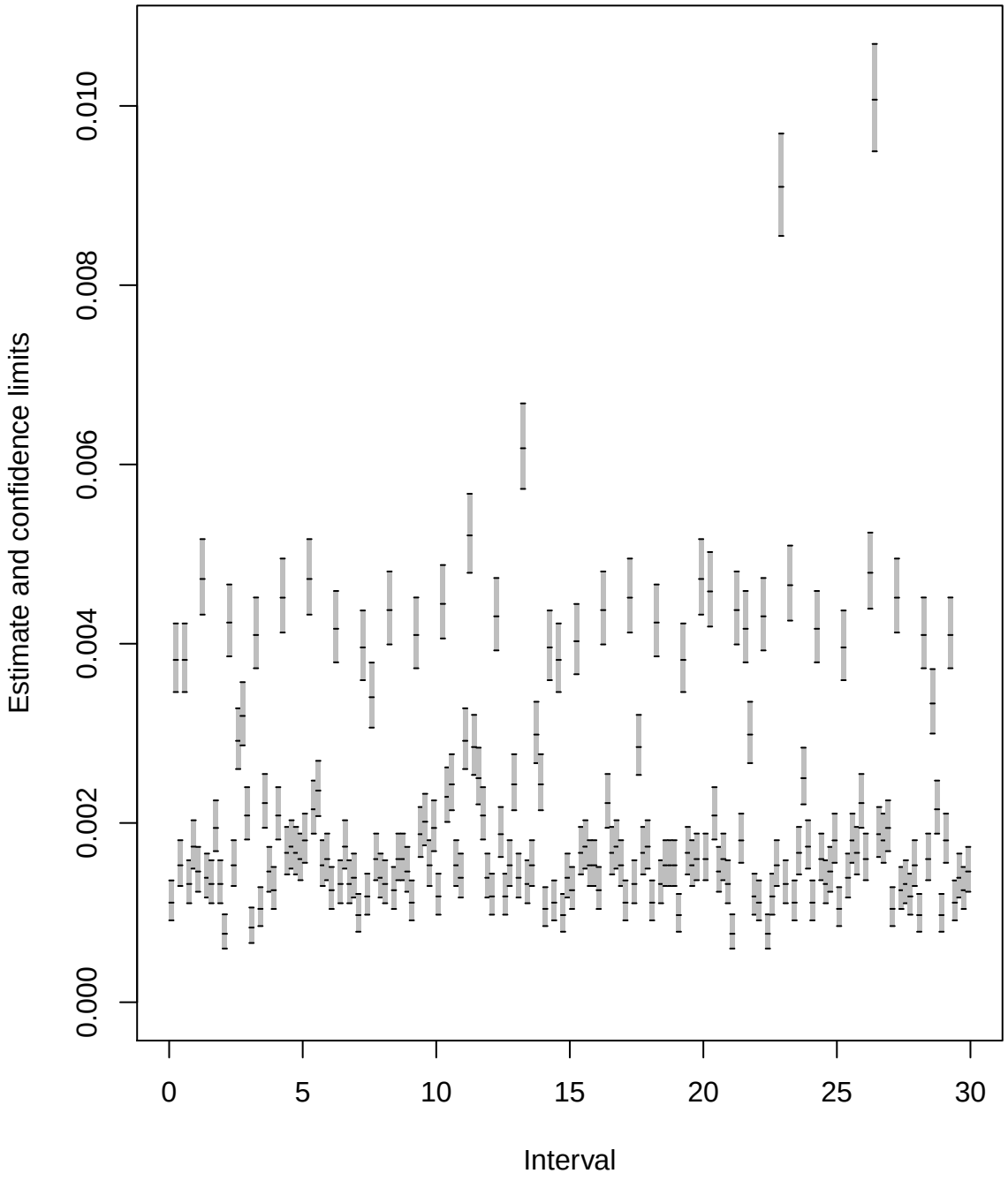

Figure 6.5: Confidence intervals with a 4 hour grouping. Even more evidence of two groups with a hint that the top group may be periodic.

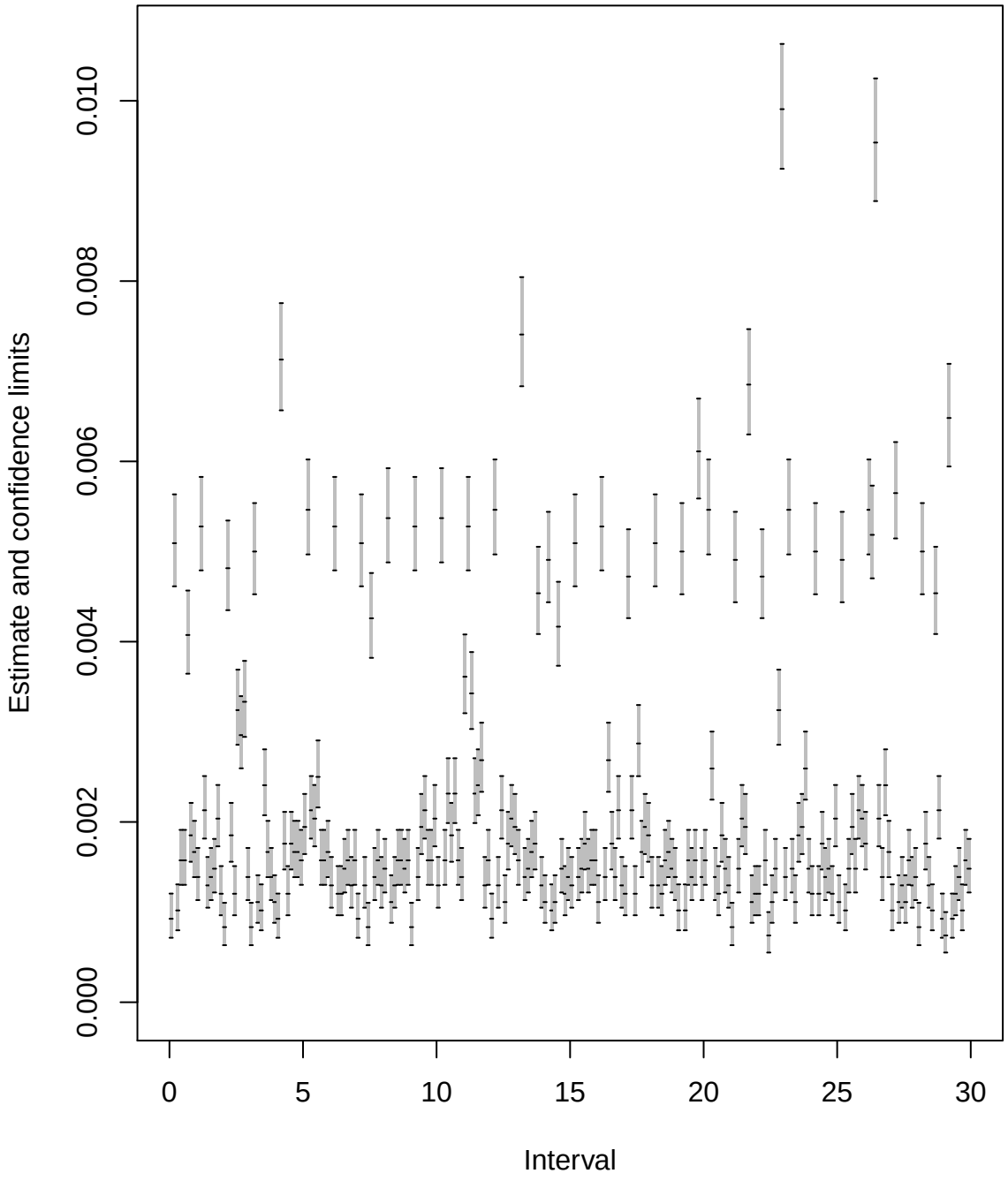

Figure 6.6: Confidence intervals with a 3 hour grouping. The upper group is looking more and more periodic.

