MLE of Hawkes' Self-Exciting Point Process

RK

May 2, 2015

Abstract

The purpose of this document is to summarize the method formulated by T.Ozaki in the paper titled, "Maximum Likelihood Estimation of Hawkes' self-exciting point process"

Introduction

Let $N(t)$ be a point process such that

$$Pr(\Delta N(t) = 1|N(s)(s \leq t)) = \Lambda(t)\Delta t + o(\Delta t)$$
$$Pr(\Delta N(t) > 1|N(s)(s \leq t)) = o(\Delta t)$$

Hawke's model is a generalized point process model whose intensity function is given by

$$\Lambda(t) = \mu + \int_{-\infty}^{t} g(t-u)dN(u)$$

This paper deals with a response function of the type

$$g(t) = \alpha e^{-\beta t}$$

Log-likelihood of the Hawke's model

Given the occurrence observations $t_1, t_2, \ldots t_n$ for an interval $[0, T], (T \geq t_n)$, the log-likelihood of a point process with an intensity function

$$\Lambda(t|\theta) = \mu + \int_{-\infty}^{t} g(t-u|\theta)dN(u)$$

is given by

$$\log L(t_1, t_2, \ldots, t_n|\theta) = -\int_0^T \Lambda(t|\theta)dt + \int_0^T \log \Lambda(t|\theta)dN(t)$$

where $\theta = (\theta_1, \theta_2, \ldots \theta_n)$. This is the familiar expression $\log\left(\lambda t e^{-\lambda t}\right) = -\lambda t + \log(\lambda t)$, in the case of constant rate. The above log likelihood is defined under the assumption that the occurrence observations are observed from time $t = 0$ to a given time $T \geq t_n$. However in most identification problems, only $t_1, t_2, \ldots t_n$ are given and T is not specified. Hence the paper assumes $T = t_n$.

The log likelihood function for the specified intensity function is :

$$\log L(t_1, t_2, \ldots, t_n|\theta) = -\int_0^T \left[\mu + \int_{-\infty}^t \alpha e^{-\beta(t-u)} dN(u)\right] dt + \int_0^T \log\left(\mu + \int_{-\infty}^t \alpha e^{-\beta t} dN(u)\right) dt$$

Exchanging the variables u and t in the integrals
$\log L(t_1, t_2, \ldots, t_n|\theta) =$

$$-\mu t_n - \int_0^T \left[\int_u^{t_n} \alpha e^{-\beta(t-u)} dt\right] dN(u) + \int_0^T \log\left(\mu + \int_{-\infty}^t \alpha e^{-\beta t} dN(u)\right) dt$$

$\Rightarrow \log L(t_1, t_2, \ldots, t_n|\theta) =$

$$-\mu t_n + \int_0^T \left[\frac{\alpha}{\beta}\left(e^{-\beta(t_n-u)} - 1\right)\right] dN(u) + \int_0^T \log\left(\mu + \int_{-\infty}^t \alpha e^{-\beta t} dN(u)\right) dt$$

$\Rightarrow$

$$\log L(t_1, t_2, \ldots, t_n|\theta) = -\mu t_n + \sum_{i=1}^n \left[\frac{\alpha}{\beta}\left(e^{-\beta(t_n-t_i)} - 1\right)\right] + \sum_{i=1}^n \log(\mu + \alpha A(i))$$

where $A(i) = \sum_{t_j<t_l} e^{-\beta(t_i-t_j)}$

The author also derives the gradient and hessian for the above likelihood function. Given the likelihood function, gradient and Hessian, one can use any of the three methods that are mentioned in the paper. I have used `R` to work through this paper. I have used `nlm` function that does nonlinear optimization. You can specify the gradient and Hessian, but you have the option of allowing the function to figure out those quantities.

