Exercise set 5

Bjørn Christian Weinbach

September 30, 2020

Clear R environment

```r
rm(list = ls())
```

Exercise 4.3

A nonhomogeneous Poisson process has the mean value function

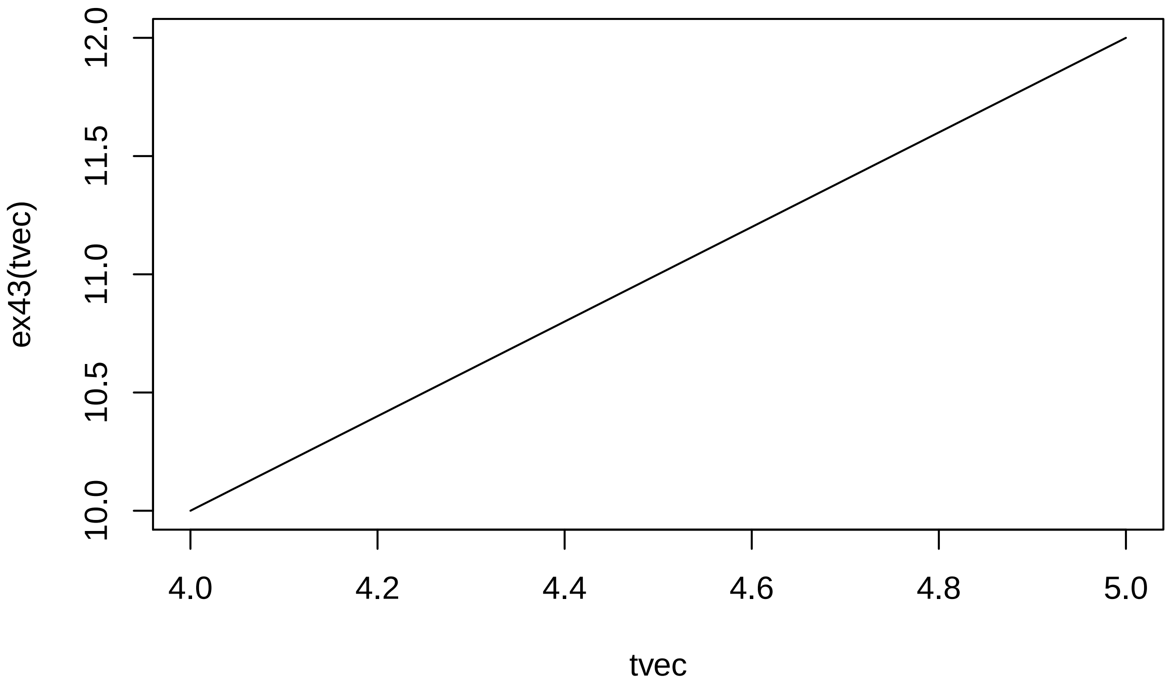
$$m(t) = t^2 + 2t, \qquad t \geq 0.$$

Determine the intensity function $\lambda(t)$ of the process, and write a program to simulate the process on the interval $[4, 5]$. Compute the probability distribution of $N(5) - N(4)$, and compare it to the empirical estimate obtained by replicating the simulation.

