HawkesModel

Fitting Hawkes model to marathon record data

Produce estimates for model, exponential Hawkes process, using MLE: - baseline intensity, - reproduction mean
- exponential fertility function rate

Note: data is in days from the first world record which is set as time 0. The model is fit through day 40,300 (the last record is at day 40,231).

Three parameter estimates are first output.

NOTE: the results change if one doesn't set a seed. . . and by more than I would expect!!!! Maybe need to set starting values or something to ensure optimal?

```
record_table_mod<-read_rds("record_table_mod.rds")

days_between = as.numeric(diff(record_table_mod$Date_ymd))
daysfromstart <- cumsum(days_between)
daysfromstart <- c(0,daysfromstart)  ### get data in terms of days from first record (time 0)

set.seed(86)
optMarathon<-mle(daysfromstart,"Exponential",40300)  # end date picked number greater than longest time
optMarathon$par
```

```
## [1] 0.001198153 0.488775833 1.804699115
```

```
summary(optMarathon)
```

```
##        Length Class            Mode
## par     3     -none-           numeric
## model   1     Rcpp_Exponential S4
## events 50     -none-           numeric
## end     1     -none-           numeric
## opt    20     nloptr           list
```

```
optMarathon$events
```

```
##  [1]     0   161   203   288   290   403  1753  1772  2319  4412  6289  7651
## [13]  9736  9746  9749  9963 14148 16031 16395 16508 16773 17551 18293 19041
## [25] 19931 20049 20070 20413 20543 20777 21681 22225 22644 23932 25398 26209
## [37] 26798 27848 28029 29122 32930 33329 34232 34764 36227 36591 37683 38418
## [49] 38782 40231
```

```
optMarathon$end
```

```
## [1] 40300
```

```
optMarathon$model$param
```

```
##              [,1]
## [1,] 0.001198153
## [2,] 0.488775833
## [3,] 1.804699115
```

```
optMarathon$model$mean()  # expected value
```

```
## [1] 0.002343693
```

```
optMarathon$model$dmean()  # Jacobian matrix of expected value
```

```
##              [,1]
## [1,] 1.956089060
## [2,] 0.004584473
## [3,] 0.000000000
```

```
optMarathon$model$ddmean()  # Hessian matrix of expected value
```

```
##          [,1]       [,2] [,3]
## [1,] 0.000000 3.82628441    0
## [2,] 3.826284 0.01793527    0
## [3,] 0.000000 0.00000000    0
```

```
optMarathon$model$loglik(daysfromstart, optMarathon$end)  #  log-likelihood
```

```
## [1] -404.5783
```

```
optMarathon$model$dloglik(daysfromstart, optMarathon$end)  # Jacobian matrix of log-lik
```

```
##            [,1]
## [1,]  -3.386179
## [2,] -46.484059
## [3,]  -3.251138
```

```
optMarathon$model$ddloglik(daysfromstart, optMarathon$end) # hessian matrix of log-lik
```

```
##                [,1]        [,2]        [,3]
## [1,] -3.278008e+07 -383.530165 420.5948277
## [2,] -3.835302e+02   -6.253200  -1.0310182
## [3,]  4.205948e+02   -1.031018   0.6398502
```