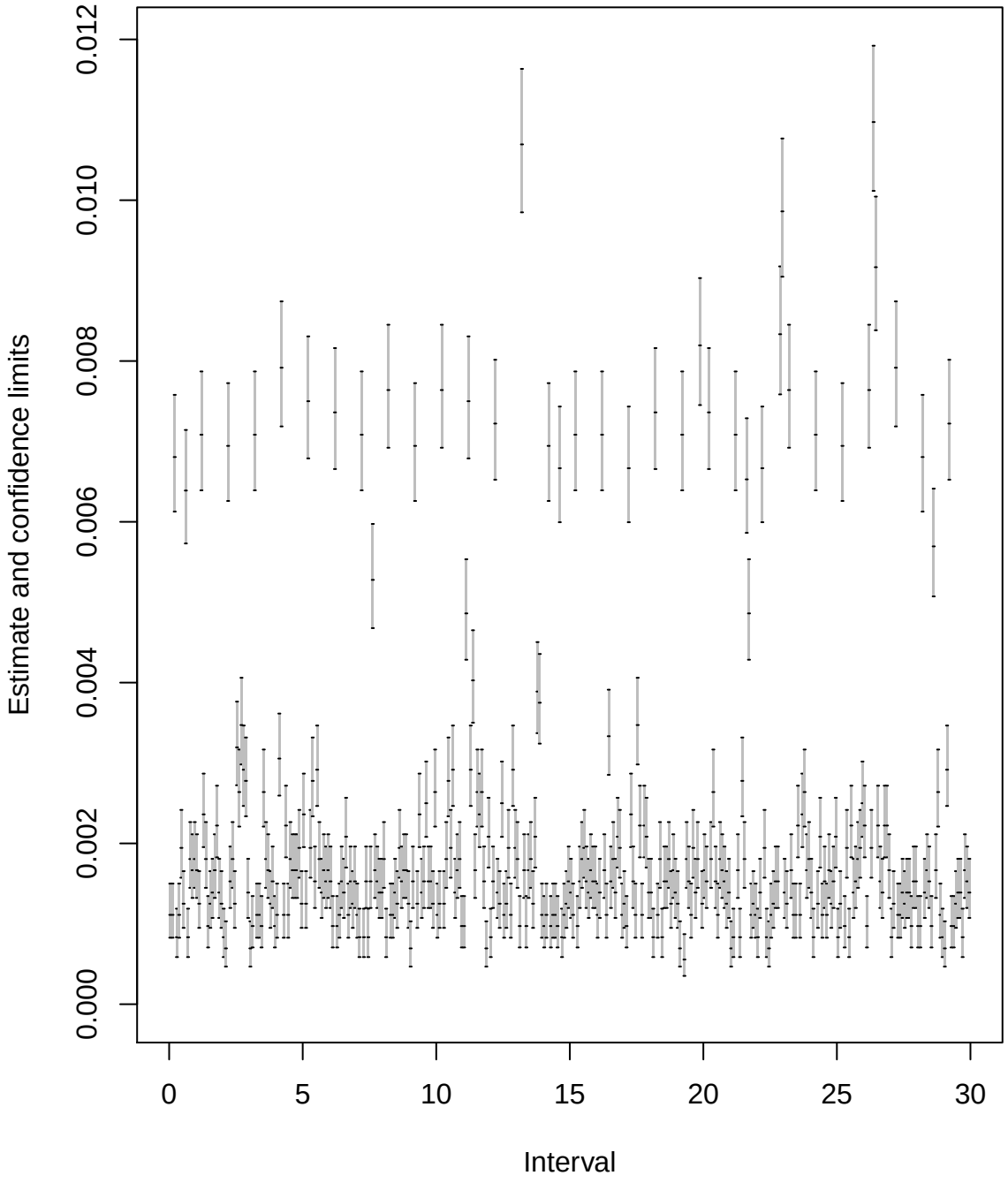

Figure 6.7: Confidence intervals with a 2 hour grouping. The two group periodic pattern is still there with some intermediate points.

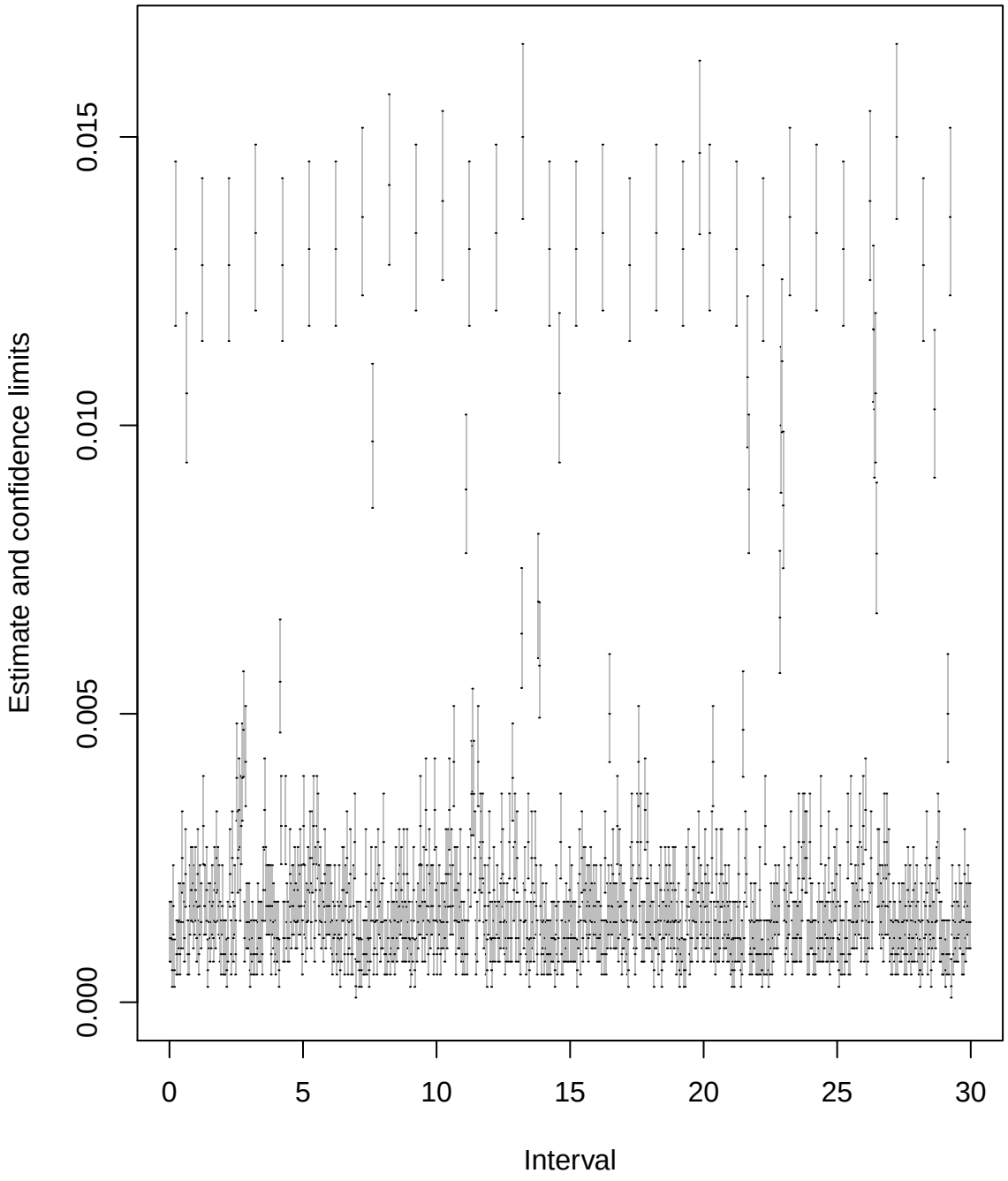

Figure 6.8: Confidence intervals with a 1 hour grouping. It is now clear that we have periodic peaks.

with the default optimization method since it ran on the full data set with no problems and only took a few seconds. If the program had failed or taken too long to converge I would have tried using different optimization methods and or used one of the smaller data sets.

I decided that initial verification of the model would be graphical in nature. The program would fit the exponential Fourier model to the full data set and the fitted intensity curve coloured red would be superimposed on the grey partitioned data rectangles with the corresponding confidence intervals. If the fitted curve corresponds closely with the data then it would be considered to be a good fit. The display of the grey data can be controlled by varying the number of partitions and the confidence coefficient. The red line of the fitted model is independent of the number of partitions. It is dependent on and is calculated from whole data file so it does not change as we vary the number of partitions. The fitted model can be controlled by changing n in $\lambda(x) = \exp\Big(a_0 + \sum_{j=1}^{n} b_j \sin(2\pi \frac{x}{T} f_j) + c_j \cos(2\pi \frac{x}{T} f_j)\Big)$. And $f_j = j f_1$.

If $n = 0$ then we are fitting a constant, equivalent to a homogeneous Poisson process with intensity function. $\lambda(x) = \exp(a_0) = \mu$. In figures 6.9 and 6.10 we see a horizontal red line representing the homogeneous rate. It passes through the data as an overall average. Although it is a good fit passing through the center of the data, it does not model the periodic nature of the data shown in 6.10.

The rate is the (number of observations in 30 days) / (Total time in seconds in 30 days) = 57601/2592000 = 0.022 and this corresponds to the height of the red line in 6.9 and 6.10.

The next step is to try to model periodicity by setting n to 1. This corresponds to the intensity function $\lambda(x) = \exp\Big(a_0 + b_1 \sin(2\pi \frac{x}{T} f_1) + c_1 \cos(2\pi \frac{x}{T} f_1)\Big)$. The graphs 6.11 to 6.14 show how the fitted function is superimposed on the data. As the number of partitions increased from 30, 60, 90 then 120. The apparent goodness of fit changed. The fitted red function stays the same. The gray summary rectangles change. The more partitions we have means that each rectangle contains less data points and there is more variability displayed. The high data peaks become higher and the low data troughs become lower. With 30 partitions each rectangle represents the intensity for the whole day. With 60 partitions, each rectangle represents the intensity for half a day. Graph 6.13 with a 8 hour grouping looks best.

The next step is to try to model periodicity by setting n to 2. This corresponds to the intensity function $\lambda(x) = \exp\Big(a_0 + b_1 \sin(2\pi \frac{x}{T} f_1) + c_1 \cos(2\pi \frac{x}{T} f_1) + b_2 \sin(2\pi \frac{x}{T} f_2) + c_2 \cos(2\pi \frac{x}{T} f_2)\Big)$. And $f_2 = 2f_1$. The fitted model now has two peaks and troughs per day fig 6.15 shows the fitted model superimposed on a 12 hour grouping. It is difficult to see the pattern due to the peaks being very cramped together. The rotated graph in fig 6.16 is much clearer. In fig 6.17 with 8 hour groups the main peaks and troughs match the data. The secondary peaks and troughs are not as clear. Although the troughs are a better fit than the peaks. One interesting feature is the two high outliers on days 23 and 27. These correspond to the secondary peaks. A further refinement to a 6 hour grouping in fig 6.18 shows The main peaks match best the secondary peaks correspond to three high outliers. The troughs fitting is average. A further refinement to 4 hour grouping in fig 6.19 shows an improvement in the fit. The main peaks match, secondary peaks can be identified on several days and the troughs match.

