Gen. Math. Notes, Vol. 21, No. 2, April 2014, pp. 104-113
ISSN 2219-7184; Copyright © ICSRS Publication, 2014
www.i-csrs.org
Available free online at http://www.geman.in

Monte Carlo Simulation for Interpreting Improved Time Points of a Given Point Process Using Conditional Intensity Function

Behrouz Fathi-Vajargah[1] and Hassan Khoshkar-Foshtomi[2]

[1,2]Department of Statistics, University of Guilan, Iran
[1]E-mail: fathi@guilan.ac.ir
[2]E-mail: hassan.khoshkar@yahoo.com

(Received: 23-1-14 / Accepted: 4-3-14)

Abstract

In this paper, we first bring an introductory of a point process and focus on some of its important properties and concepts. Then, we interpret the improved time points of a given process based on conditional intensity function. Finally, we present an efficient algorithm and prove its performance by performing successful results.

Keywords: *Conditional intensity function, Point process, Time points.*

1 Introduction

The basic concepts of general point process, Poisson point process and non-homogeneous Poisson process with basic related definitions and theorems have been introduced in [2, 3, 5].

Definition 1.1 (Conditional Intensity Function): *The conditional intensity function is a convenient and intuitive way of specifying how the present depends on the past in an evolutionary point process. Consider the conditional density f^* and its corresponding cumulative distribution function F^*. Then the conditional intensity function (or Hazard function) is defined by*

$$\lambda^*(t) = \frac{f^*(t)}{1 - F^*(t)}. \tag{1.1}$$

The conditional intensity function can be interpreted heuristically in the following way: consider an infinitesimal interval around t, say dt, then

$$\begin{aligned}
\lambda^*(t)dt &= \frac{f^*(t)dt}{1 - F^*(t)} \\
&= \frac{P(\text{point in } dt | H_t)}{P(\text{point not before } t | H_t)} \\
&= \frac{P(\text{point in } dt, \text{ point not before } t | H_t)}{P(\text{point not before } t | H_t)} \\
&= P(\text{point in } dt | \text{point not before } t, H_t) \\
&= P(\text{point in } dt | H_t) \\
&= E(N(dt) | H_t). \qquad (1.2)
\end{aligned}$$

Here $N(A)$ denotes the number of points falling in an interval, and the last equality follows from the assumption that no points coincide, so that there is either zero or one point in an infinitesimal interval. In other words, the conditional intensity function specifies the mean number of events in a region conditional on the past.

Example 1.2 (Poisson Process): The (nonhomogeneous) Poisson process is among other things characterized by the number of points in disjoint sets being independent. The conditional intensity function inherits this independence. The Poisson process is quite simply the point process where the conditional intensity function is independent of the past, i.e. the conditional intensity function is equal to the intensity function of the Poisson process, $\lambda^*(t) = \lambda(t)$.

2 Simulation

Simulation turns out to be fairly easy when the conditional intensity function is specified. The conditional intensity function leads to different approaches for simulating a point process.

2.1 Why Simulating a Point Process?

Simulations of point processes are useful for many things: What does a point pattern typically look like? Simulating a point process a couple of times for a given model and a given set of parameters will provide valuable information on what a typical point pattern looks. Is it clustered or regular? Is it non homogeneous or homogeneous? Does it look anything remotely like the data you are going to spend the next week fitting the model to?

Prediction: Given an observed past, what does the future hold? The specification of the conditional intensity function means that it is easy to include the already observed past, and then simulate the future.

Model Checking: Prediction can also be used for model checking if we only use the data in the first half of the observation interval to fit a model, and then simulate predictions of the second half to see if this corresponds to the second half of the observed data. Or we can use all of the data, and compare with simulations of the whole data set.

Summary Statistics: Many quantities can be calculated explicitly from the conditional intensity function, such as the probability of getting no events in the next month or the mean time to the next event. However, particularly complicated summary statistics may not be available on closed form, but can instead be approximated by simulation. For example, the mean number of events in a given time interval may not be available on closed form for a complicated model, but we can then approximate it by the average number of points in a number of simulations.

2.2 Simulation Time Points (Method)

In this section, we broach the topic of simulation assessment. In modeling, the existence of a logically consistent simulation algorithm for some process is tantamount to a constructive proof that the process exists. Furthermore, simulation methods have become a key component in evaluating the numerical characteristics of a model, in checking both qualitative and quantitative features of the model, and in the centrally important task of model-based prediction. A brief survey of the principal approach to point process simulation and of the theoretical principle on which this approach is based therefore seemed to us an important complement to the rest of the text.

