IIE Transactions (2004) **36**, 1155–1160

ISSN: 0740-817X print / 1545-8830 online
DOI: 10.1080/07408170490507693

TECHNICAL NOTE

Nonparametric estimation and variate generation for a nonhomogeneous Poisson process from event count data

LAWRENCE M. LEEMIS

Department of Mathematics, The College of William & Mary, Williamsburg, VA 23187-8795, USA
E-mail: leemis@math.wm.edu

Received November 2001 and accepted March 2004

Given a finite time horizon that has been partitioned into subintervals over which event counts have been accumulated for multiple realizations of a population NonHomogeneous Poisson Process (NHPP), this paper develops point and confidence-interval estimators for the cumulative intensity (or mean value) function of the population process evaluated at each subinterval endpoint. As the number of realizations tends to infinity, each point estimator is strongly consistent and the corresponding confidence-interval estimator is asymptotically exact. If the NHPP has a piecewise constant intensity (rate) function, then the proposed point and confidence-interval estimators for the cumulative intensity function are valid over the entire time horizon and not just at the subinterval endpoints; and in this case algorithms are presented for generating event times from the estimated NHPP. Event count data from a call center illustrate the point and interval estimators.

1. Introduction

A NonHomogeneous Poisson Process (NHPP) is often appropriate for the modeling of a series of events that occur over time in a nonstationary fashion. Two common application areas are the modeling of arrivals to a waiting line (queueing theory) and the failure times of a repairable system (reliability theory). NHPPs have been used to model event occurrences in a variety of applications, ranging from arrivals to a computer electronics store (White, 1999) to product repair times (Nelson, 2003). This article considers the nonparametric estimation of the cumulative intensity function for a NHPP from a data set of k realizations of event counts over predefined subintervals. Variate generation algorithms are given for the generation of all of the events in a particular cycle and for the generation of the next event, which is appropriate for a general-purpose, discrete-event simulation language.

A NHPP generalizes a Homogeneous Poisson Process (HPP). Events occur at a constant rate of λ events per unit time in a HPP. A NHPP is governed by an intensity function, $\lambda(t)$, which may vary with time. The cumulative intensity function:

$$\Lambda(t) = \int_0^t \lambda(\tau)\mathrm{d}\tau, \quad t > 0,$$

gives the expected number of events by time t. The number of events that occur on the interval $(a, b]$ is Poisson with mean $\int_a^b \lambda(t)\mathrm{d}t$. Events can be generated for use in discrete-event simulation as $\Lambda^{-1}(E_1), \Lambda^{-1}(E_2), \ldots$, where $E_1, E_2, \ldots$ are the event times in a unit HPP (Cinlar, 1975).

Section 2 describes the estimation technique. Section 3 gives two variate generation algorithms. Section 4 illustrates the use of the technique on a data set. Section 5 gives extensions to the work and some conclusions.

2. Estimation

Oftentimes event-time data are given as counts that occur in disjoint subintervals as opposed to the event times themselves (Law and Kelton, 2000, p. 391). This section describes point and interval estimates for the intensity and cumulative intensity functions based on this count data.

Assume that there are k independent realizations of a NHPP with cumulative intensity function $\Lambda(t)$ collected on the interval $(0, S]$, where S is a real, fixed constant. This interval could be part of a day (e.g., arrivals to a lunchwagon between 10:00 am and 2:30 pm) or one cycle in a process (e.g., arrivals to an emergency room during 24 hours). The estimation techniques and variate generation algorithms developed here require that the origin be the time when data are first collected and that the time units are consistent with the data. If the interval of interest is from 10:00 am to 2:30 pm, for example, then the interval is $(0, 4.5]$ if

the data are in hours or (0, 270] if the data are in minutes. The time interval (0, S] is partitioned into m subintervals:

$$(a_0, a_1], (a_1, a_2], \ldots, (a_{m-1}, a_m],$$

where $a_0 = 0$ and $a_m = S$. The subintervals do not necessarily have equal widths. Let $n_1, n_2, \ldots, n_m$ be the total number of observed events in the subintervals over the k realizations.