Simulation

This section contains a detailed method to simulate data from a Hawkes' process. This is mainly done to verify the results of maximum likelihood estimates. The basic idea behind simulation follows from the conditional Hazard function

$$\frac{f(t|t_1, \ldots, t_n, \theta)}{1 - F(t|t_1, t_2, \ldots t_k, \theta)} = \Lambda(t|t_1, t_2, \ldots t_k, \theta)$$

$\Rightarrow$

$$\log(1 - F(t|t_1, t_2, \ldots t_k, \theta)) = -\int_{t_k}^u \Lambda(t|t_1, t_2, \ldots, t_k, \theta) dt = -\int_{t_k}^u \mu + \sum_{i=1}^k g(t - t_i|\theta) dt$$

In the above equation, the time of $(k+1)$st event u satisfied the above equation, one can generate a uniform random variable U and solve for u in the following equation to generate the time of the next event

$$\log U + \int_{t_k}^u (\mu + \sum_{i=1}^k g(t - t_i|\theta)) dt = 0$$

For the exponentially decaying response function, the above equation reduces to

$$\log U + \mu(u - t_k) - \frac{\alpha}{\beta}(\sum_{i=1}^k e^{-\beta(u-t_i)} - \sum_{i=1}^k e^{-\beta(t_k-t_i)}) = 0$$

Consider the expression in the above equation

$$Y(k) = \sum_{i=1}^{k} e^{-\beta(t_k - t_i)}) - \sum_{i=1}^{k} e^{-\beta(u - t_i)}$$

This can be written as

$$\sum_{i=1}^{k} e^{-\beta(t_k - t_i)} - \sum_{i=1}^{k} e^{-\beta(t_k - t_i)} e^{-\beta(u - t_k)}$$

$\Rightarrow$

$$\left(1 - e^{-\beta(u - t_k)}\right) \sum_{i=1}^{k} e^{-\beta(t_k - t_i)}$$

$\Rightarrow$

$$\left(1 - e^{-\beta(u - t_k)}\right) (1 + e^{-\beta(t_k - t_{k-1})} \sum_{i=1}^{k-1} e^{-\beta(t_{k-1} - t_i)})$$

$\Rightarrow$

$$Y(k) = S(k) \left(1 - e^{-\beta(u - t_k)}\right), \quad S(k) = e^{-\beta(t_k - t_{k-1})} S(k-1) + 1$$

Hence one can solve the following equation

$$\log U + \mu(u - t_k) - \frac{\alpha}{\beta}(\sum_{i=1}^{k} e^{-\beta(u - t_i)} - \sum_{i=1}^{k} e^{-\beta(t_k - t_i)}) = 0$$

using the following recursion

$$\log U + \mu(u - t_k) + \frac{\alpha}{\beta} S(k) \left(1 - e^{-\beta(u - t_k)}\right) = 0$$

where $S(1) = 1$ and $S(k) = e^{-\beta(t_k - t_{k-1})} S(k-1) + 1$

In order to solve the equation numerically, one might require the first derivative of the above equation. The following procedure is given in the paper to solve the above equation :

1. Take the initial value as $t_k - \log U/\mu$
2. Update the value as follows

$$u_{i+1} = u_i - \frac{f(u_i)}{f'(u_i)}$$

where

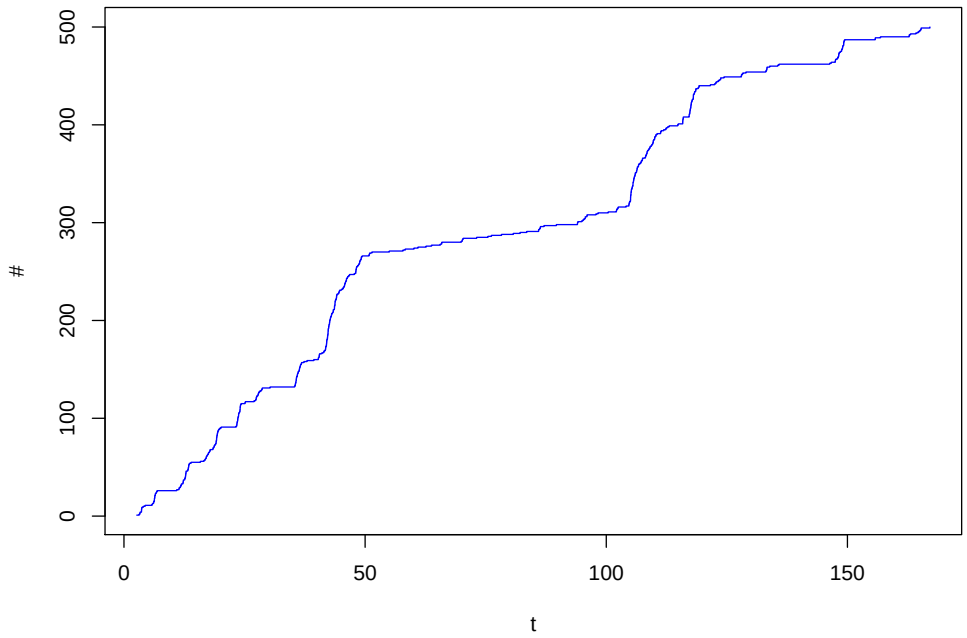
$$\begin{aligned} f(u) =& f(u) = \log U + \mu(u - t_k) + \frac{\alpha}{\beta} S(k) \left(1 - e^{-\beta(u - t_k)}\right) \\ f'(u) =& \mu + \alpha S(k) \left(e^{-\beta(u - t_k)}\right) \end{aligned}$$