This section provides a brief introduction to simulation method for evolutionary model; that is, for model retaining a time-like dimension that then dictates the probability structure through the conditional intensity function. Simulation method can be developed also for spatial point patterns, but considerable conceptual simplicity results from the ability to order the evolution of the process

in 'time'. The growth in importance of Markov chain Monte Carlo method for simulating spatial processes is a tacit acknowledgement of the fact that such method introduces an artificial time dimension even into problems where no such dimension is originally present.

The most important theoretical result is a construction, originating in Kerstan (1964) and refined and extended in Br´emaud and Massouli´e (1996). There we transformed a point process with general conditional intensity to a Poisson process; here we convert a Poisson process back into a process with general conditional intensity. For this purpose, we use an auxiliary coordinate in the state space, so we consider a unit-intensity Poisson process, $\widetilde{N}$ say, on the product space $\chi = \mathbb{R} \times \mathbb{R}_+$. The realizations of $\widetilde{N}$ consist of pairs $(\mathrm{x}_j, \mathrm{y}_j)$. Also, let H_t denote the σ-algebra of events defined on a simple point process over the interval $[0,t)$ and H the history $\{\mathrm{H}_t\}$. The critical assumption below is that λ^* is H-adapted.

Theorem 2.1 *Let* $\widetilde{N}$, H *be defined as above, let* $\lambda^*(t)$ *be a nonnegative, left-continuous,* H*-adapted process, and define the point process* N *on* $\mathbb{R}$ *by*

$$N(dt) = \widetilde{N}(dt \times (0, \lambda^*(t)]). \tag{2.1}$$

Then N *has* H*-conditional intensity* $\lambda^*(t)$.

Proof: Arguing heuristically, it is enough to note that

$$E[N(dt) \mid \mathrm{H}_{t-}] = E[\widetilde{N}(dt \times (0, \lambda^*(t-)]) \mid \mathrm{H}_{t-}] = \lambda^*(t)dt. \qquad \square \tag{2.2}$$

There is no requirement in this proposition that the conditional intensity be a.s. uniformly bounded as was required in the original Shedler-Lewis algorithm. When such a bound exists, it leads to straightforward versions of the thinning algorithm, as in algorithm 2.3 below.

The result can be further extended in various ways, for example to situations where more general histories are permitted or where the initial process is not Poisson but has a conditional intensity function that almost surely bounds that of the process to be simulated; see [2, 4].

Example 2.2 (Standard Renewal Process on $[0,\infty]$**):** We suppose the process starts with an event at $t = 0$. Let $h(u)$ denote the hazard function for the lifetime distribution of intervals between successive points, so that (see [10]) the conditional intensity function has the form

$$\lambda^*(t) = h(t - t_{N(t)}) \qquad (t \geq 0), \tag{2.3}$$

Where $t_{N(t)}$ is the time of occurrence of the last event before time t. However, $\lambda^*(t)$ should be defined on the history of $\widetilde{N}$ rather than on N. To this end, we first define the sequence of points t_i in terms of $\widetilde{N}$. With $t_0 = 0$, define sequentially

$$t_{n+1} = \min\{x_i : x_i > t_n \text{ and } y_i < h(x_i - t_n)\} \qquad (n = 0, 1, ...) \tag{2.4}$$

and then define $\lambda^*(t)$ as above. Notice that the right-hand side of this expression is F_t-measurable and the whole process is F-adapted.

Thinning algorithm generally follow much the same lines as in theorem 2.1 and the example above. The main difficulty arises from the range of y_i being unbounded, which provides a flexibility that is difficult to match in practice. The original Shedler-Lewis algorithm (Lewis and Shedler, 1976; see [1, 6]) was for a non-homogeneous Poisson process in a time interval where the intensity is bounded above by some constant, M say. Then, the auxiliary dimension can be taken as the bounded interval $(0, M)$ rather than the whole of $\mathbb{R}_+$, or equivalently the y_i could be considered i.i.d. uniformly distributed random variables on the interval $(0, M)$. Equivalently again, the time intensity could be increased from unity to M and the y_i taken as i.i.d. uniform on $(0,1)$, which leads to the basic form of the thinning algorithm outlined in the algorithm 2.3 below.

In discussing the simulation algorithm 2.3 below, it is convenient to introduce the term list-history to stand for the actual record of times, or times and marks, of events observed or simulated up until the current time t. We shall denote such a list-history by H, or H_t if it is important to record the current time in the notation. Thus, a list-history H is just a vector of times $\{t_1, ..., t_{N(t)}\}$ or a matrix of times and marks $\{(t_1, \kappa_1), ..., (t_{N(t)}, \kappa_{N(t)})\}$. We shall denote the operation of adding a newly observed or generated term to the list-history by $H \to H \cup t_j$ or $H \to H \cup (t_j, \kappa_j)$. In the discussion of conditioning relations such as occur in the conditional intensity, the list-history H_t bears to the σ-algebra H_t a relationship similar to that between an observed value x of a random variable X and the random variable X itself.