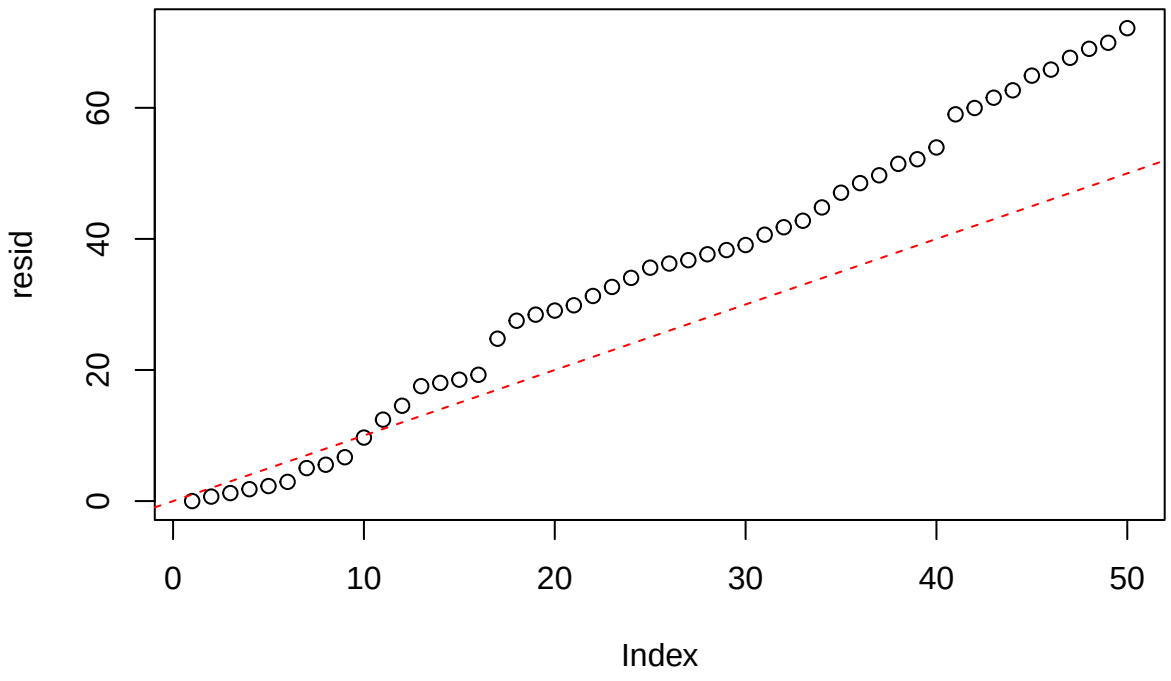
Residuals

From the help: "Outputs the residuals (values of the compensator at the times of arrival) of a Hawkes process. Useful function for diagnosis through the random time change theorem: the residuals should follow a unit rate Poisson process" Based on the example in the help I assume should follow the y=x line... we see divergence here suggesting an issue (in what direction?)

```
resid<-residuals(daysfromstart, fun = optMarathon$par[1], repr = optMarathon$par[2],
                 family = "exp", rate = optMarathon$par[3])
resid
```

```
##  [1]  0.0000000  0.6816784  1.2207766  1.8113955  2.2893373  2.9267347
##  [7]  5.0330165  5.5445573  6.6887226  9.6852318 12.4229401 14.5435998
## [13] 17.5305239 18.0312812 18.5214748 19.2688320 24.7718765 27.5167737
## [19] 28.4416771 29.0658442 29.8721305 31.2930691 32.6708741 34.0558681
## [25] 35.6109998 36.2411576 36.7550947 37.6548368 38.2993725 39.0685161
## [31] 40.6404219 41.7809927 42.7717945 44.8037909 47.0490585 48.5095361
## [37] 49.7040238 51.4508599 52.1565013 53.9548580 59.0061990 59.9730377
## [43] 61.5437453 62.6699384 64.9116115 65.8365149 67.6336733 69.0030913
## [49] 69.9279947 72.1528937
```