3. Stop the iteration when $|u_{i+1} - u_i| \leq \epsilon$

Simulation of univariate Hawkes' process in R

```
simulate_uni_hawkes <- function(mu, alpha, beta, n){
  arrivals     <- numeric()
  eps          <- 1e-6

  # recurrence function
  S            <- function(k){
    if(k==1) return(1)
    return( 1 + S(k-1)* exp(-beta * (arrivals[k]- arrivals[k-1])) )
  }
  # the function to be solved to obtain, u
  fu           <- function(u, U, k){
    return(log(U) + mu*(u - arrivals[k]) +
             alpha/beta * S(k) * (1 - exp(-beta * (u - arrivals[k]))))
  }
    fu_prime   <- function(u, U , k){
    return( mu + alpha * S(k) * (exp(-beta * (u - arrivals[k]))))
  }

  # iterative procedure to solve f(u)
  solve_u      <- function(U,k) {
    u_prev     <- arrivals[k] - log(U)/mu
    u_next     <- u_prev - fu(u_prev, U, k)/ fu_prime(u_prev, U, k)
    while( abs(u_next - u_prev) > eps){
      u_prev   <- u_next
      u_next   <- u_prev - fu(u_prev, U, k)/fu_prime(u_prev, U, k)
    }
    return(0.5 * (u_prev + u_next))
  }

  set.seed(1)
  t1           <- -log(runif(1))/mu
  arrivals     <- c( arrivals, t1)
  k            <- length(arrivals)
  while(k < n) {
    U          <- runif(1)
    t_next     <- solve_u( U, k)
    arrivals   <- c(arrivals, t_next)
    k          <- length(arrivals)
  }
  return(arrivals)
}
```

Simulating Hawkes' with parameters $(\alpha, \beta, \mu) = (4, 5, 0.5)$



Estimating via MLE

The log likelihood given in the paper is coded below

```
loglik     <- function(params, arrivals){
  mu_i     <- params[1]
  alpha_i  <- params[2]
  beta_i   <- params[3]
  term_1   <- -mu_i*arrivals[n]
  term_2   <- sum(alpha_i/beta_i*(exp( -beta_i * (arrivals[n] - arrivals)) - 1))
  Ai       <- c(0, sapply(2:n, function(z) {
              sum(exp( -beta_i * (arrivals[z]- arrivals[1:(z - 1)])))
             }))
  term_3  <- sum(log( mu_i + alpha_i * Ai))
  return(-term_1- term_2 -term_3)
}
```

Using `optim` function

```
case1_solution1  <- optim(c(0.1,1,2), loglik, arrivals = sample_f1)
paste( c("mu","alpha", "beta"), round(case1_solution1$par,2), sep=" = ")

## [1] "mu = 0.57"    "alpha = 3.54" "beta = 4.36"
```