Assume that the population NHPP has an intensity function $\lambda(t)$ that is piecewise constant on each subinterval of the partition $(a_0, a_1], (a_1, a_2], \ldots, (a_{m-1}, a_m]$. Since the average intensity function on the interval $(a_{i-1}, a_i]$ is the rate per unit time of the events that occur on that interval, the maximum likelihood estimator is the average number of events that occurred on the interval, normalized for the length of the interval:

$$\hat{\lambda}(t) = \frac{n_i}{k(a_i - a_{i-1})}, \quad a_{i-1} < t \le a_i,$$

for $i = 1, 2, \ldots, m$. The associated cumulative intensity function estimate is a continuous, piecewise-linear function on (0, S]:

$$\hat{\Lambda}(t) = \left(\sum_{j=1}^{i-1} \frac{n_j}{k} \right) + \frac{n_i(t - a_{i-1})}{k(a_i - a_{i-1})}, \quad a_{i-1} < t \le a_i,$$

for $i = 1, 2, \ldots, m$. (If there are no events observed on interval i, i.e., $n_i = 0$, then the intensity function estimate is zero on interval i and the cumulative intensity function estimate is constant on interval i. In the variate generation algorithms to be described in the next section, no events will be generated for such an interval. This is useful for modeling an interval where no events should occur, e.g., lunchbreaks.) This estimator passes through the points $(a_i, \sum_{j=1}^{i}(n_j/k))$ for $i = 1, 2, \ldots, m$. Asymptotic properties of this estimator in the case of equal-width subintervals are considered by Henderson (2003).

The population intensity function $\lambda(t)$ will not be piecewise constant over each subinterval $(a_{i-1}, a_i]$ in the arbitrary partition of (0, S] in most applications. As shown in Leemis (1991), $\hat{\Lambda}(t)$ can only be consistent in this general case at the endpoints of the subintervals as $k \to \infty$, i.e., $\lim_{k\to\infty} \hat{\Lambda}(a_i) = \Lambda(a_i)$ with probability one for $i = 0, 1, \ldots, m$. This means that the usual confidence-interval estimate for the cumulative intensity function:

$$\hat{\Lambda}(t) - z_{\alpha/2}\sqrt{\frac{\hat{\Lambda}(t)}{k}} < \Lambda(t) < \hat{\Lambda}(t) + z_{\alpha/2}\sqrt{\frac{\hat{\Lambda}(t)}{k}},$$

for $0 < t \le S$, where $z_{\alpha/2}$ is the $1 - \alpha/2$ fractile of the standard normal distribution, is only asymptotically exact at the endpoints of the subintervals, i.e.:

$$\lim_{k\to\infty} \Pr\left(\hat{\Lambda}(a_i) - z_{\alpha/2}\sqrt{\frac{\hat{\Lambda}(a_i)}{k}} < \Lambda(a_i) < \hat{\Lambda}(a_i) + z_{\alpha/2}\sqrt{\frac{\hat{\Lambda}(a_i)}{k}} \right) = 1 - \alpha,$$

for $i = 0, 1, \ldots, m$. For this reason, the displays of the point and confidence-interval estimates for the cumulative intensity functions given in the examples in Section 4 are given only at the subinterval endpoints, and connected by dashed lines (to indicate the appropriate point estimator if $\lambda(t)$ was piecewise constant over the subintervals).

A piecewise-constant interval estimate for $\lambda(t)$ for one realization is found using the technique described in Rigdon and Basu (2000, p. 114), and an alternative confidence interval based on the chi-square distribution for multiple realizations is given in Casella and Berger (2002, pp. 434–435).