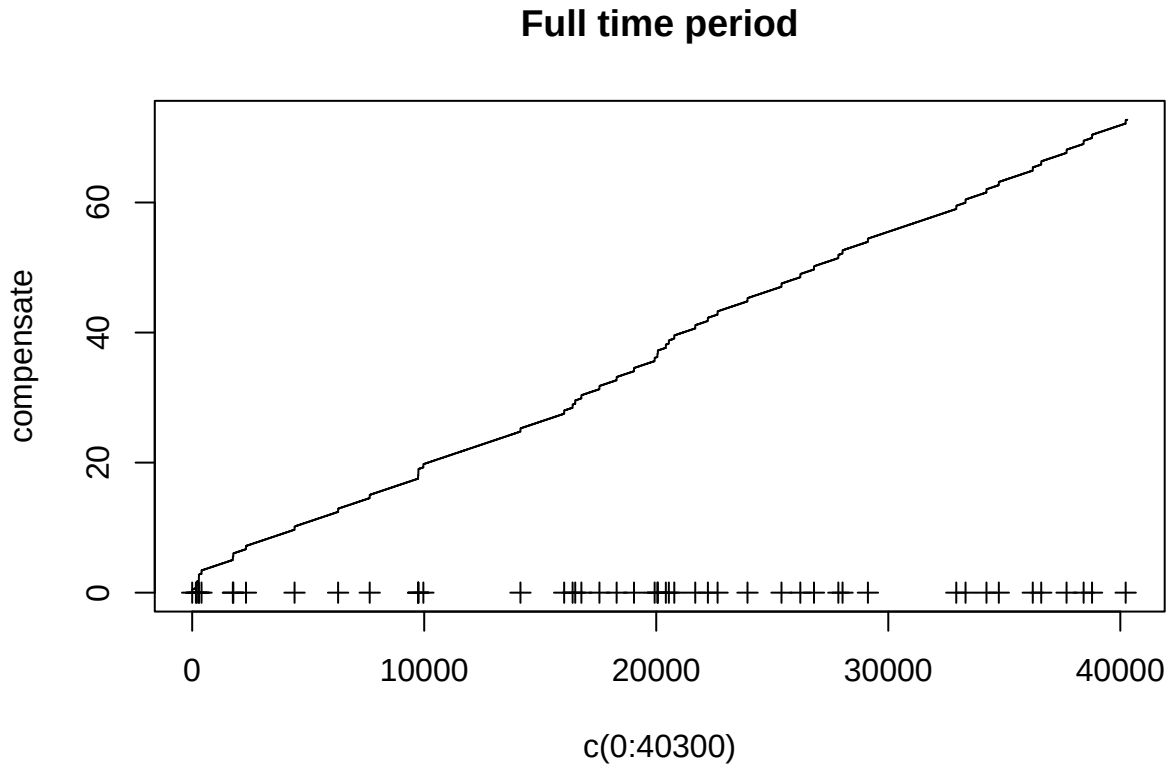
```
plot(resid)
abline(0, 1, col="red", lty="dashed")
```