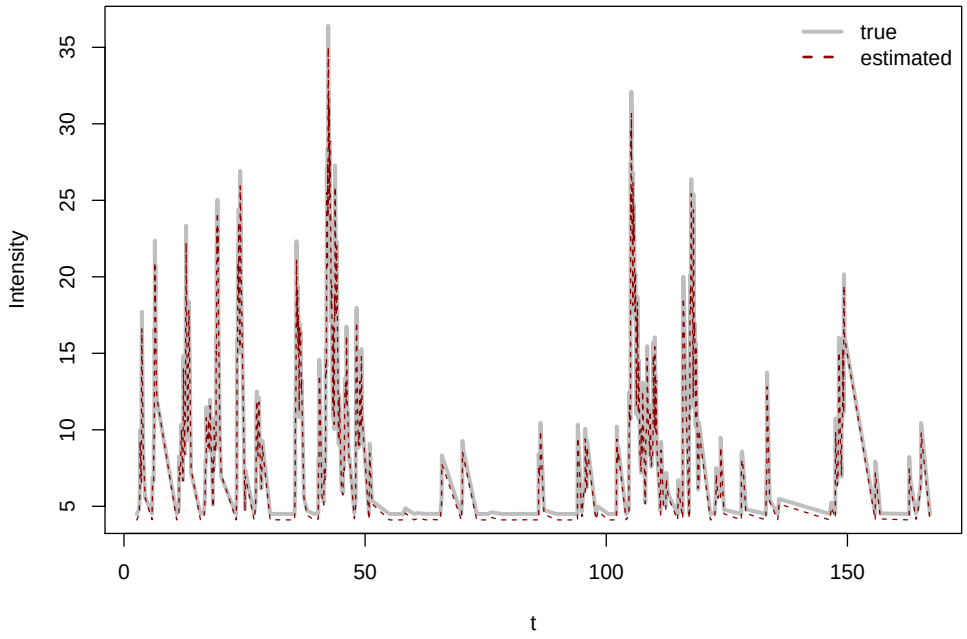
Using `nlm` function

```
case1_solution2  <- nlm(loglik, c(0.1,1, 6), hessian = TRUE, arrivals = sample_f1)
paste( c( "mu", "alpha", "beta"), round(case1_solution2$estimate,2), sep=" = ")

## [1] "mu = 0.57"    "alpha = 3.54" "beta = 4.36"

cov_matrix       <- solve(case1_solution2$hessian)
print(cov_matrix)

##          [,1]   [,2]     [,3]
## [1,] 0.006471 0.0030 0.009212
## [2,] 0.003000 0.1539 0.156905
## [3,] 0.009212 0.1569 0.209999
```

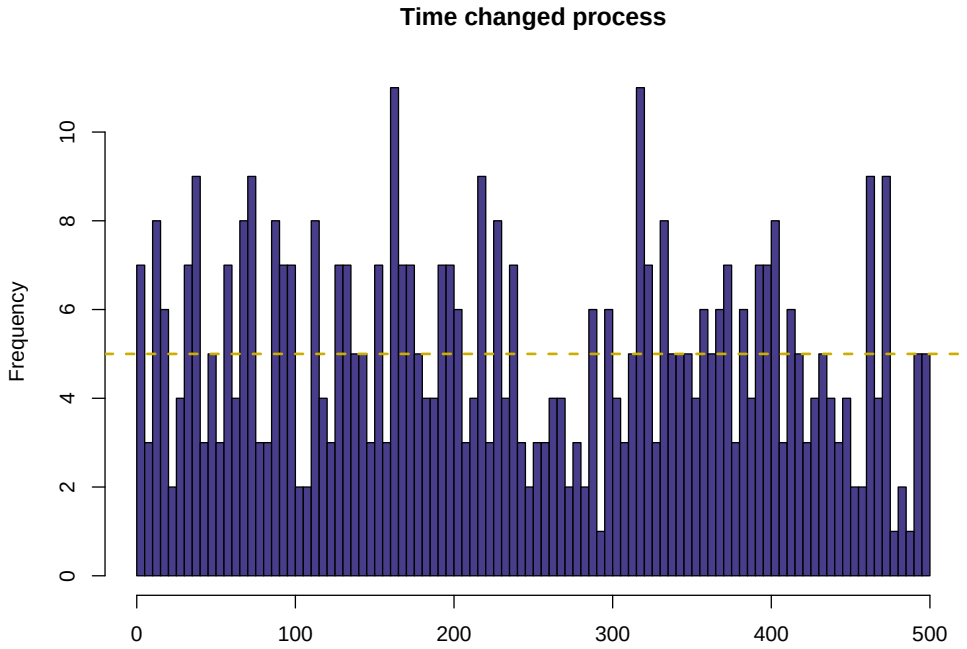
Estimated vs. Actual intensity function

```
estimated_intensity  <- function(params, arrivals){
  mu       <- params[1]
  alpha    <- params[2]
  beta     <- params[3]
  Ai       <- sapply(1:n, function(z) {
              sum(exp( -beta * (arrivals[z]- arrivals[1:z])))
             })
  return(mu + alpha *Ai)
}
```

```
compensator       <- function(params, arrivals){
  mu     <- params[1]
  alpha  <- params[2]
  beta   <- params[3]
  result <- sapply(1:n, function(z){
    mu*arrivals[z] + sum(alpha/beta*(1-exp(-beta*(arrivals[z]-arrivals[1:z]))))
  })
  return(result)
}
```