3. Variate generation

A realization of a Poisson process for modeling in a discrete-event simulation can be generated by inversion. Let $T_1, T_2, \ldots$ denote the event times for the NHPP with cumulative intensity function $\hat{\Lambda}(t)$ generated on (0, S]. Furthermore, let $E_1, E_2, \ldots$ be the event times of a unit *homogeneous* Poisson process. Using the algorithm below,

```
max ← Σ_{i=1}^{m} n_i/k                          (upper bound for HPP)
i ← 1                                            (initialize interval counter)
j ← 1                                            (initialize variate counter)
cumint ← n_i/k                                   (initialize cumulative intensity)
generate U_j ~ U(0, 1)                           (generate first random number)
E_j ← −log(1 − U_j)                              (generate first HPP event time)
while (E_j ≤ max)                                (while more events to generate)
   while (E_j > cumint)                          (while in wrong interval)
      i ← i + 1                                  (increment interval counter)
      cumint ← cumint + n_i/k                    (increment cumulative intensity)
   endwhile
   T_j ← a_i − (cumint − E_j)k(a_i − a_{i−1})/n_i (generate jth NHPP event time)
   j ← j + 1                                     (increment variate counter)
   generate U_j ~ U(0, 1)                        (generate jth random number)
   E_j ← E_{j−1} − log(1 − U_j)                  (generate next HPP event time)
endwhile
return (T_1, T_2, ..., T_{j−1})                 (return NHPP event times)
```

the NHPP event times can be generated from the inputs $a_0, a_1, a_2, \ldots, a_m; n_1, n_2, \ldots, n_m$, and k. The logarithm function used here is the natural logarithm (log base e). Comments are given at the right in brackets.

The algorithm is valid when the population NHPP with intensity function $\lambda(t)$ is piecewise constant. Any departure from this assumption results in an approximate estimator $\hat{\Lambda}(t)$ and associated approximate variate generation algorithm between the subinterval endpoints.

The algorithm given above is appropriate when $\sum_{i=1}^{m} n_i/k$ is of modest size so that there is adequate memory available to store the event times prior to the execution of the discrete-event simulation model. This approach is not appropriate for a general-purpose simulation language since the number of events to be generated for the NHPP is not known in advance and could require excessive memory. The second algorithm given below uses the next-event approach (Banks *et al*., 2001), which schedules the next event when the current event is being processed. The algorithm has the same static inputs ($a_0, a_1, \ldots, a_m$; $n_1, n_2, \ldots, n_m$, and k) as the first algorithm, except that this algorithm returns the next event time given that the current event occurs at the dynamic input time $T \in (0, S]$. The algorithm returns the time of the next NHPP event $\hat{\Lambda}^{-1}(\hat{\Lambda}(T) + E)$, where $E \sim$ expon(1), or -1 if the ending time S is encountered. The algorithm is illustrated in Fig. 1.

The algorithm returns -1 if there are no further events to be generated, or the next event time. The variable *cumint.now* contains the cumulative intensity function associated with the time of the current NHPP event T, i.e., $\hat{\Lambda}(T)$. The variable *cumint.new* contains the cumulative intensity function associated with the time of the next HPP event, $\hat{\Lambda}(T) + E$, where E is a unit exponential random variable. At the end of the execution of this algorithm, the variable *cumint* contains the cumulative intensity function value at the right interval endpoint associated with the returned event time.

A more sophisticated implementation of this "next-event" algorithm would store *max*, *j*, *cumint*, and *cumint.new* between the generation of events, effectively eliminating the first seven lines of the algorithm. The procedure could begin with the generation of the $U(0, 1)$, which would save substantial execution time for large m.

4. Example

Table 1 contains counts by hour and day of the week of 9512 arriving phone calls associated with a call center over the period 8/12/01 to 8/18/01 which is open from 8 am to 9 pm daily. Time is measured in hours, which is aligned to the origin so that $S = 13$ and the time interval of observation is (0, 13]. No callers obtained busy signals. No infomercials, which create a spike in call volume, were run during this period. The organization occasionally runs infomercials at various times during the day. These infomercials create a spike in the volume to the call center just after the infomercial is aired. No infomercials were run during this 7-day period so the appropriate population cumulative intensity function $\Lambda(t)$ is a baseline process that excludes the additional volume generated by infomercials. A separate analysis is necessary to evaluate the magnitude and duration of the additional call volume generated by an infomercial. Although there are approximately 5% of the calls which are abandoned, we treat this input as the incoming stream, realizing that the estimate obtained will be a bit pessimistic (i.e., an overestimate). The totals in the bottom row of the table indicate that call volume is not homogeneous throughout

