MSMS 206 : Practical 11

Ananda Biswas

May 8, 2025

Question : Write a program to generate random sample from Poisson Process with parameter λ.

Also write programs to generate $X(t)$ if

(a) $P(X(t) = k) = \binom{k+\alpha-1}{k} \left(\frac{\beta}{t+\beta}\right)^{\alpha} \left(\frac{t}{t+\beta}\right)^{k}$; $k = 0, 1, 2, \ldots$.

(b) $P(X(t) = k) = \left(\frac{t}{t+\mu}\right)^{k} \left(\frac{\mu}{t+\mu}\right)$; $k = 0, 1, 2, \ldots$.

A realization of Poisson Process

We take $\lambda = 2$ and observe the Poisson Process upto time $T = 5$.

```
T <- 5

lambda <- 2
```

We observe the arrival times upto time T, for that we generate exponentially distributed random numbers with rate λ until total time becomes T. The count of the arrival times form a realization of a Poisson Process with parameter λ.

```
times <- c(0)

i <- 2

sum_times <- 0

while(sum_times < T){

  times[i] <- round(rexp(1, rate = lambda), digits = 2)

  sum_times <- sum(times)

  i <- i + 1
}

occurrence_times <- cumsum(times[-length(times)])

x <- 0:(length(occurrence_times)-1)
```

```r
df1 <- data.frame(States = x, Occurrence_Time = occurrence_times)
```

✎ States and their arrival times are as follows :

```r
df1
```

```
##    States Occurrence_Time
## 1       0            0.00
## 2       1            0.09
## 3       2            0.74
## 4       3            0.83
## 5       4            2.51
## 6       5            2.62
## 7       6            3.36
## 8       7            4.13
## 9       8            4.17
## 10      9            4.39
## 11     10            4.47
## 12     11            4.62
## 13     12            4.85
```

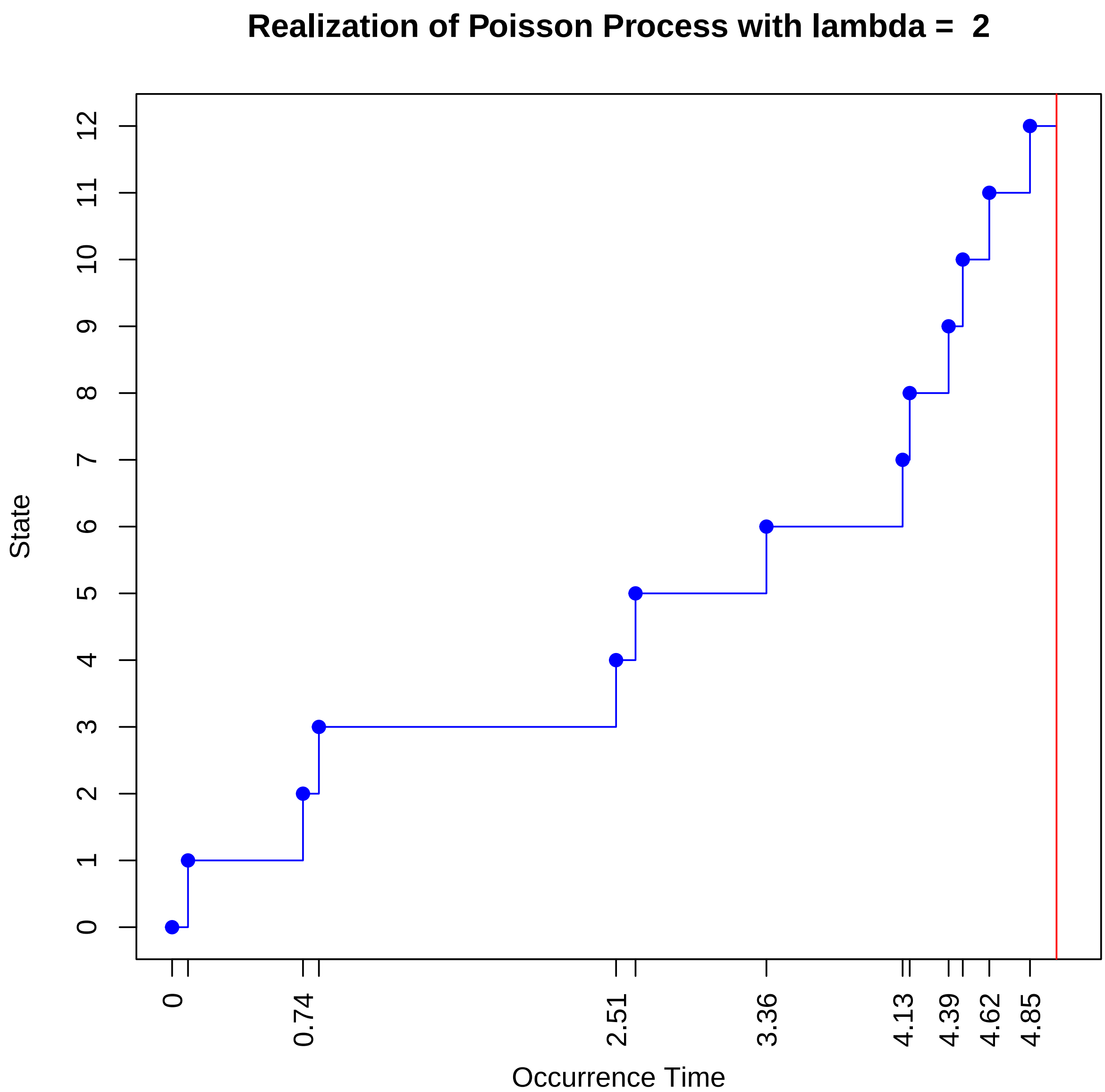
✎ A realization of a Poisson Process with $\lambda = 2$ in $(0, 5]$ is as follows :