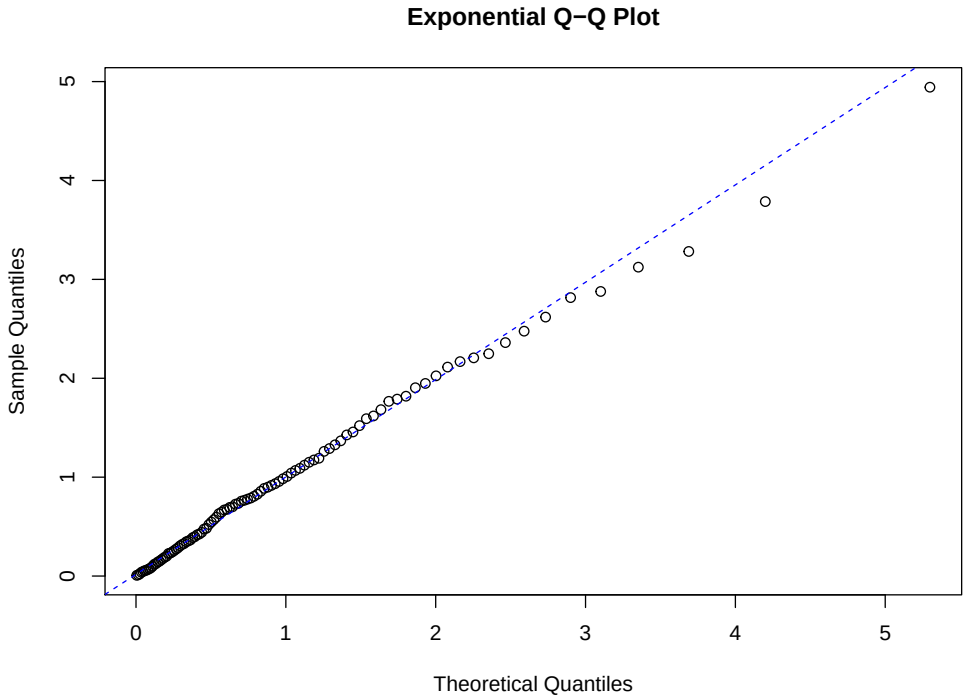


Another way to test the parameters is to apply time change to the original process and check whether the resulting process is a standard Poisson.

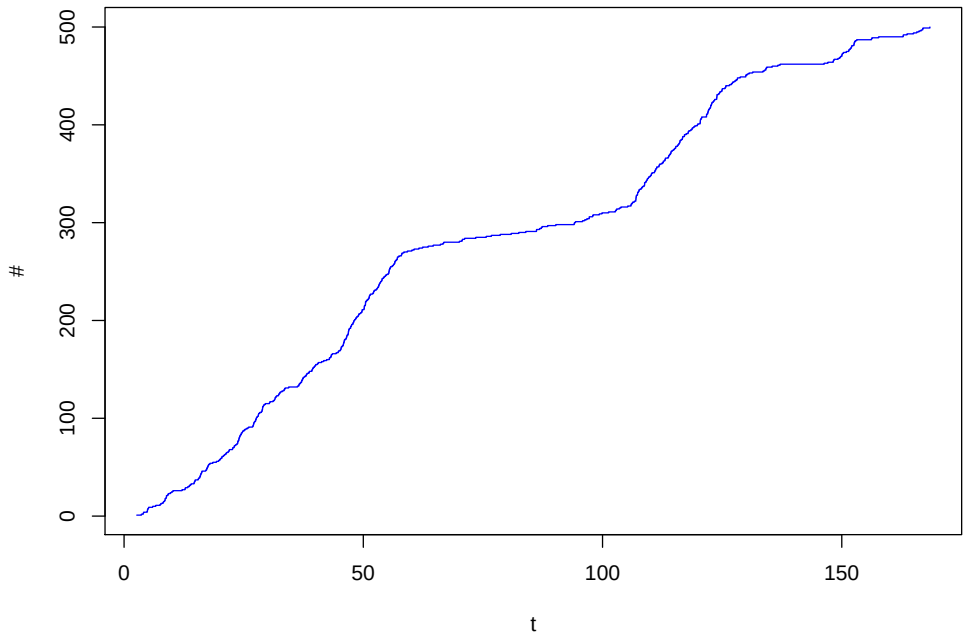


Splitting the time interval in to steps of 5 units should result in a frequency count of 5. The above histogram confirms that the parameters estimation via MLE has not gone wrong