The algorithm requires an extension of theorem 2.1 to the situation where the process may depend on an initial history H_0; we omit detail but note the following. Such a history will be reflected in the list-history by a set of times or times and marks of events observed prior to the beginning of the simulation. This is an important feature when we come to prediction algorithms and wish to start the simulation at the 'present', taking into account the real observations that have

been observed up until that time. It is also important in the simulation of stationary processes, for which the simulation may be allowed to run for some initial period $(-B, 0)$ before simulation proper begins. The purpose is to allow the effects of any transients from the initial conditions to become negligible. Finding the optimal length of such a preliminary 'burn-in' period is an important question in its own right. Its solution depends on the rate at which the given process converges toward equilibrium from the initial state, but in general this is a delicate question that is affected by the choice of initial state as well as decay parameters characteristic of the process as a whole.

Suppose, then, that the process to be simulated is specified through its conditional intensity $\lambda^*(t)$, that there exists a finite bound M such that

$$\lambda^*(t) \le M \text{ for all possible past histories,} \tag{2.5}$$

and that the process is to be simulated over a finite interval $[0, A)$ given some initial list-history H_0.

Algorithm 2.3 (Thinning Algorithm for Processes with Bounded Conditional Intensity):

1) Simulate $x_1, ..., x_i$ according to a Poisson process with rate M (for example, by simulating successive interval lengths as i.i.d. exponential variables with mean $\frac{1}{M}$), stopping as soon as $x_i > A$.
2) Simulate $y_1, ..., y_i$ as a set of i.i.d. uniform $(0,1)$ random variables.
3) Set $k = 1$, $j = 1$.
4) If $x_k > A$, terminate. Otherwise, evaluate $\lambda^*(x_k) = \lambda(x_k \mid H_{x_k})$.
5) If $y_k \le \frac{\lambda^*(x_k)}{M}$, set $t_j = x_k$, update H to $H \cup t_j$, and advance j to $j + 1$.
6) Advance k to $k + 1$ and return to step 4.
7) The output consists of the list $\{j;\ t_1, ..., t_j\}$.

This algorithm is relatively simple to describe. In the more elaborate versions that appear shortly, it is convenient to include a termination condition (or conditions), of which steps 1 and 4 above are simple. In general, we may need some limit on the number of points to be generated that lies outside the raison d'etreof the algorithm.

3 Practical Example

Let us we want to make a hybrid format of Poisson point process and non-homogeneous Poisson process based on the above introduced algorithm in an unit square with intensity $\lambda = 1$ and we will generate an arbitrary intensity function for investigating on time points. In the second part of this example we bring its scatter plots where sketched for different values of λ .

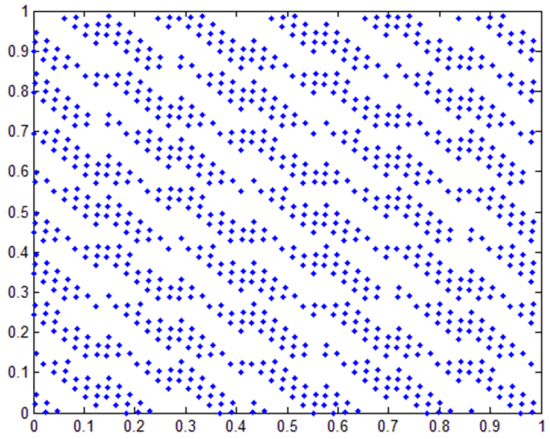

Fig 1: Two dimensional scatter plots of time points directly generated by conditional intensity function using rand function

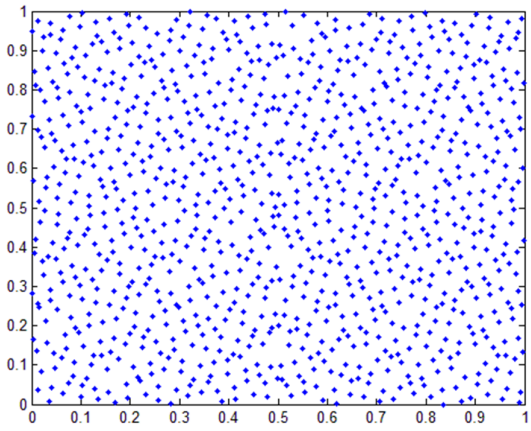

Fig 2: Two dimensional scatter plots of time points where directly generated by conditional intensity function using modified rand library function

The efficiencies of algorithm 2.3 are presented in Fig1. We conclude that the time points for conditional intensity function have not uniform distribution on the desired area of the unit square, this effect on convergence of time points. That is,

based on the conditional intensity function of time points the convergence will happen longer than unconditional one. In this case, if we employ the partitioned rand function (improved uniform random number generator), as it has been shown in Fig2, we have more uniform distributed random number generated in unit square which this effect on speed up convergence too, in shorter time.