---

$max \leftarrow \sum_{j=1}^{m} n_j/k$ (maximum cumulative intensity)
$j \leftarrow 1$ (initialize interval index)
while $(T > a_j)$ (while wrong interval)
  $j \leftarrow j + 1$ (find interval index)
endwhile
$cumint.now \leftarrow \sum_{i=1}^{j-1} n_i/k + n_j(T - a_{j-1})/(k(a_j - a_{j-1}))$ (calculate $\hat{\Lambda}(T)$)
$cumint \leftarrow \sum_{i=1}^{j} n_i/k$ (initialize cumulative intensity interval bound)
generate $U \sim U(0, 1)$ (generate a random number U)
$E \leftarrow -\log(1 - U)$ (generate a unit exponential random variate)
$cumint.new \leftarrow cumint.now + E$ (calculate $\hat{\Lambda}(T) + E$)
if $(cumint.new \leq max)$ then (if there are more events to generate)
  while $(cumint.new > cumint)$ (while *cumint* is in the wrong interval)
    $j \leftarrow j + 1$ (increment interval counter)
    $cumint \leftarrow cumint + n_j/k$ (increment cumulative intensity)
  endwhile
  return$(a_j - (cumint - cumint.new)k(a_j - a_{j-1})/n_j)$ (NHPP event time $\hat{\Lambda}^{-1}(\hat{\Lambda}(T) + E)$)
else
  return(-1) (-1 to indicate no more NHPP events to generate)
endif

Table 1. Arrival counts to a call center

	Time													
Day	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*Total*
Sunday	42	47	79	101	83	74	79	105	88	94	84	51	68	995
Monday	63	144	133	163	140	104	137	145	163	150	113	91	79	1625
Tuesday	75	129	148	144	134	128	132	135	150	119	102	66	58	1520
Wednesday	76	115	97	127	98	120	130	130	124	97	92	51	77	1334
Thursday	57	108	184	134	131	109	129	135	118	108	94	77	69	1453
Friday	72	134	139	129	123	114	106	156	145	123	102	67	68	1478
Saturday	56	91	93	96	77	83	86	109	127	95	81	68	45	1107
Total	441	768	873	894	786	732	799	915	915	786	668	471	464	9512

the day and that the intensity function has two modes: one between 10 am and 12 pm and a second between 3 pm and 5 pm. It also appears that the volume varies by day of the week.