The next step is to try to model periodicity by setting n to 3. This corresponds to the intensity

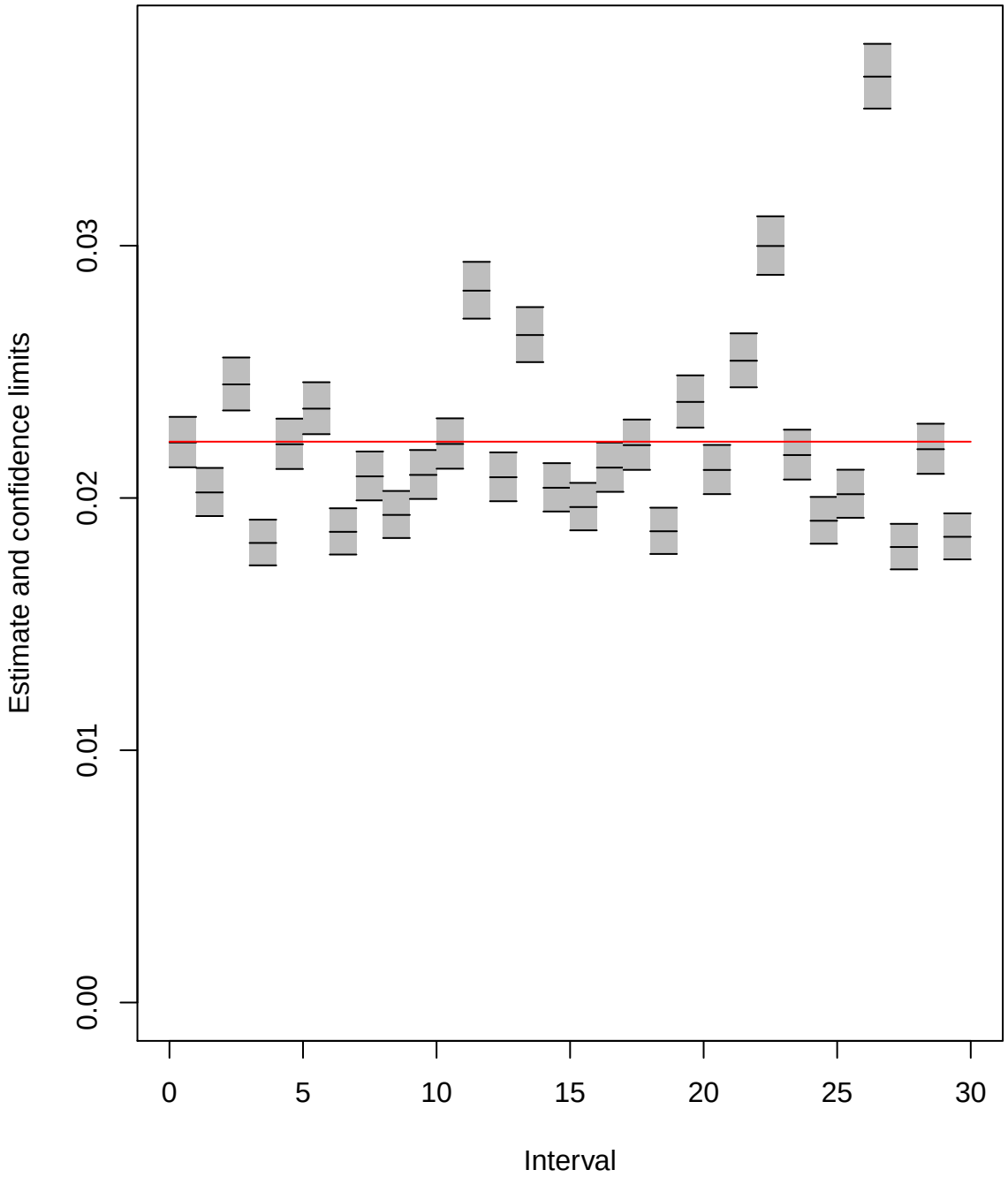

Figure 6.9: Confidence intervals with a 1 day grouping and a homogeneous rate fitted in red. n=0.

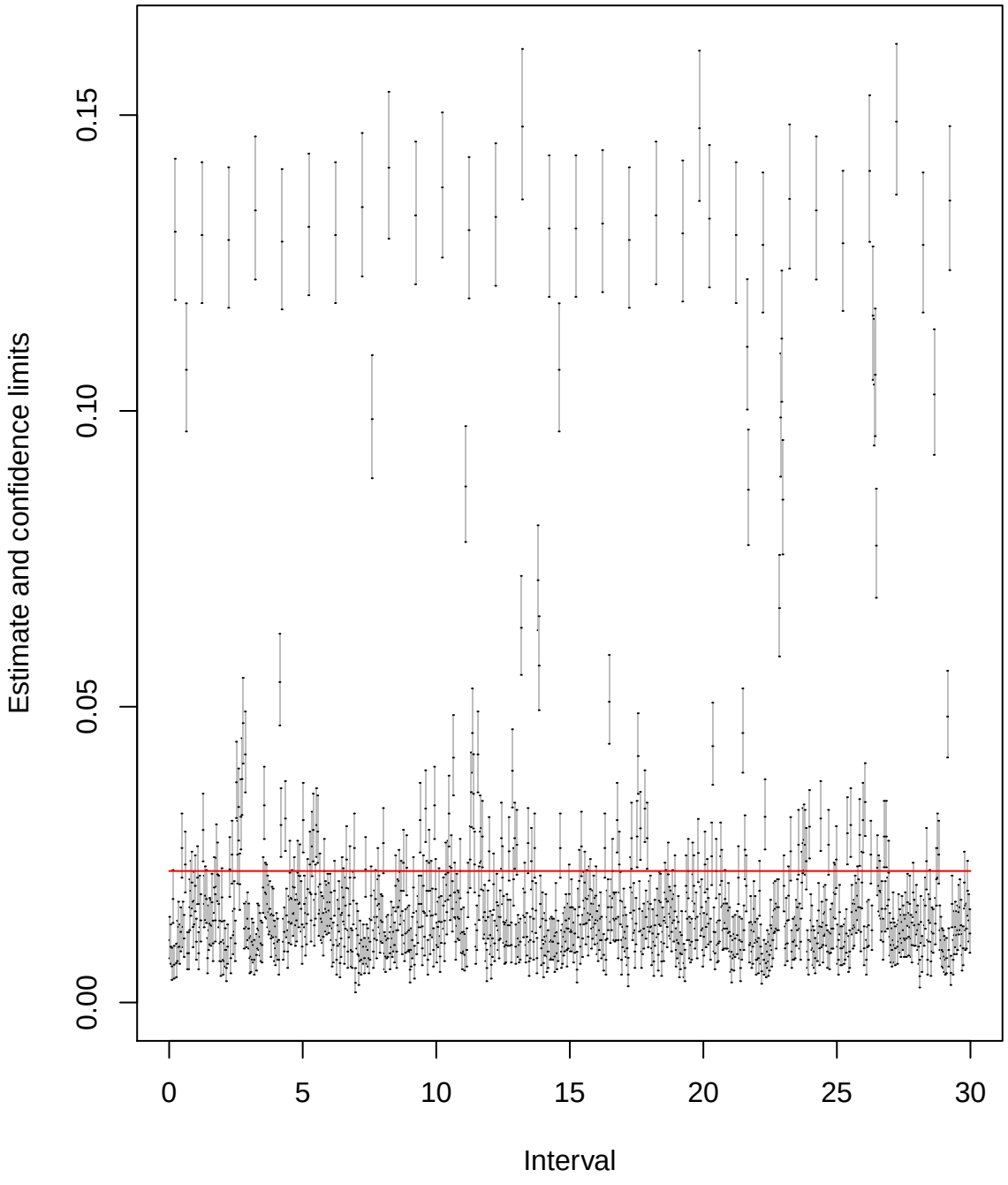

Figure 6.10: Confidence intervals with a 1 hour grouping and a homogeneous rate fitted in red. n=0.

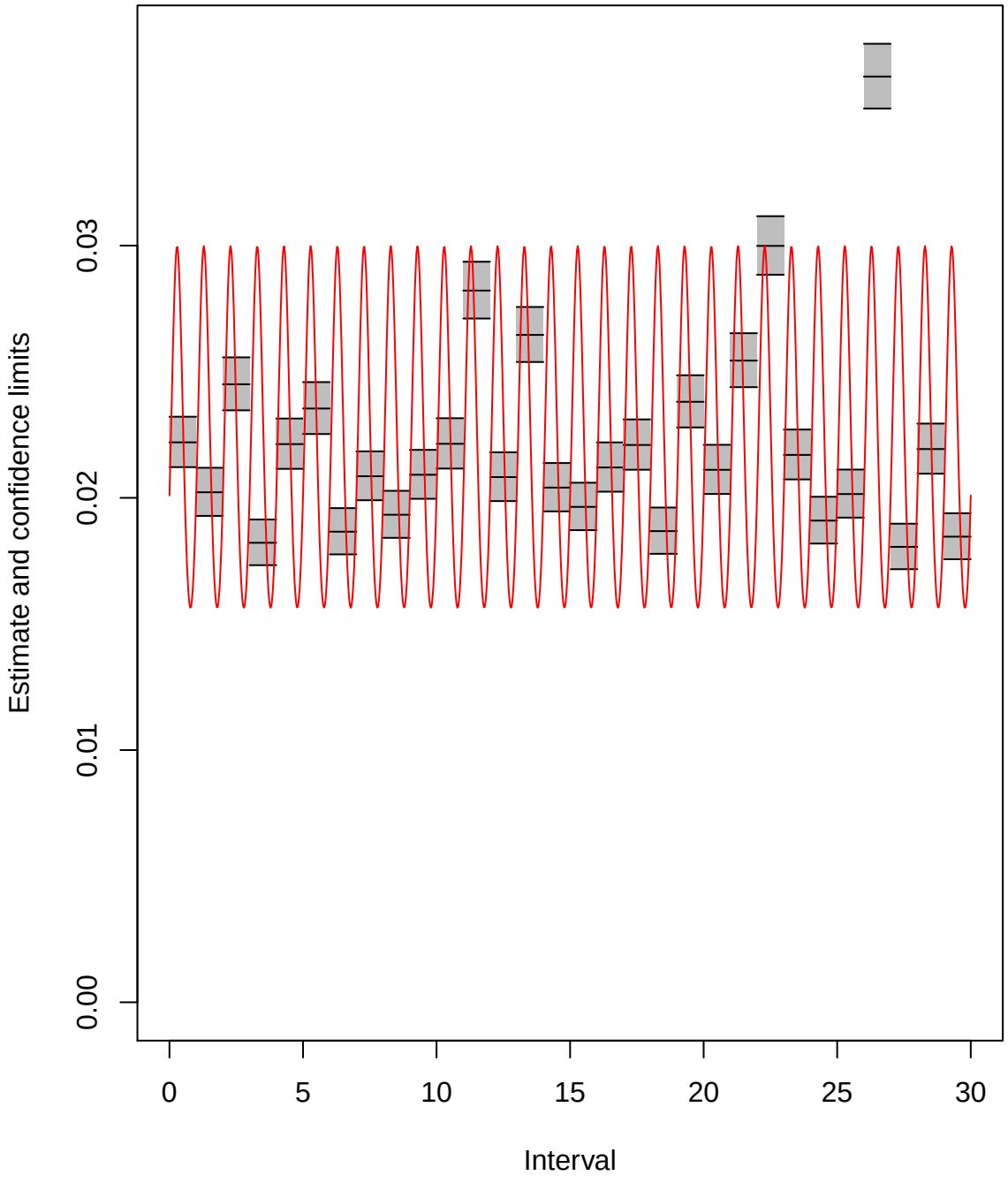

Figure 6.11: Confidence intervals with a 1 day grouping and an intensity function with one Fourier level fitted in red. The one day grouping is not fine enough to show any periodicity. n=1.