QQ PLOT


Exponential Q−Q Plot
Sample Quantiles
Theoretical Quantiles
0
1
2
3
4
5

Simulating Hawkes' with parameters $(\alpha, \beta, \mu) = (0.8, 1, 0.5)$



Estimating via MLE

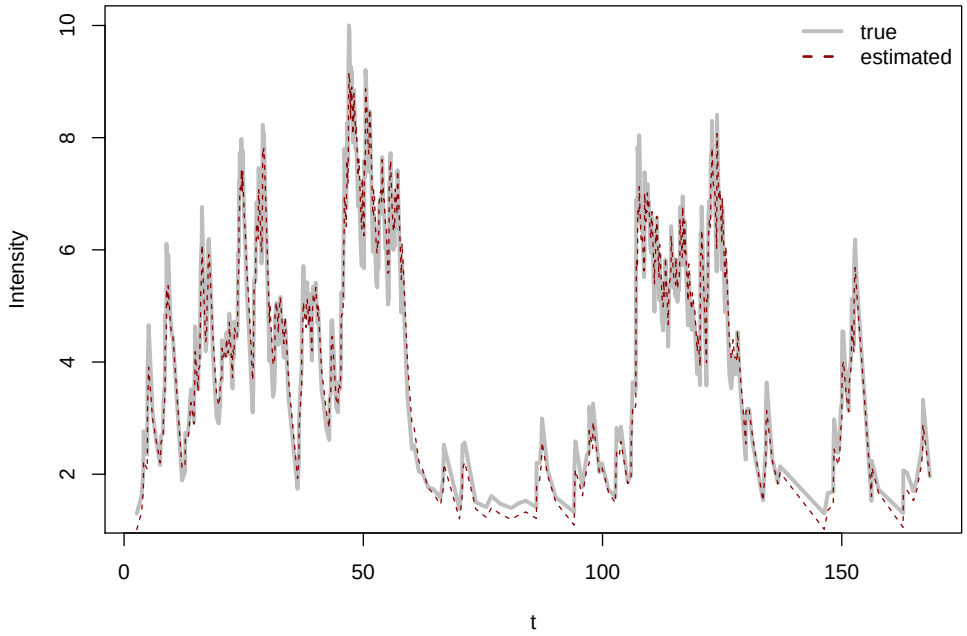
Using `optim` function

```
case2_solution1  <- optim(c(0.1, 1,2), loglik, arrivals = sample_f2)
paste( c("mu","alpha", "beta" ), round(case2_solution1$par,2), sep=" = ")

## [1] "mu = 0.43"    "alpha = 0.59" "beta = 0.68"
```