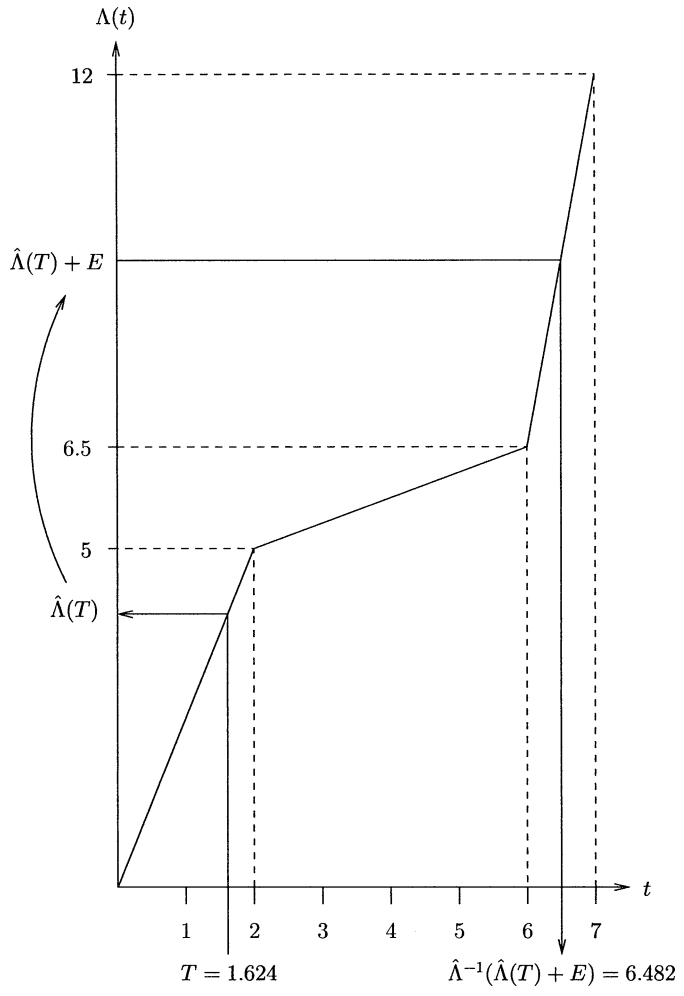

Fig. 1. Geometry associated with the next-event approach for variate generation when $m = 3$, $a_0 = 0$, $a_1 = 2$, $a_2 = 6$, $a_3 = 7$; $n_1 = 10$, $n_2 = 3$, $n_3 = 11$, $k = 2$, $T = 1.624$, $\hat{\Lambda}(T) = 4.060$, $E = 5.091$, $\hat{\Lambda}(T) + E = 9.151$, $\hat{\Lambda}^{-1}(\hat{\Lambda}(T) + E) = 6.482$. The dashed lines indicate the interval boundaries where the cumulative intensity function estimate changes slope.

The lightest day in terms of call volume was Sunday. The heaviest day in terms of call volume was Monday. Figure 2 contains a plot of the estimated cumulative intensity function for Sunday and Monday. The 99% confidence intervals for the cumulative intensity function indicate that there is a statistically significant difference between the call volume on the two days.

Assuming that an exhaustive pairwise comparison of the seven days under consideration indicates that the weekends can be clumped together into a single model (with $k = 2$) and that the weekdays can be clumped together into a single model (with $k = 5$), Figure 3 contains the two cumulative intensity function estimates and the associated 99% confidence intervals. Not surprisingly, the confidence interval limits for the weekdays are narrower than those for the weekends due to the larger sample size. These cumulative intensity function estimates can be used as input models in a discrete-event simulation for decision-making purposes (e.g., staffing).

5. Extensions and conclusions

In most practical settings, the number of realizations collected k, will be a fixed constant from interval to interval. In some instances, however, such as subintervals containing an unusually low number of events or subintervals of particular interest, k may not be fixed. With only minor modifications (e.g., replacing k by k_i or k_j), the estimators and variate generation algorithms can be modified to accommodate the generalization.

This article has shown that it is straightforward to estimate and simulate NHPPs from count data collected over predefined subintervals. One drawback with count data is that the boundaries on the cells are arbitrary. If the cells are too narrow, then sampling variability can cause unreliable estimates. If the cells are too wide, then it is possible to miss a trend. Therefore, event times, rather than counts, are preferred. In these cases, nonparametric estimates suggested, for example, by Leemis (1991) and Arkin and Leemis (2000), or parametric estimates suggested, for example, by Kuhl *et al.* (1997) or Rigdon and Basu (2000), are appropriate.