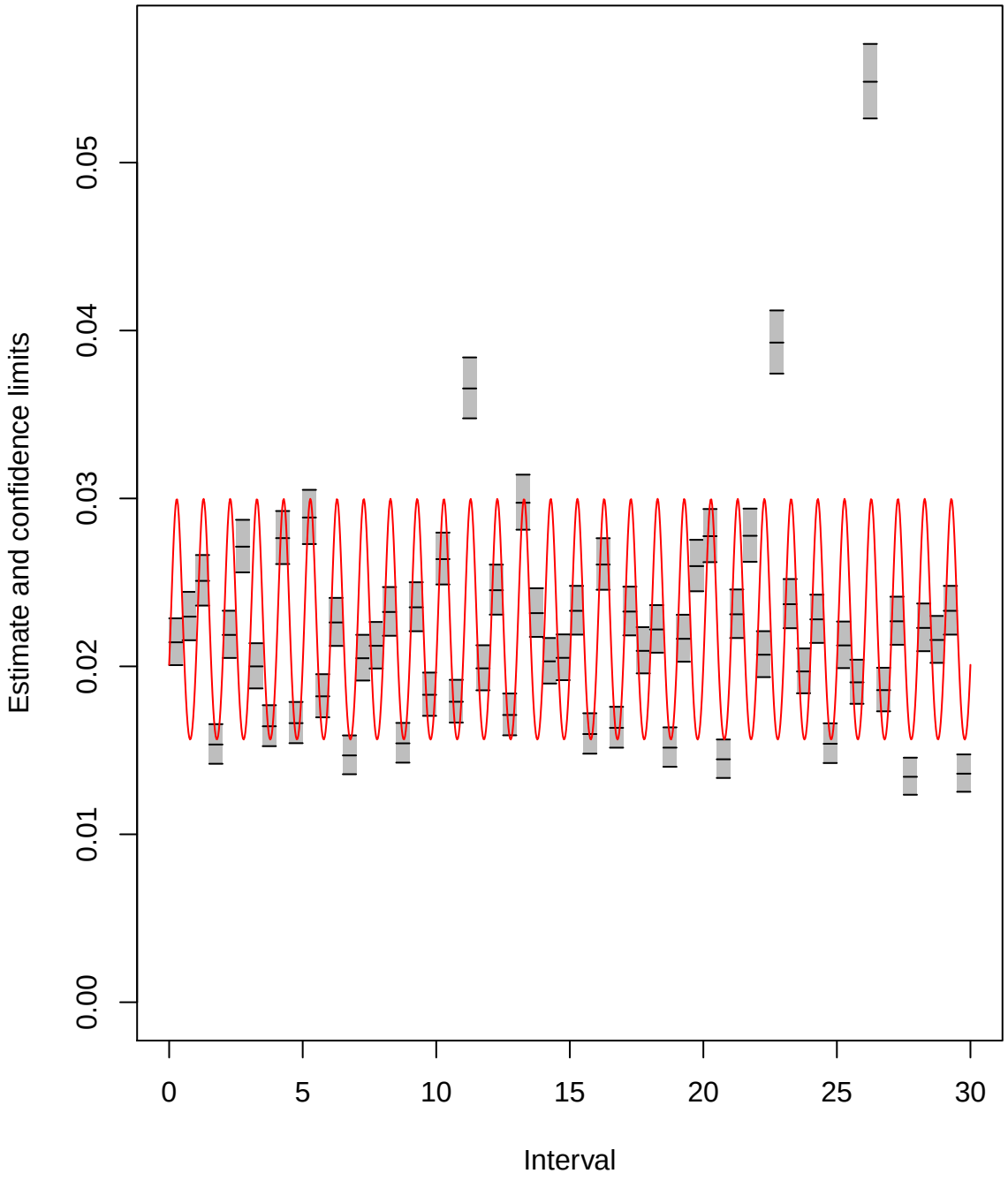

Figure 6.12: Confidence intervals with a 12 hour grouping. We are starting to see the peaks and troughs in the data matching the peaks and troughs in the model. n=1.

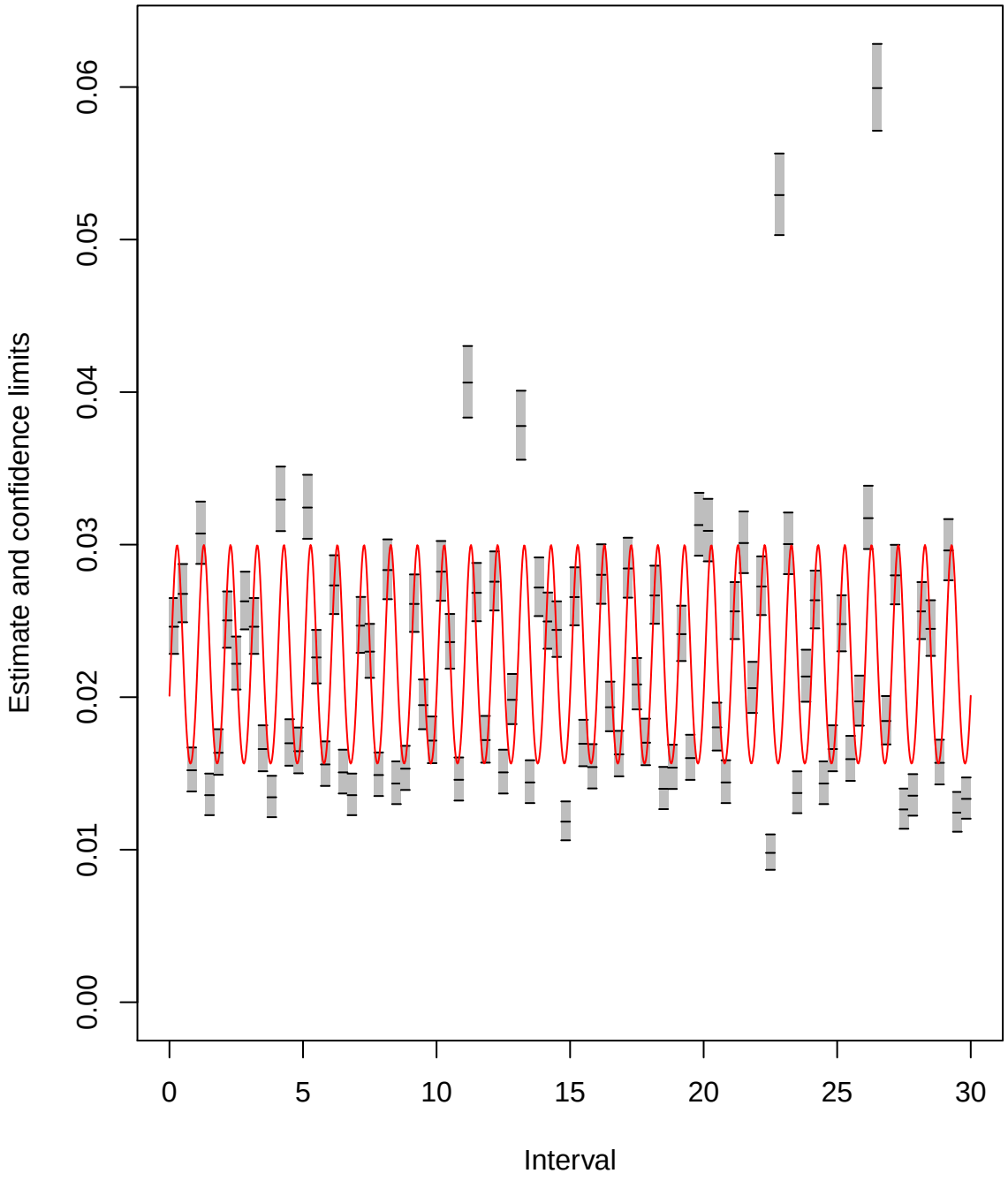

Figure 6.13: Confidence intervals with a 8 hour grouping. It is now clear that the peaks and troughs in the data matching the peaks and troughs in the model. n=1.