$$X(t) = \begin{cases} 0, & \text{if } 0 \leq t < 0 \\ 1, & \text{if } 0 \leq t < 0.09 \\ 2, & \text{if } 0.09 \leq t < 0.74 \\ \vdots & \vdots \\ 11, & \text{if } 4.62 \leq t < 4.85 \\ 12, & \text{if } 4.85 \leq t \leq 5 \end{cases}$$

➔ Visualization

```r
plot(c(x, x[length(x)]) ~ c(occurrence_times, T),
     type = "s",
     col = "blue",
     xaxt = "n",
     yaxt = "n",
     xlim = c(0, T + 0.05),
     xlab = "Occurrence Time",
     ylab = "State",
     main = paste("Realization of Poisson Process with lambda = ", lambda))

axis(1, at = occurrence_times, labels = occurrence_times, las = 2)
axis(2, at = x, labels = x)
points(occurrence_times, x, cex = 1, col = "blue", pch = 19)
abline(v = T, col = "red")
```



➔ $X(t) \sim$ Negative Binomial

$$P(X(t)=k) = \binom{k+\alpha-1}{k}\left(\frac{\beta}{t+\beta}\right)^{\alpha}\left(\frac{t}{t+\beta}\right)^{k};\ \ k = 0, 1, 2, \ldots$$

$i.e.$ $X(t) \sim$ Negative Binomial $\left(\alpha, \frac{\beta}{t+\beta}\right)$.

Previously λ was a fixed parameter. Now λ will be sampled from $\text{Gamma}(\text{shape} = \alpha, \text{rate} = \beta)$. Here we take $\alpha = 2$, $\beta = 1$.

```
alpha <- 2; beta <- 1
```

```
T <- 5

lambda <- rgamma(1, shape = alpha, rate = beta)
```

```
times <- c(0)

i <- 2

sum_times <- 0

while(sum_times < T){

  times[i] <- round(rexp(1, rate = lambda), digits = 2)

  sum_times <- sum(times)

  i <- i + 1
}

occurrence_times <- cumsum(times[-length(times)])

x <- 0:(length(occurrence_times)-1)
```

```
df2 <- data.frame(States = x, Occurrence_Time = occurrence_times)
```

```
df2

##   States Occurrence_Time
## 1      0            0.00
## 2      1            0.26
## 3      2            1.67
```

➔ $X(t) \sim \text{Geometric}$

$$P(X(t)=k) = \left(\frac{t}{t+\mu}\right)^k \left(\frac{\mu}{t+\mu}\right); \quad k = 0,1,2,\ldots \ \textit{i.e.} \ X(t) \sim \text{Geometric}\left(\frac{\mu}{t+\mu}\right).$$

Previously λ was a fixed parameter. Now λ will be sampled from $\text{Gamma}(\text{shape} = 1, \text{rate} = \mu) \Leftrightarrow \text{Exp}(\text{rate} = \mu)$.

Here we take $\mu = 2$.

```
mu <- 2
```

```
T <- 5

lambda <- rgamma(1, shape = 1, rate = mu)
```

```
times <- c(0)

i <- 2

sum_times <- 0

while(sum_times < T){

  times[i] <- round(rexp(1, rate = lambda), digits = 2)

  sum_times <- sum(times)

  i <- i + 1
}

occurrence_times <- cumsum(times[-length(times)])

x <- 0:(length(occurrence_times)-1)
```

```
df3 <- data.frame(States = x, Occurrence_Time = occurrence_times)
```

```
df3

##   States Occurrence_Time
## 1      0            0.00
## 2      1            0.72
## 3      2            3.23
## 4      3            3.50
## 5      4            4.80
```