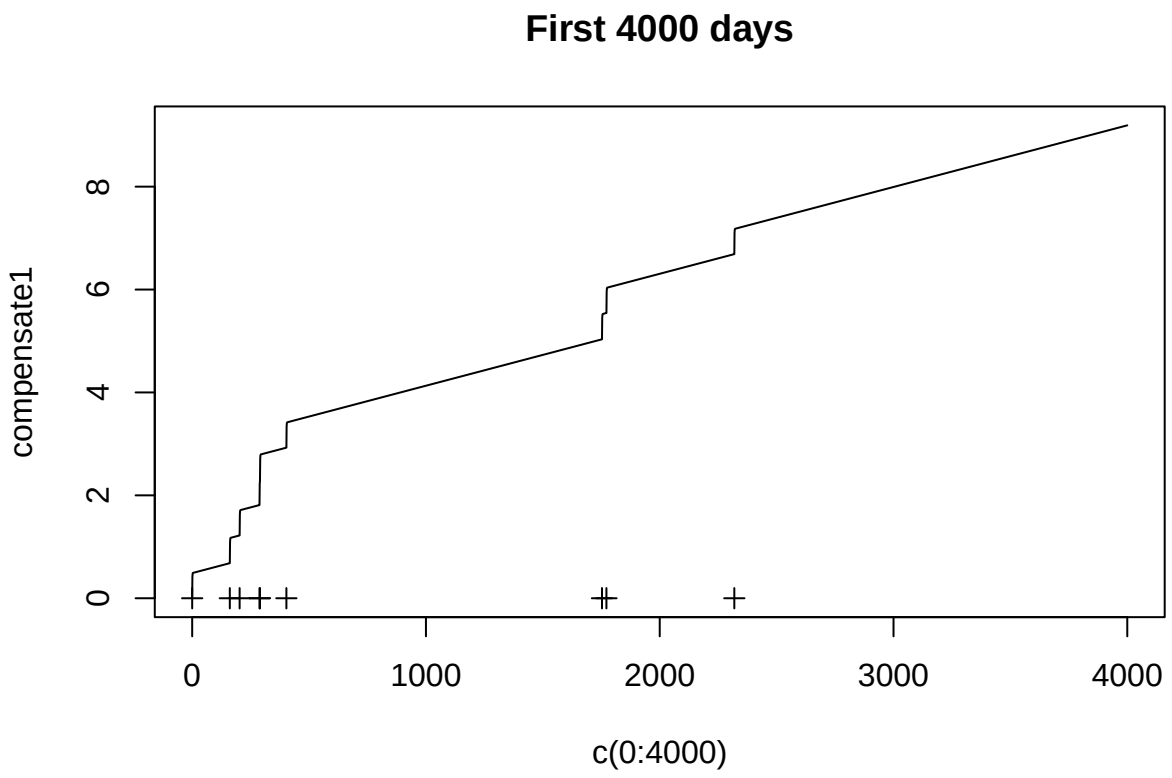
Compensator

From the help: the compensator (integrated intensity) of a Hawkes process... kind of cool, can see how the events (world records) up the intensity. Event times are the plus symbols plotted below the intensity function line.

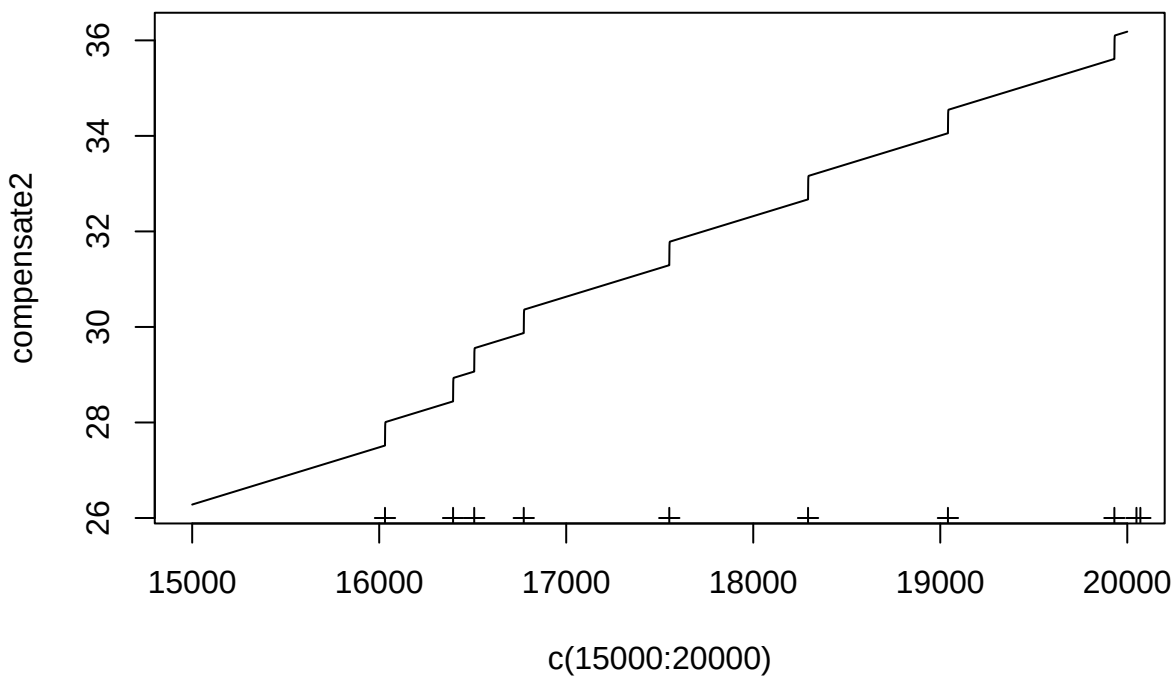
```
compensate<-compensator(daysfromstart, t=0:40300, fun = optMarathon$par[1], repr = optMarathon$par[2],
                  family = "exp", rate = optMarathon$par[3])
plot(c(0:40300),compensate,main="Full time period", type = "l")
points(daysfromstart,rep(0,length(daysfromstart)), pch=3)
```