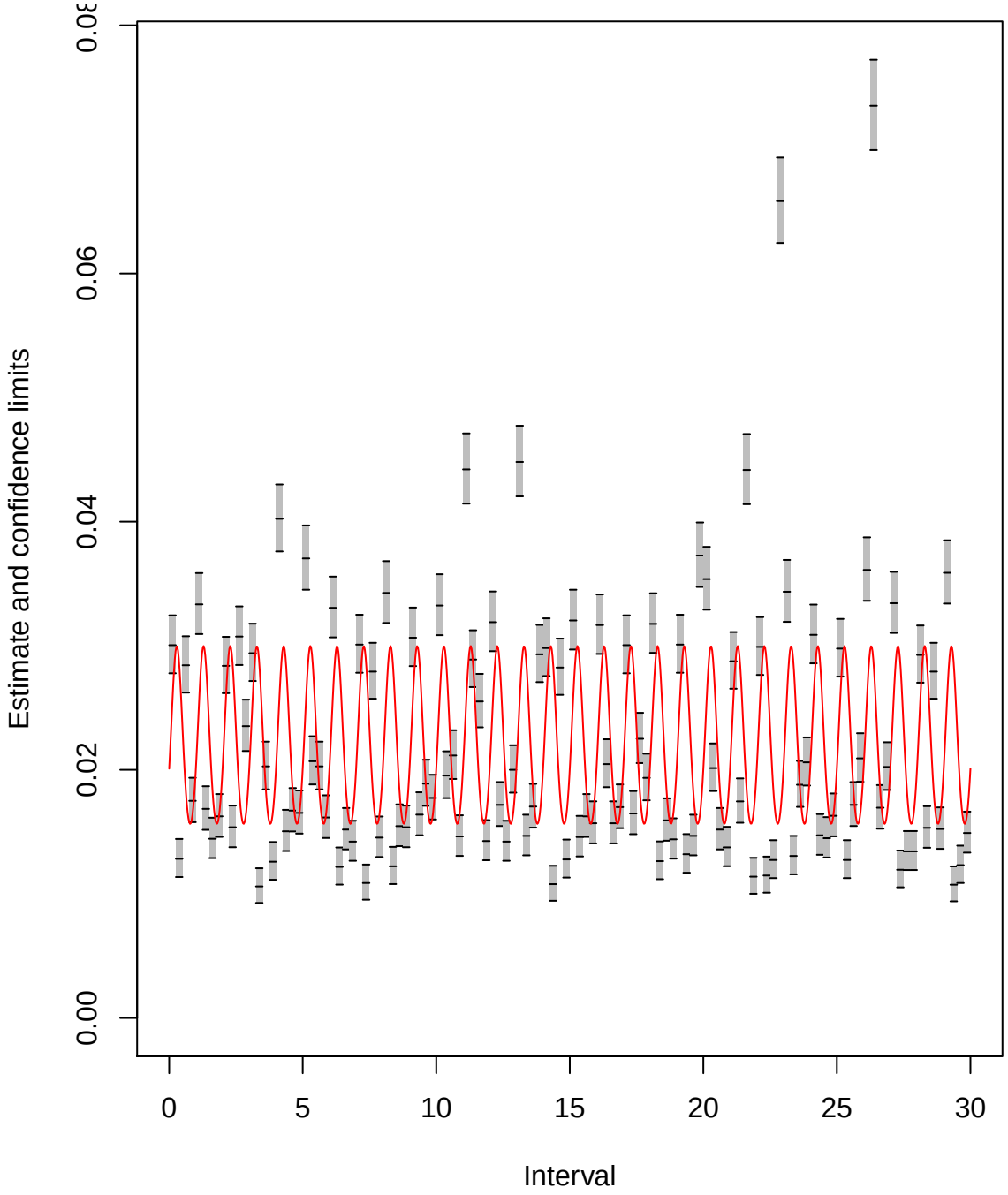

Figure 6.14: Confidence intervals with a 6 hour grouping. We can still see the peaks and troughs in the data matching the peaks and troughs in the model although the random variability in the data is starting to mask it. n=1.

n	a_0	b_1	c_1	b_2	c_2	b_3	c_3
0	-3.806						
1	-3.832	0.317	-0.0749				
2	-3.877	0.285	-0.104	0.0487	-0.457		
3	-3.922	0.0481	-0.528	-0.105	-0.412	-0.475	0.0530

Table 6.1: Table of estimated parameter values for the exponential Fourier model.

function $\lambda(x) = \exp\Big(a_0 + b_1 \sin(2\pi \frac{x}{T} f_1) + c_1 \cos(2\pi \frac{x}{T} f_1) + b_2 \sin(2\pi \frac{x}{T} f_2) + c_2 \cos(2\pi \frac{x}{T} f_2) + b_3 \sin(2\pi \frac{x}{T} f_3) + c_3 \cos(2\pi \frac{x}{T} f_3)\Big)$. And $f_2 = 2f_1, \quad f_3 = 3f_1$. The fitted model now has three peaks and troughs per day. In figs 6.20 and 6.21 the main peaks and troughs correspond to the data. However the third level peaks and troughs are becoming difficult to see clearly.

Table 6.1 contains the values of the fitted parameters which were estimated and used to draw the red continous line fitted models. The value of a_0 is almost constant for all n values. However is decreases slowly as n increases. For $n = 0 \quad \exp(a_0) = \exp(-3.806) = 0.022$ This is the value of the intensity of the homogeneous Poisson process for the whole data, as can be seen in fig 6.9 and 6.10.

My conclusion is that from looking at the graphs 6.9 to 6.10, The fitted models do fit the data.

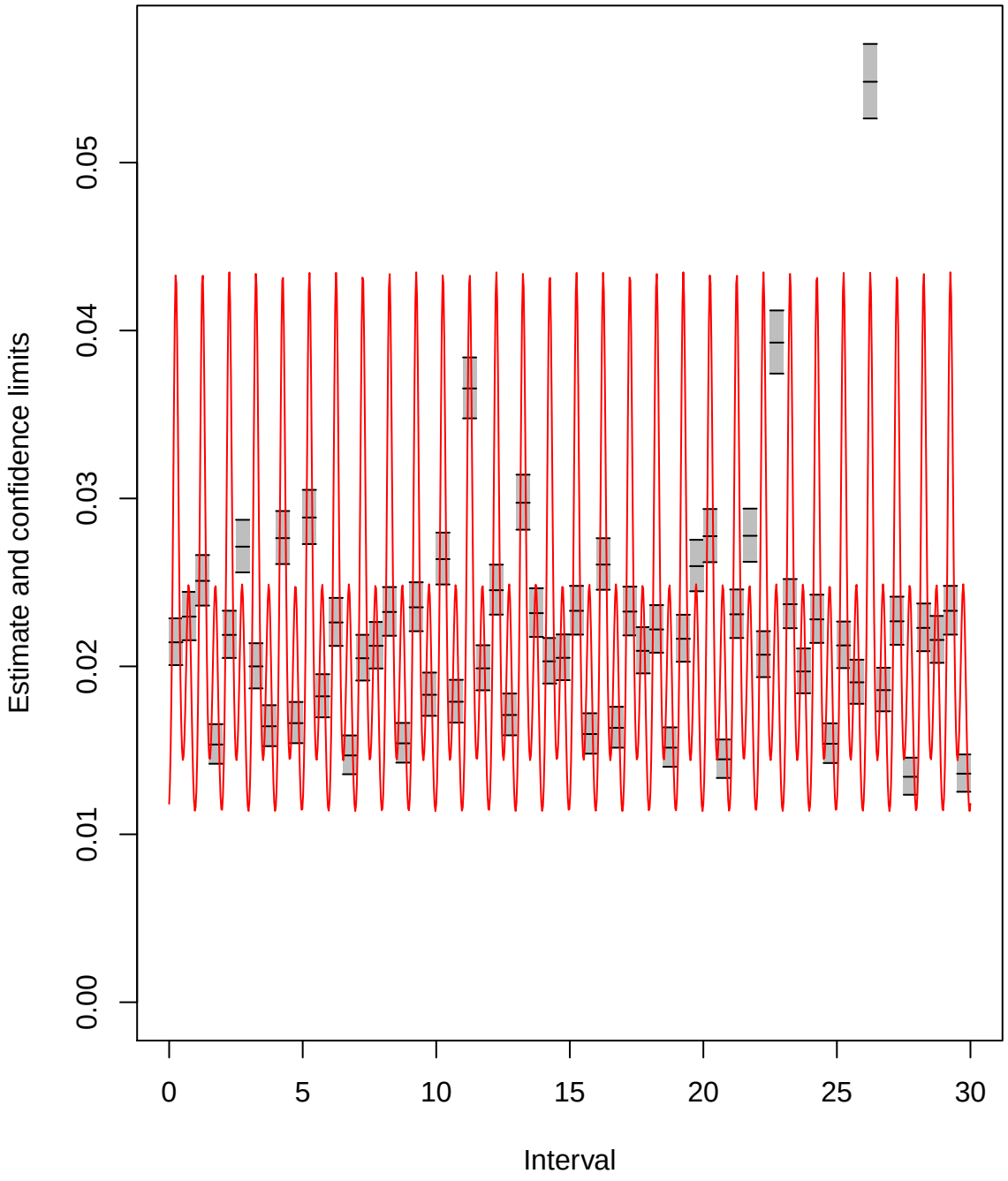

Figure 6.15: Confidence intervals with a 12 hour grouping. The graph is very cramped, it is difficult to see the fitted model clearly. n=2.