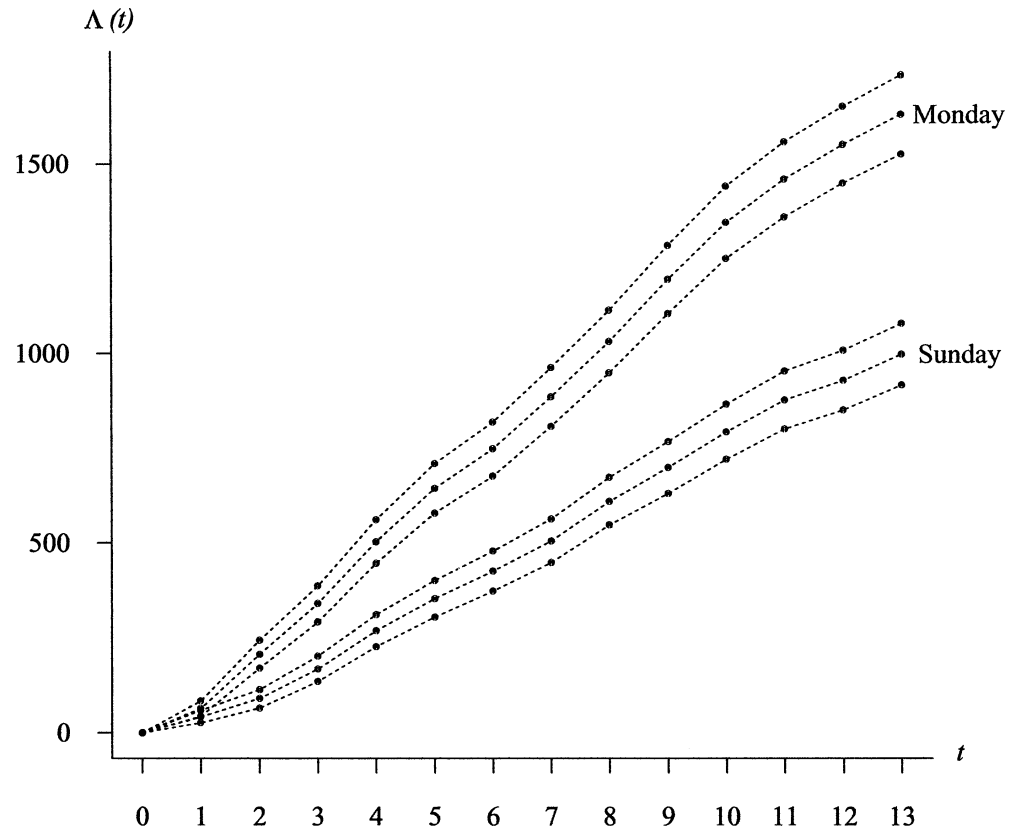

Fig. 2. Cumulative intensity function estimate for Sunday and Monday.