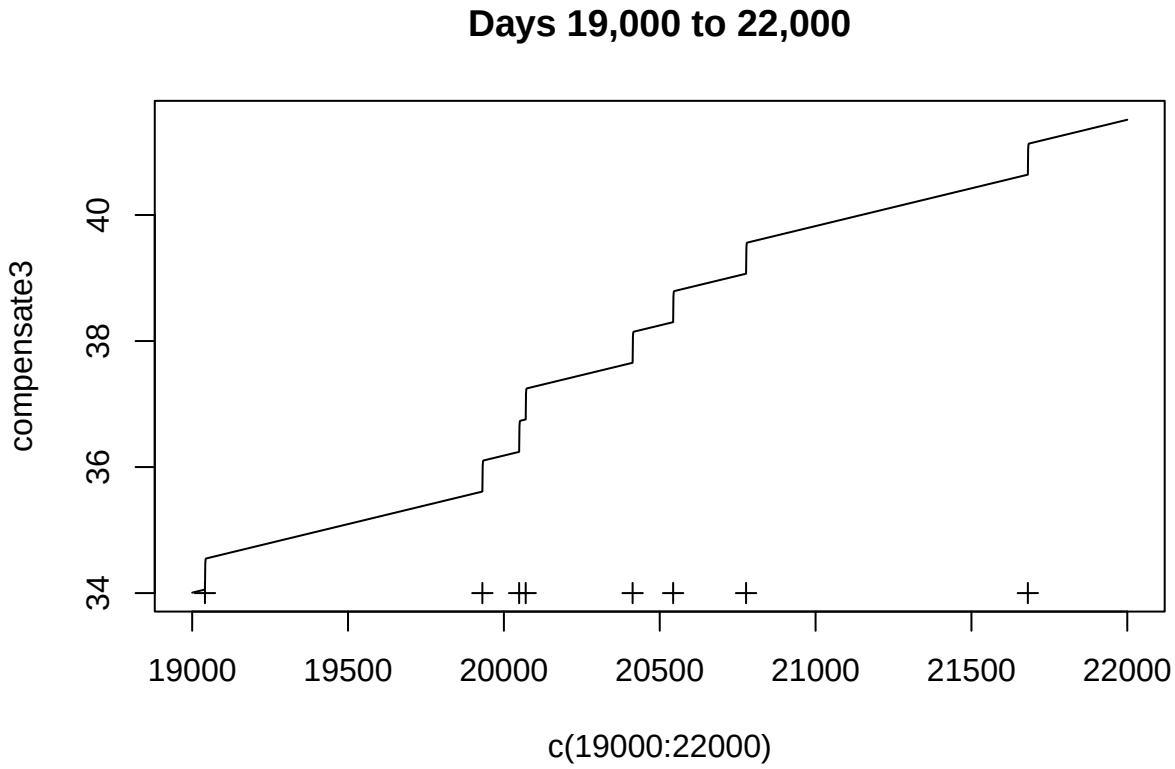

```
compensate1<-compensator(daysfromstart, t=0:4000, fun = optMarathon$par[1], repr = optMarathon$par[2],
                  family = "exp", rate = optMarathon$par[3])
plot(c(0:4000),compensate1,main="First 4000 days", type = "l")
points(daysfromstart,rep(0,length(daysfromstart)), pch=3)
```




```
compensate2<-compensator(daysfromstart, t=15000:20000, fun = optMarathon$par[1], repr = optMarathon$par
                  family = "exp", rate = optMarathon$par[3])
plot(c(15000:20000),compensate2,main="Days 15,000 to 20,000", type = "l")
points(daysfromstart,rep(26,length(daysfromstart)), pch=3)
```



```
compensate3<-compensator(daysfromstart, t=19000:22000, fun = optMarathon$par[1], repr = optMarathon$par
                  family = "exp", rate = optMarathon$par[3])
plot(c(19000:22000),compensate3,main="Days 19,000 to 22,000", type = "l")
points(daysfromstart,rep(34,length(daysfromstart)), pch=3)
```