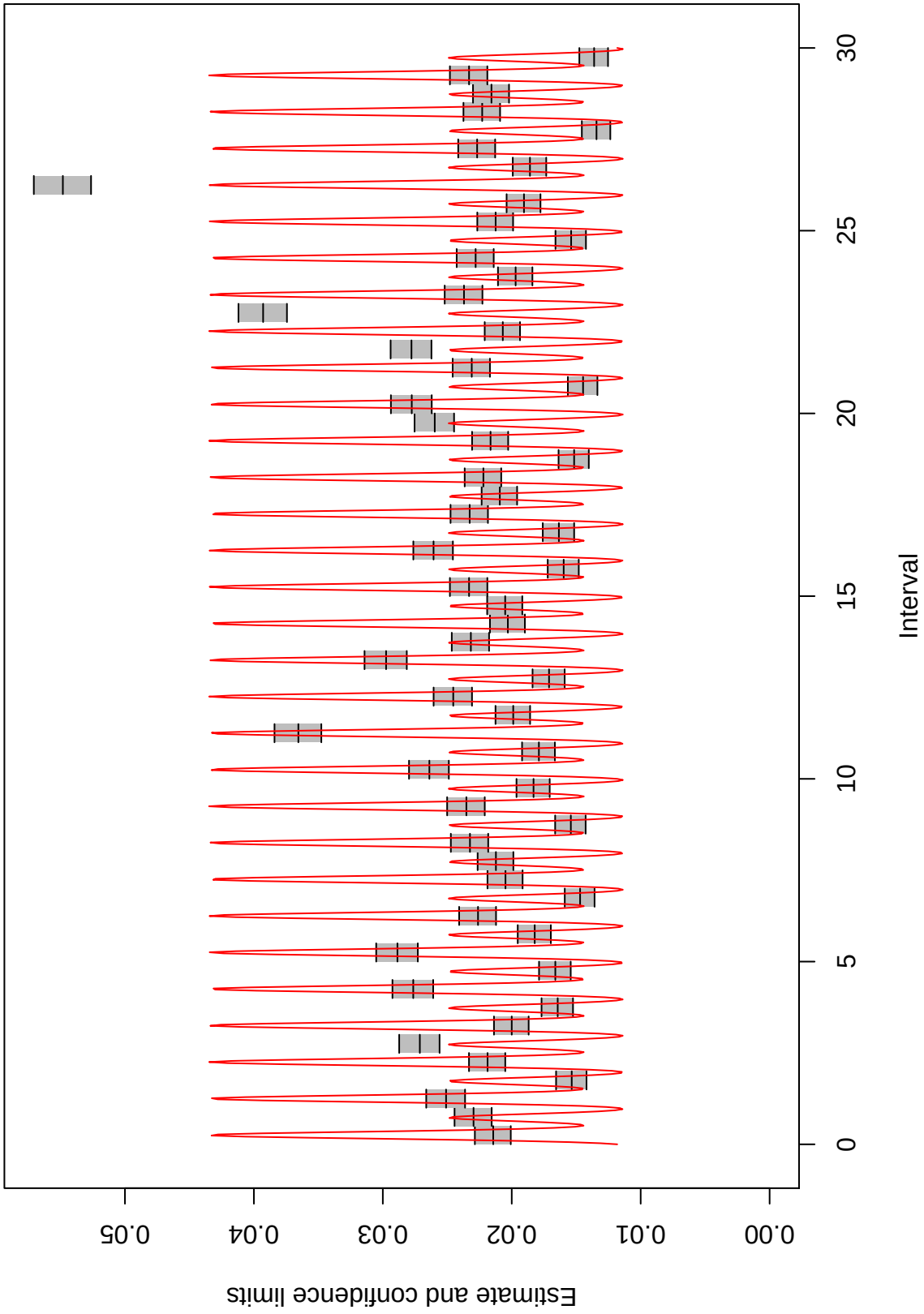

Figure 6.16: Confidence intervals with a 12 hour grouping. We are starting to see the peaks and troughs in the data matching the peaks and troughs in the model. n=2.

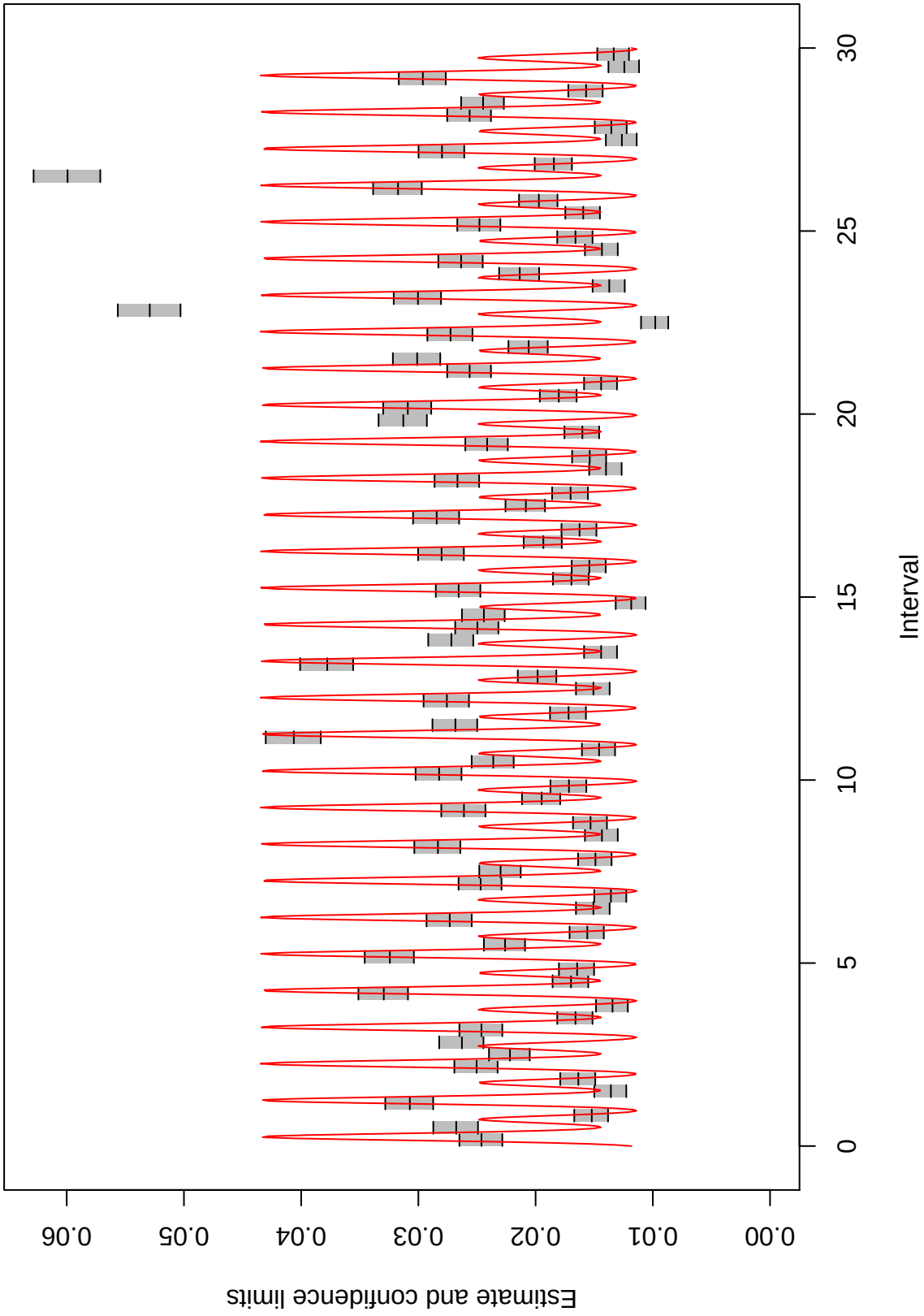

Figure 6.17: Confidence intervals with a 8 hour grouping. It is now clear that the main peaks and troughs in the data matching the peaks and troughs in the model. The secondary troughs are a better fit than the secondary peaks. Note the two high outliers corresponding to the secondary peaks! n=2.

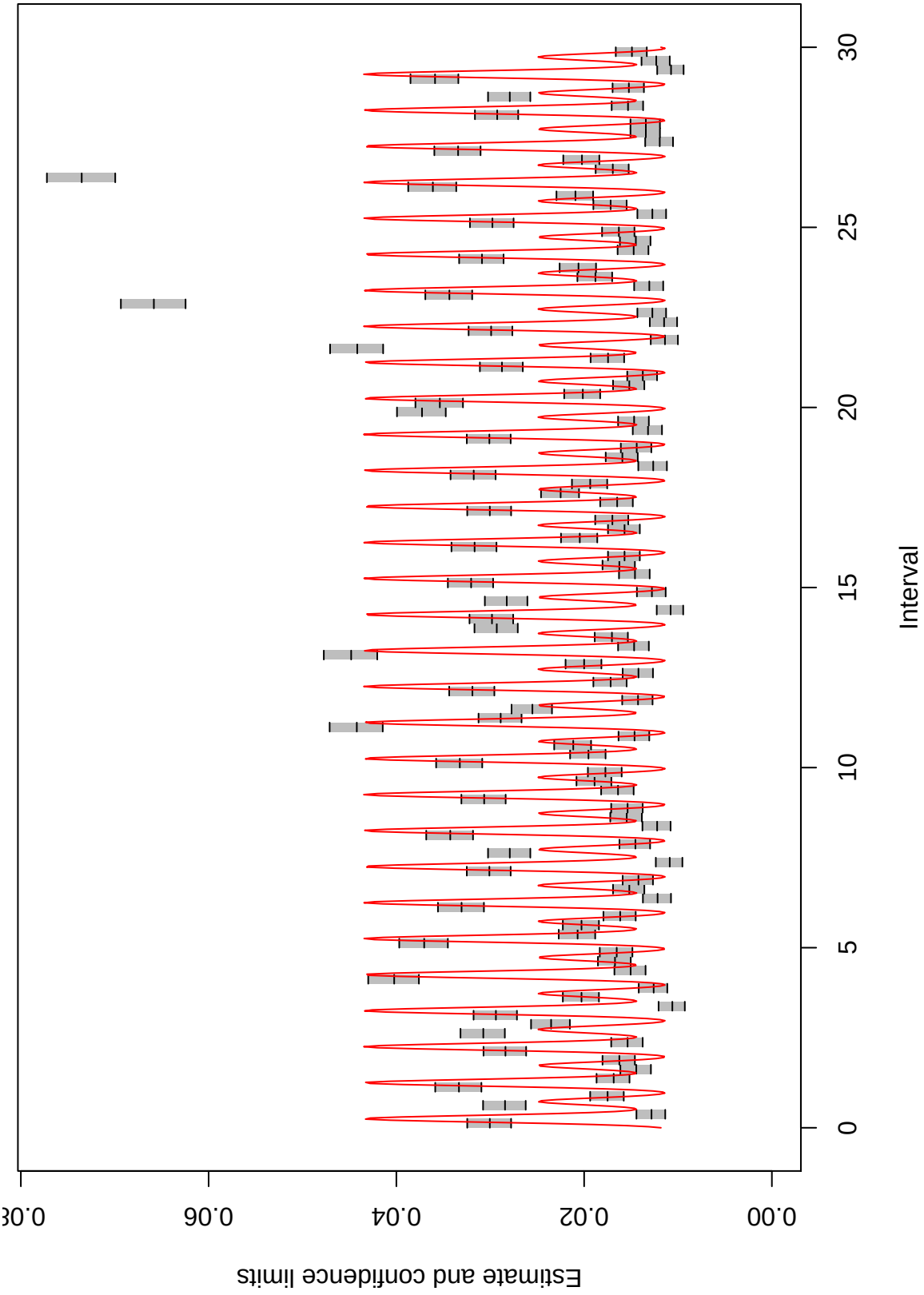

Figure 6.18: Confidence intervals with a 6 hour grouping. The main peaks match best the secondary peaks correspond to three high outliers. The troughs fitting is average. n=2

