MPCS 58020 2015 Homework 2

Assignment date: July 3, 2015

Due date: July 15, 2015

Revised July 7, 2015, to omit problems 2.9 and 2.10

Revised July 12, 2015, to correct page numers for problems 2.1,2.2,2.3

Solve the following problems. Send solutions (programs, results, and analytical results) to the TA (rahaman@cs.uchicago.edu) with the subject line "[MPCS 58020 2015] HW2".

These problems cover Chapters 4 and 5 of Ross, *Simulation* and Lecture 2.

1. ***Random permutations*** *can be generated using a variation of the discrete inverse transform method, as described in Ross, Example 4b.*

   A deck of 50 cards are labeled with the numbers 1,2,...,50. The cards are shuffled and then turned over one card at a time. Say that a "hit" occurs whenever card labeled when the ith card to be turned over is labeled with the number i. Let the random variable X be the total number of hits after all cards have been turned over.

   (a) Without a simulation, derive the expected value and variance of X.

   (b) Compose and run a simulation to estimate the expected value and variance of X.

2. ***Exponential random variables*** *have the PDF* $f(x) = \lambda e^{-\lambda x}$ *and the CDF* $F(x) = 1 - e^{-\lambda x}\,dx$ *over the interval* $(0, \infty)$.

   In many applications, the exponential distribution can describe a continuous quantity that may take on any positive value, but for which larger values are increasingly unlikely. For example, the time it takes for a radioactive particle to to decay is an exponential random variable. Ross, Example 5b, describes how to use the inverse transform method to simulate exponential random variables.

   A casualty insurance company has 1000 policyholders, each of whom will independently present a claim in the next month with probability 0.05.

Assuming that the amounts of the claims made are independent exponential random variables with mean \$800, use a simulation to estimate the probability that the sum of these claims will exceed \$50,000.

3. *A **Poisson process** is a series of events in which the interarrival times (i.e, the times between events) are exponential random variables. Poisson processes are used in queuing theory for a huge variety of applications. A homogeneous Poisson process is one in which the interarrival times are independent and identically-distributed. Ross, Section 5.4, describes how to generate a homogeneous Poisson process.*

   Buses arrive at a sporting event according to a Poisson process with rate 5 per hour. Each bus is equally likely to contain either 20,21,...,40 fans, with the numbers in the different buses being independent. Write an algorithm to simulate the arrival of fans to the event by time $t = 1$.

4. In class, we used the rejection method to generate random numbers with PDF:

$$f(x) = 3x^2 \quad 0 < x1$$

   by generating numbers from:

$$g(x) = 1 \quad 0 < x < 1$$

   . We can make a more efficient implementation by using a different PDF in place of $g(x)$.

   (a) Use the rejection method to generate random numbers from $f(x)$ by generating random numbers from:

$$h(x) = 2x \quad 0 < x < 1$$

   .

   (b) Compare the efficiency (average time and iterations per random number accepted; and associated variances for both) of the two implementations (with $g(x)$ and $h(x)$).

   (c) Derive a performance model for the expected number of iterations per random number accepted (and the associated variance) for both implementations. How does your performance model compare to your results?

These problems cover Chapter 2.2 and 2.3 of Shumway and Stoffer, *Time Series Analysis and its Applications* (the 3rd ed, with 603 pages and a yellow cover) and Lecture 3.

- Problem 2.1, page 78. Use the either of the included data files, `jj_tables.txt` or `jj_series.txt`. The series contains the same values as the table, listed in chronological order by (year,quarter).
- Problem 2.2, page 78. Use the included data files `cmort_table.txt` (cardiovascular mortality), `tempr_table.txt` (temperature), and `part_table.txt` (particulate pollution); or the corresponding `*_series.txt` files. The series contain the same values as the respective tables, listed in chronological order by (year, week).
- Problem 2.3, page 78.
- Problem 2.8, parts a,b,c,d, page 80. Use the included data file `varve.txt`. Values are listed in chronolical order by year.