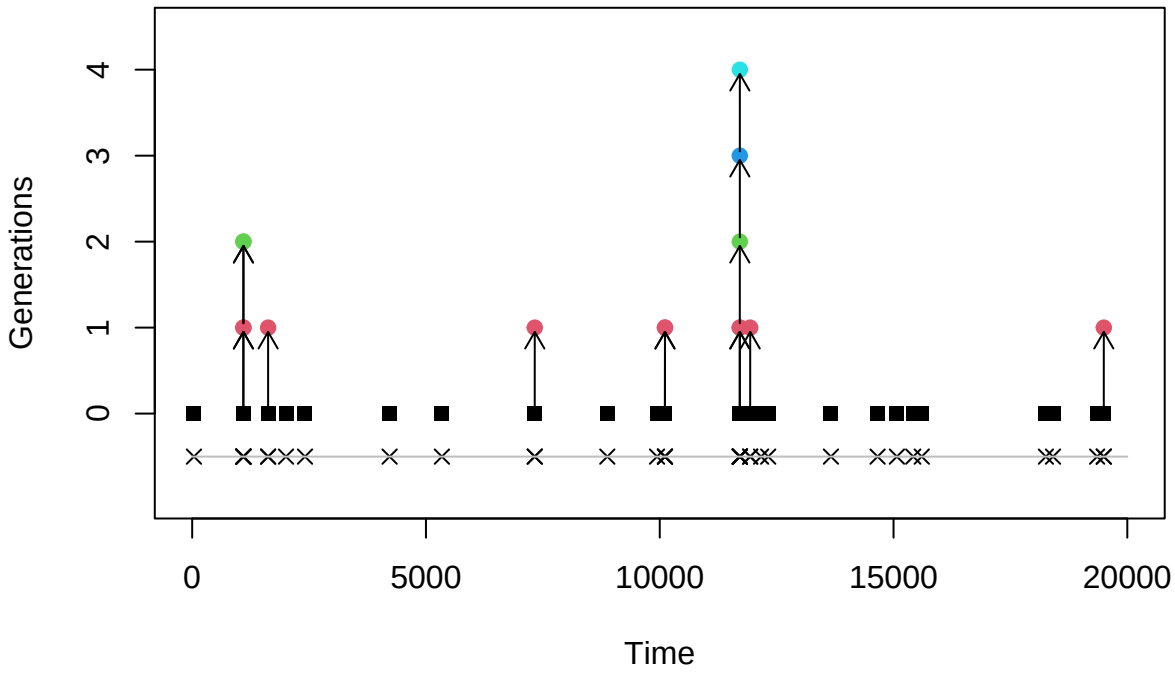


Simulation using Hawkes model with MLE parameter estimates from data

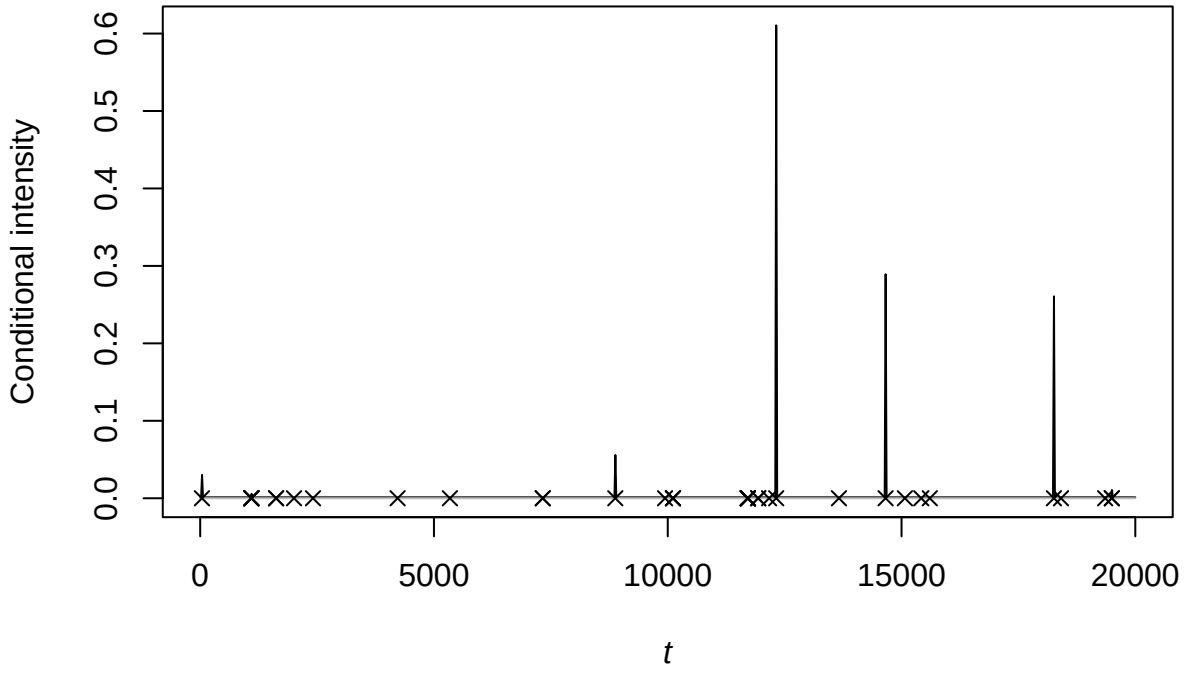
Simulation for the same number of days used in the estimation. Last plot is the residuals - totally unclear to me what these are. . . I guess of the simulated data vs the exact values from the model? Generally seem to follow the y=x line but some runs don't.