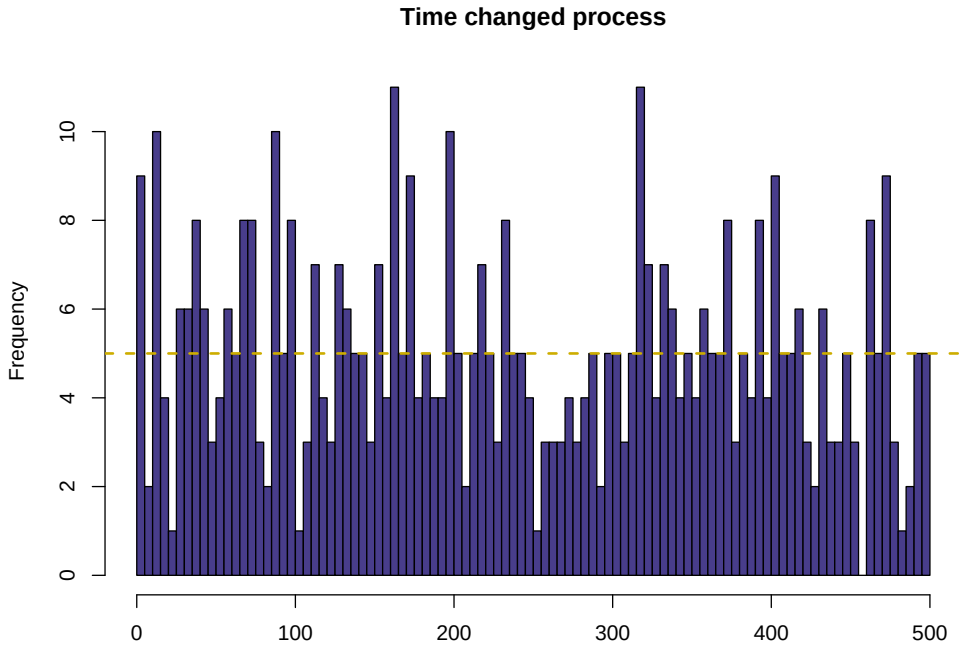
Using `nlm` function

```
case2_solution2  <- nlm(loglik, c(0.1, 1, 6), hessian = TRUE, arrivals = sample_f2)
paste( c("mu", "alpha", "beta"), round(case2_solution2$estimate,2), sep=" = ")

## [1] "mu = 0.43"    "alpha = 0.59" "beta = 0.68"

cov_matrix       <- solve(case1_solution2$hessian)
print(cov_matrix)

##          [,1]   [,2]     [,3]
## [1,] 0.006471 0.0030 0.009212
## [2,] 0.003000 0.1539 0.156905
## [3,] 0.009212 0.1569 0.209999
```

Estimated vs. Actual intensity function

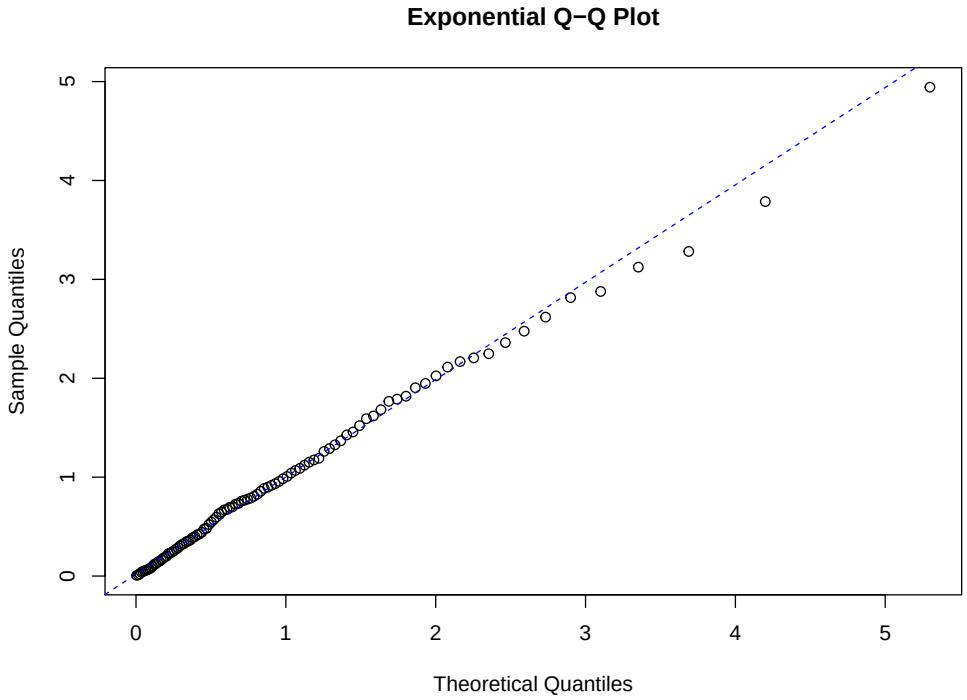


Another way to test the parameters is to apply time change to the original process and check whether the resulting process is a standard Poisson.



Splitting the time interval in to steps of 5 units should result in a frequency count of 5. The above histogram confirms that the parameters estimation via MLE has not gone wrong

QQ PLOT


Exponential Q−Q Plot
Sample Quantiles
Theoretical Quantiles
0
1
2
3
4
5