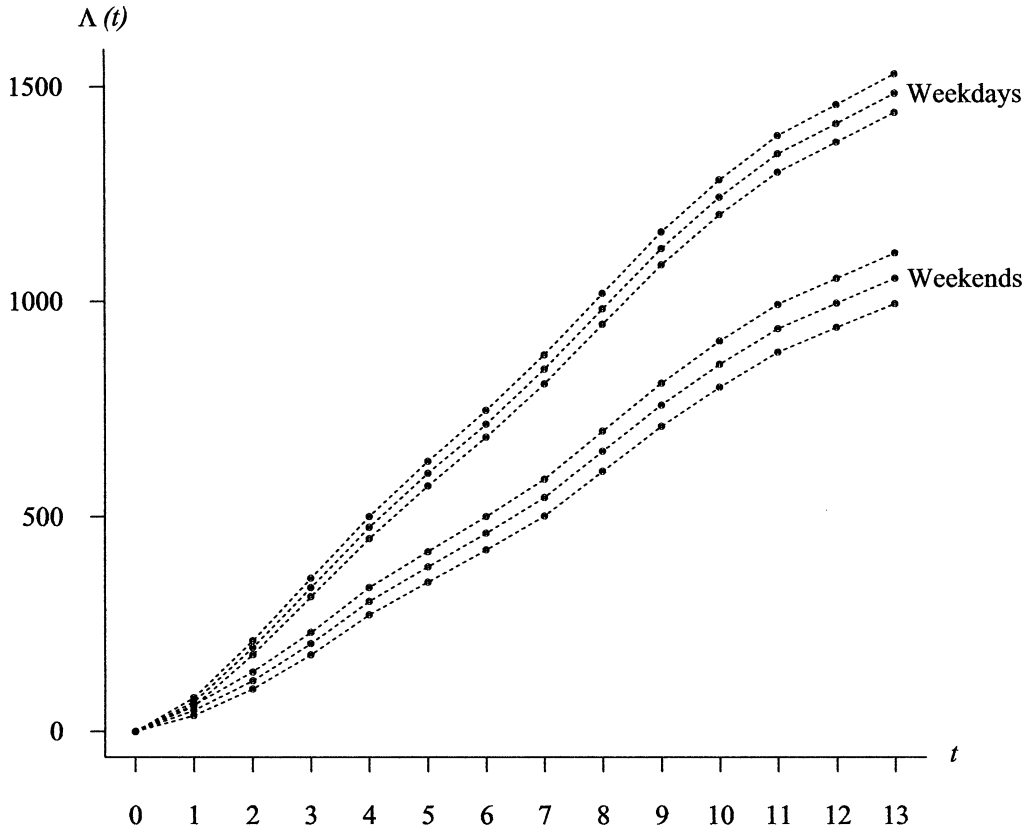

Fig. 3. Cumulative intensity function estimate for weekdays and weekends.

Acknowledgement

The author acknowledges helpful suggestions from four anonymous referees.

References

Arkin, B.L. and Leemis, L.M. (2000) Nonparametric estimation of the cumulative intensity function for a nonhomogeneous Poisson process from overlapping realizations. *Management Science*, **46**(7), 989–998.

Banks, J., Carson, J.S., Nelson, B.L. and Nicol, D.M. (2001) *Discrete-Event System Simulation*, 3rd edn., Prentice–Hall, Englewood Cliffs, NJ.

Casella, G. and Berger, R.L. (2002) *Statistical Inference*, 2nd ed., Wadsworth & Brooks/Cole, Pacific Grove, CA.

Cinlar, E. (1975) *Introduction to Stochastic Processes*, Prentice–Hall, Englewood Cliffs, NJ.

Henderson, S.G. (2003) Estimation for nonhomogeneous Poisson processes from aggregated data. *Operations Research Letters*, **31**, 375–382.

Kuhl, M.E., Wilson, J.R. and Johnson, M.A. (1997) Estimating and simulating Poisson processes having trends and multiple periodicities. *IIE Transactions*, **29**, 201–211.

Law, A.M. and Kelton, W.D. (2000) *Simulation Modeling and Analysis*, 3rd edn., McGraw–Hill, New York, NY.

Leemis, L.M. (1991) Nonparametric estimation of the intensity function for a nonhomogeneous Poisson process. *Management Science*, **37**(7), 886–900.

Nelson, W.B. (2003) *Recurrent Events Data Analysis for Product Repairs, Disease Recurrences, and Other Applications*, ASA/SIAM, Philadelphia, PA.

Rigdon, S.E. and Basu, A.P. (2000) *Statistical Methods for the Reliability of Repairable Systems*, Wiley, New York, NY.

White, K.P. (1999) Simulating a nonstationary Poisson process using bivariate thinning: the case of "typical weekday" arrivals at a consumer electronics store, in *1999 Winter Simulation Conference Proceedings*, Farrington, P., Nembhard, P.A., Sturrock, H.B., and Evans, G.W., (eds.), Piscataway, NJ: Institute of Electrical and Electronic Engineers, pp. 458–461.

Biography

Lawrence Leemis is a professor in the Mathematics Department at the College of William and Mary. He received his B.S. and M.S. degrees in Mathematics and his Ph.D. degree in Industrial Engineering from Purdue University. He has also taught courses at Purdue University, The University of Oklahoma and Baylor University. His consulting, short course and research contract work includes contracts with AT&T, NASA/Langley Research Center, Delco Electronics, Department of Defense, Air Logistic Command, ICASE, Komag, Federal Aviation Administration, Tinker Air Force Base, Magnetic Peripherals, Woodmizer, Yorktown Naval Weapons Station, and Argonne National Laboratory. His research and teaching interests are in reliability and simulation. He has published journal articles in the areas of reliability, simulation, and quality control. His textbooks are *Reliability: Probabilistic Models and Statistical Methods* (1995), published by Prentice-Hall and *Simulation: A First Course* (with Steve Park, forthcoming), published by Prentice-Hall. He has served as an Associate Editor for the *IEEE Transactions on Reliability*, Book Review Editor for the *Journal of Quality Technology*, and on the Editorial Board for the *IIE Transactions*.

Contributed by Reliability and Engineering Department