The second plot is the intensity - not easy to see, I think maybe a scale issue (not sure why it appears 0 with a few spikes, and to me those spikes don't correspond to plot one events).

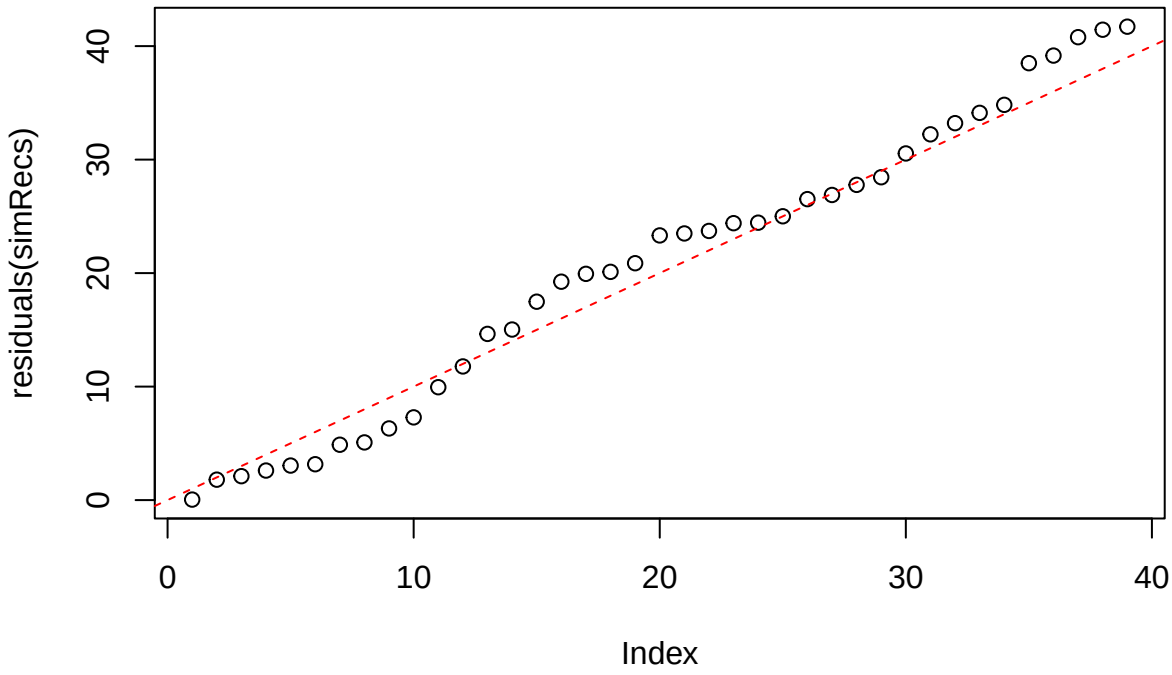
```
set.seed(96)
simRecs <- hawkes(20000, fun = optMarathon$par[1], repr = optMarathon$par[2],
                  family = "exp", rate = optMarathon$par[3])
plot(simRecs, intensity = FALSE)
```




```
plot(simRecs, intensity = TRUE)
```




```
plot(residuals(simRecs))
abline(0, 1, col="red", lty="dashed")
```



test