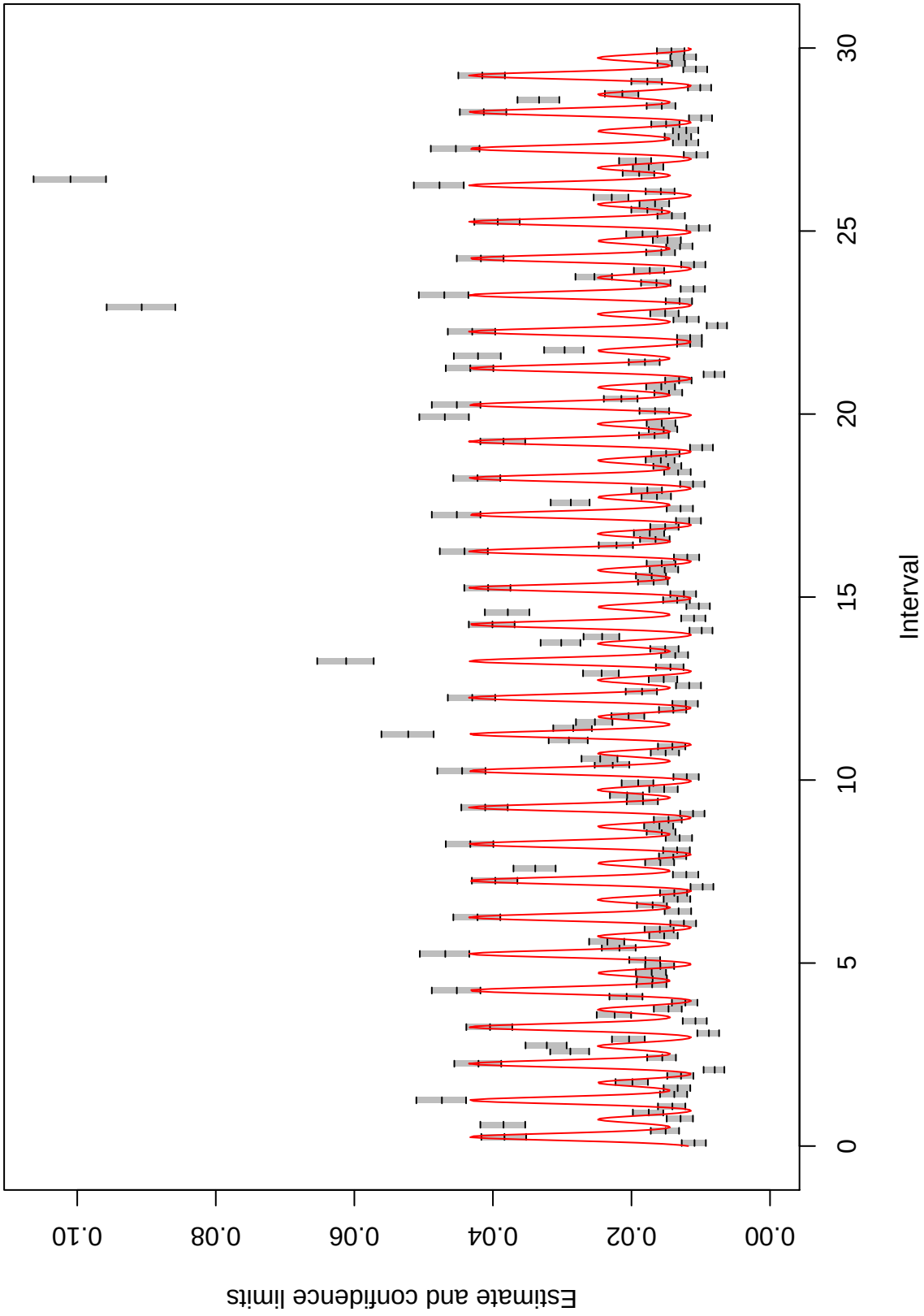

Figure 6.19: Confidence intervals with a 4 hour grouping. Both the main and secondary peaks and troughs correspond to the data. n=2.

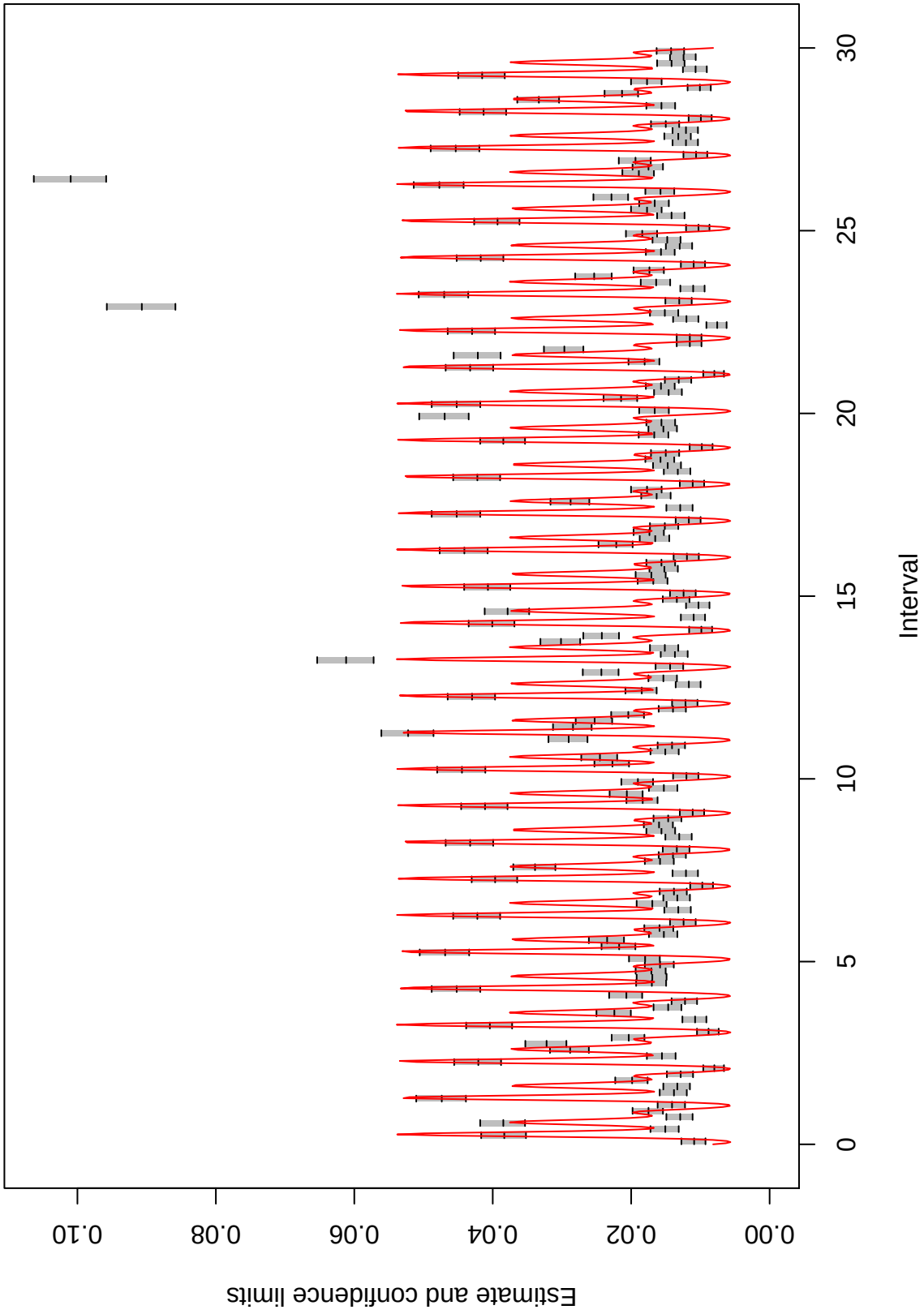

Figure 6.20: Confidence intervals with a 4 hour grouping. The main peaks and troughs correspond to the data. The secondary troughs are also a fairly good fit. The secondary peaks are a worse fit. It is very difficult to see how the tertiary peaks and troughs match the data. n=3.