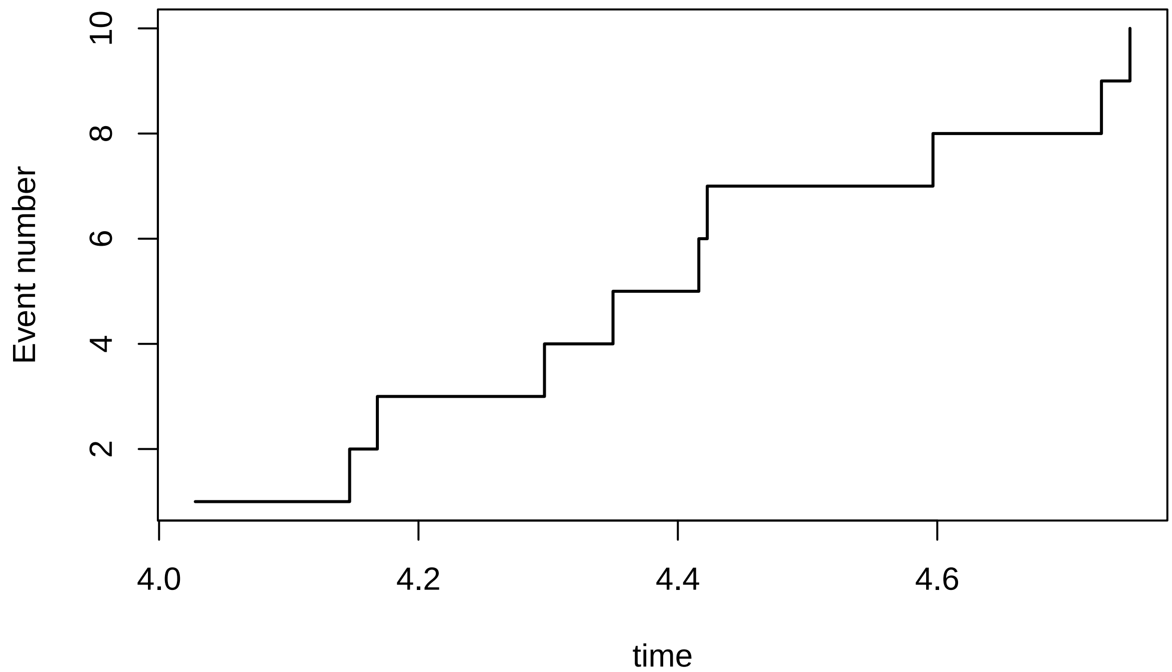
```r
# Function for simulating arrival times for a NHPP between a and b using thinning
simtNHPP <- function(a,b,lambdamax,lambdafunc){
  if(max(lambdafunc(seq(a,b,length.out = 100)))>lambdamax)
    stop("lambdamax is smaller than max of the lambdafunction")
  expectednumber <- (b-a)*lambdamax
  Nsim <- 3*expectednumber
  timesbetween <- rexp(Nsim,lambdamax)
  timesto <- a+cumsum(timesbetween)
  timesto <- timesto[timesto<b]
  Nevents <- length(timesto)
  U <- runif(Nevents)
  timesto <- timesto[U<lambdafunc(timesto)/lambdamax]
  timesto
}

# Specify the intensity function for the traffic example
ex43 <- function(t)
  2*t + 2

tvec <- seq(4, 5,by=0.01)
plot(tvec, ex43(tvec),type="l")
```




```r
# Generate data with the traffic intensity and plot them
NHPPtimes <- simtNHPP(a=4,b=5,lambdamax=12,lambdafunc=ex43)
plot(NHPPtimes,1:length(NHPPtimes),type="s",xlab = "time",
     ylab = "Event number",lwd=1.5)
points(NHPPtimes,rep(0,length(NHPPtimes)),pch=21,bg="red")
```




```
# Rerun the lines above several times
```

Exercise 6.1

Compute a Nonte Carlo estimate of

$$\int_0^{\frac{\pi}{3}} sin(t)dt$$

```
mcint <- function(Nsim, a, b, func) {
  return((b - a)*mean(func(runif(Nsim, a, b))))
}

intfunc <- function(x) {
  return(sin(x))
}

print(paste("Monte Carlo Integration:", mcint(10000, 0, pi/3, intfunc)))
```

```
## [1] "Monte Carlo Integration: 0.501879927471869"
```

```
print(paste("Exact integral:", -cos(pi/3) + cos(0)))
```

```
## [1] "Exact integral: 0.5"
```

Exercise 6.2

Refer to example 6.3. Compute a Monte Carlo estimate of the standard normal cdf, by generating from the uniform(0, x) distribution. Compare estimate with normal cdf function pnorm.

```
# From example 6.3 in Statistical Computing with R
mcstdnormx <- function(Nsim, x) {
  u <- runif(Nsim)
  g <- x * exp( -(u*x)^2 / 2 )
  return((1 - 0) * mean(g) * sqrt(2 / pi))
}

mcstdnormx(10000, 2)
```

```
## [1] 0.9553032
```

```
pnorm(2, 0, 1, lower.tail=TRUE, log.p=FALSE)
```

```
## [1] 0.9772499
```

Confidence interval of $\Phi(2)$

Exercise 6.3

Compute a Monte Carlo estimate $\hat{\theta}$ of

$$\int_0^{0.5} e^{-x} dx$$

by sampling from a Uniform(0, 0.5) and estimate the variance of of $\hat{\theta}$