Also, we should note that in improved algorithm, not only we use partitioned rand function but also we consider step length $\frac{1}{10}$. Since the conditional desired intensity function has complex format and its elimination does not have negative effect its computations we easily ignored it!

Now, we investigate on second part of this example. Here with regards to the algorithm and using partitioned rand function, we use different values of λ.

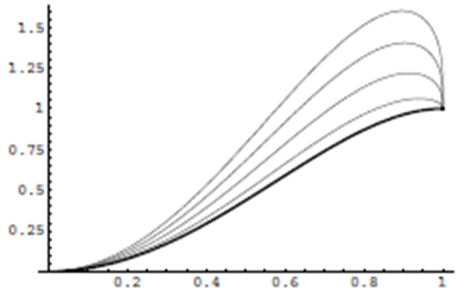

Fig 3: Conditional intensity as a function of t for $\lambda = 1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}$

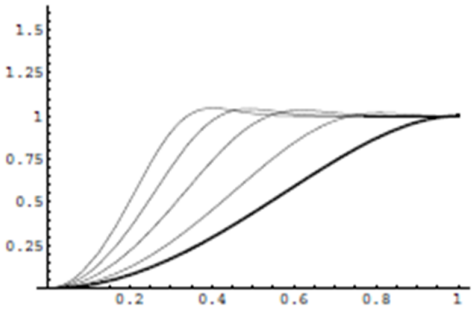

Fig 4: Conditional intensity as a function of t for $\lambda = 1, 4, 9, 16, 25$

As we can see in Fig3, 4 for evaluated time points of algorithm (horizontal axis) are considered for intensity function this example (vertical axis) using different λ .

4 Conclusion

In this paper we introduce an efficient algorithm for simulating time points of a general point process. We also set improved rand function (partitioned rand) instead of original one in Matlab software, and then run it to get results. This makes more uniformity in unit square and will increase the speed of convergence.

References

[1] T.M. Brown and M. Nair, A simple proof of the multivariate random time change theorem for point processes, *J. Appl. Probab*, 25(1988), 210-214.

[2] D.J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes (Volume I: Elementary Theory and Methods)*, Springer, New York, (2003).

[3] D.J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes (Volume II: General Theory and Structure)*, Springer, New York, (2008).

[4] D.J. Daley, On a class of renewal functions, *Proc. Cambridge Philos. Soc*, 61(1965), 519-526.

[5] D.J. Daley and D. Vere-Jones, A summary of the theory of point processes, *In Lewis*, 8(1972), 299-383.

[6] R.B. Davies, Testing the hypothesis that a point process is Poisson, *Adv. Appl. Probab*, 9(1977), 724-746.

[7] J. Møller and J.G. Rasmussen, Perfect simulation of Hawkes processes, *Adv. In Appl. Probab*, 37(3) (2005), 629-646.

[8] J. Møller and R.P. Waagepetersen, *Statistical Inference and Simulation for Spatial Point Processes*, Chapman & Hall, Boca Raton, Florida, (2004).

[9] Y. Ogata, Statistical models for earthquake occurrences and residual analysis for point processes, *Journal of the American Statistical Association*, 83(401) (1988), 9-27.

[10] B. Picinbono, Time intervals and counting in point processes, *IEEE Trans. Inform. Theory*, 50(2004), 1336-1340.

[11] I. Pinelis, Monotonicity properties of the relative error of a Pade approximation for Mills' ratio, *Journal of Inequalities in Pure and Applied Mathematics*, 3(20) (2002), 1-8.

[12] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, Cambridge, U.K, (1991).

[13] J.G. Rasmussen, *Temporal Point Processes the Conditional Intensity Function*, Department of Mathematics, Aalborg University, Denmark, (2011).

[14] S.H.M. Ross, *Introduction to Probability Models*, Academic Press, San Diego, (2003).
[15] R.Y. Rubenstein, *Simulation and the Monte Carlo Method*, Wiley, New York, (1981).