Poisson Process: Confirmation of the Mean and Distribution via Simulation

Jonathan Morgan

Department of Mathematics

2018

January

1 Verify the Poisson Distribution

A Poisson process of intensity, or rate, $\lambda > 0$ is an integer valued stochastic process $\{X(t); t \geq 0\}$ for which

(i) for any time points $t_0 = 0 < t_1 < \cdots < t_n$, the process increments

$$X(t_1) - X(t_0), X(t_2) - X(t_1), \ldots, X(t_n) - X(t_{n-1})$$

are independent random variables;

(ii) for $s \geq 0$ and $t > 0$, the random variable $X(s+t) - X(s)$ has the Poisson distribution

$$Pr\{X(s+t) - X(s) = k\} = \frac{(\lambda t)^k e^{-\lambda t}}{k!}| \text{ for } k = 0, 1, \ldots;$$

(iii) $X(0) = 0$.

The interval for each simulation is $[0, T]$, where $T \in \mathbb{N}$ and is a user input parameter. For simplicity, $T = 1$ was hardcoded to verify that the simulation has a Poisson distribution.

Let s be a uniform random variable on $[0, 1]$, let X be an integer valued stochastic process $\{X(t); t \geq 0\}$, and let h be a forward-step along the interval $[0, T] = [0, 1]$, such that $h = 0.01$. Hence, the process increments $X(t_0 + h) - X(t_0), X(t_0 + 2h) - X(t_0 + h), \ldots, X(t_0 + nh) - X(t_0 + (n-1)h)$, where $n = 100$. If s is sampled on each increment, and if $X(t)$ is given the property

$$X(t) = \begin{cases} 1 & s < h \\ 0 & s \geq h \end{cases},$$

then, by construction, property (i) is satisfied. By extending the properties of X and restricting $X(0) = 0$, then (iii) is also satisfied. Thus, the only property in which the simulation must verify is (ii).

Each trajectory in the simulation has the structure:

```
X = 0;
for (i = 0; i < n; i++) {
    sample s;
    if (s < h) {
        X = X + 1;
    }
}
```

Hence, we see that the variable X (in the program above) is the total amount of jumps along a trajectory. To verify (*ii*), it is convenient to run the program above through multiple iterations. Each trajectory's jump total, X, must then be collected and stored for further analysis. To accomplish this, the program above is modified such that:

```
jump_totals = [ ];
for (j = 0; j < N; j++) {
    X = 0;
    for (i = 0; i < n; i++) {
        sample s;
        if (s < h) {
            X = X + 1;
        }
    }
    jump_totals[j] = X;
}
```

Hence, we see that *jump_totals* is an array of N trajectory jump totals. It takes the form

$$jump\_totals = [X_1, X_2, \ldots, X_N],$$

where X_i is the jump total for trajectory i.

With the construction of the jump totals array, it is now possible to check the distribution of the values for X_i. To accomplish this, the described simulation above was ran with the parameters

$$h = 0.01$$
$$n = 100$$
$$N = 1000.$$
$$\lambda = 1$$

In order to check the distribution of a specific jump total value, k, it is necessary to loop through the list of jump totals and count the number of occurrences where $X_i = k$. Then, divide the number of occurrences by the total number of simulations to get $Pr_{exp}\{X(1) - X(0) = k\}$, the experimental probability that $X = k$. The program to check the distribution takes the form:

$k_{count} = 0;$

for $(j = 0; j < 1000; j++)$ {

  if $(jump\_totals[j] = k)$ {

    $k_{count} = k_{count} + 1;$

  }

}

$$Pr_{exp}\{X(1) - X(0) = k\} = \tfrac{k_{count}}{1000};$$

$$Pr_{th}\{X(1) - X(0) = k\} = \frac{(\lambda t)^k e^{-\lambda t}}{k!} = \frac{(1)^k e^{-(1)}}{k!}$$

$$error = |Pr_{exp} - Pr_{th}|$$

In the above, k is a user input variable. Hence, the program is called with an integer value. The program returns three values, $Pr_{exp}, Pr_{th}, error$.