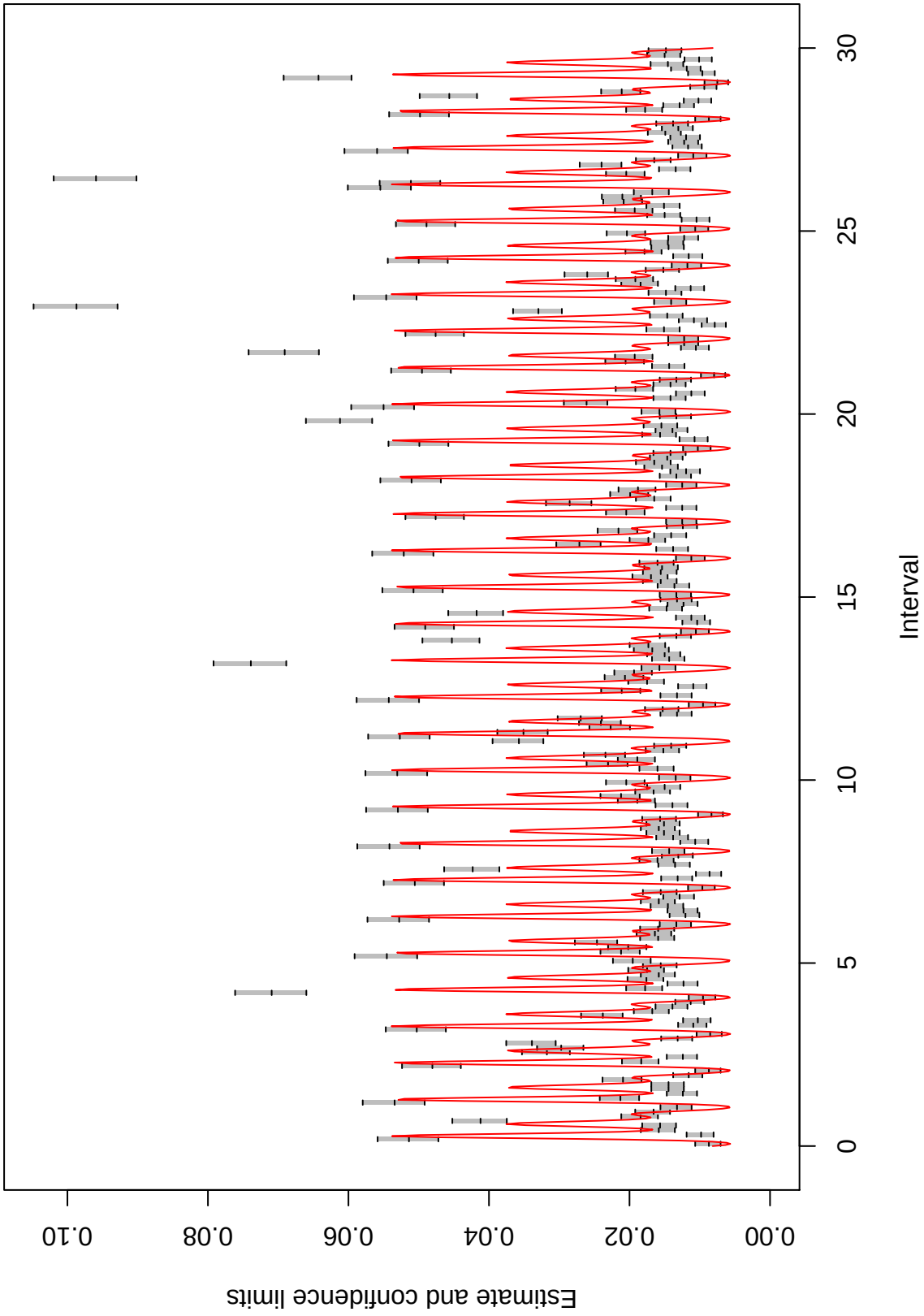

Figure 6.21: Confidence intervals with a 3 hour grouping. With the data rectangles more extreme is is easier to see that the main peaks and troughs correspond to the data. The secondary troughs are a fairly good fit. The secondary peaks are a worse fit. It is very difficult to see how the tertiary peaks and troughs match the data. $n=3$.

Chapter 7

Further work

Much more theoretical work could be done. This can be through an Internet search, researching through standard textbooks such as (Kutoyants, 1998) and independent original research. The theory that was neglected here is mostly on the formal statistical inference specific to advanced forms of the Poisson process.

The frequencies were assumed to be $f_1 = 1$, $f_n = nf_1$. In a more general program the frequencies could be estimated from the data itself.

Four classes of models were considered in this work. The model could be extended to other functional forms.

The computing could be extended in several directions. Other optimisation routines could be used as well as different options in the existing routines. We could try supplying the gradient function explicitly to the routines rather than having the routine approximating the gradient by finite difference methods. The program could also be properly designed and rewritten in a different language with a user friendly interface.

Chapter 8

Program code

The program was written in the R language. Please note that many of the indentations in the original R program have automatically been altered by the typesetting program used to write this dissertation !

```
# 1 - periodic model for the intensity lambda

seconds.per.day <- 24 * 60 * 60

f = function(x, a) {
  n = (length(a) - 1) / 2
  res = rep.int(a[1], length(x))
  if (n > 0) {
     for (j in 1:n) {
       fj = j / seconds.per.day
       res = res+a[2*j]*sin(fj*2*pi*x)+a[2*j+1]*cos(fj*2*pi*x)
     }
  }
  return(res)
}

lambda <- function(x, a) {
exp(f(x, a))
}

# testing:
# t <- seq(0, seconds.per.day, length=100)
# plot(t, lambda(t, c(1, 1, 0)), type="l")
# plot(t, lambda(t, c(1, 0, 1)), type="l")
# plot(t, lambda(t, c(1, 0, 0, 1, 0)), type="l")
```

```
# 2 - maximum likelihood fit for lambda

neg.log.likelihood <- function(a, x, L, U) {
integrate(lambda, L, U, a, subdivisions=5000)$value - sum(f(x, a))
}

find.optimal.parameters <- function(n, data, L, U, iterations=1000) {
a <- rep.int(0, 2*n+1)
res <- optim(a, neg.log.likelihood, NULL,
data, L, U,
control=list(maxit=iterations))
if (res$convergence != 0) {
cat("optim did not converge (code ", res$convergence,
"), use more iterations\n", sep="")
}
res$par
}

# testing:
 x <- scan("data3")
# find.optimal.parameters(2, x, 0, 30*seconds.per.day, iterations=1)
# find.optimal.parameters(2, x, 0, 30*seconds.per.day, iterations=100)
# find.optimal.parameters(2, x, 0, 30*seconds.per.day, iterations=1000)

# 3 - Confidence Intervals

interval.counts <- function(data, bins, a, b) {
counts <- c()
for (i in 1:bins) {
lower = a + (i-1)*(b-a)/bins
upper = lower + (b-a)/bins
counts[i] = sum(data>=lower & data<upper)
}
return(counts)
}

get.confidence.interval <- function(x, alpha=0.05) {
        if (x == 0) {
                L <- 0
                U <- - log(alpha)
        } else {
                f <- function(lambda) {
                        ppois(x, lambda) - alpha/2
                }
                lambda.upper <- x + (1 + sqrt(2*x*alpha + 1)) / alpha
                U <- uniroot(f, c(x, lambda.upper))$root
                g <- function(lambda) {
                        ppois(x - 1, lambda, lower.tail=FALSE) - alpha/2
```

```
                }
                L <- uniroot(g, c(0, x))$root
        }
        return(c(L,U))
}

#Input a vector of counts
#Output the lower and upper confidence limits for these counts
get.confidence.intervals <- function(counts, alpha) {
U <- c()
L <- c()

for (i in 1:length(counts)) {
UL = get.confidence.interval(counts[i], alpha)
L[i] = UL[1]
U[i] = UL[2]
}
return(c(L,U))
}

# 4 - plot the result

# bins= Number of intervals
do.one.plot = function(data,a,b,bins, modes, alpha=0.05, iterations=1000) {
par(oma=c(0, 0, 0, 0))
par(mai=c(0.8, 0.8, 0.1, 0.1))

counts = interval.counts(data, bins, a, b)
lud3 = get.confidence.intervals(counts, alpha)

# scale
est = counts * bins/(b-a)
LU = lud3 * bins/(b-a)

# draw the axes and labels
plot(0, 0, xlim=c(a,b)/seconds.per.day, ylim=c(0,max(LU)),
type="n",
xlab="Interval", ylab="Estimate and confidence limits")

# draw the boxes for the confidence intervals
for (i in 1:bins) {
x0 = a + (b-a) * (i-1) / bins / seconds.per.day
x1 = a + (b-a) * i / bins / seconds.per.day
rect(x0, LU[i], x1, LU[i+bins], col="gray", border=NA)
segments(x0=x0, y0=est[i], x1=x1,y1=est[i])
segments(x0=x0, y0=LU[i], x1=x1, y1=LU[i])
segments(x0=x0, y0=LU[i+bins], x1=x1, y1=LU[i+bins])
}
```

```
# draw the line for the intensity
t.sec = seq(a, b, length.out=1000)
t.day = t.sec / seconds.per.day

params <- find.optimal.parameters(modes, data, a, b, iterations)
lam <- lambda(t.sec, params)
lines(t.day, lam, col="red")

#lines(t.day, lambda2, col="blue")
}

do.one.pdf = function(data, a, b, bins, modes, alpha=0.05, iterations=1000) {
filename = sprintf("m58713-%d-%d-%02d.pdf", bins, modes, round(alpha*100))
pdf(filename, width=6, height=7)
do.one.plot(data, a, b, bins, modes, alpha, iterations)
dev.off()
}

x <- scan("data3")
do.one.pdf(x, 0, 30*seconds.per.day, 240, 3, iterations=5000)
do.one.pdf(x, 0, 30*seconds.per.day, 120, 1, iterations=5000)
```

Bibliography

G. P. Beaumont. *Intermediate Mathematical Statistics*. Chapman and Hall, 1980.

F. Daly, F. Hand, M. C. Jones, A. D. Lunn, and K. C. McConway. *Elements of Statistics*. Addison-Wesley, 1995.

Edward J. Dudewicz and S. N. Mishra. *Modern Mathematical Statistics*. Wiley, 1988.

A. Francis. *Advanced Level Statistics an integrated course*. Stanley Thornes, 1979.

J. F. C. Kingman. *Poisson Processes*. Oxford University Press, 1993.

Yu. A. Kutoyants. *Statistical Inference for Spatial Poisson Processes*. Springer-Verlag, 1998.

J. T. Byron Morgan. *Elements of Simulation*. Chapman and Hall, 1984.

M. Sheldon Ross. *Stochastic Processes*. Wiley, 1996.

M. Sheldon Ross. *Introduction to Probability Models*. Academic Press, 2000.

W. John Tukey. *Exploratory Data Analysis*. Addison Wesley, 1977.