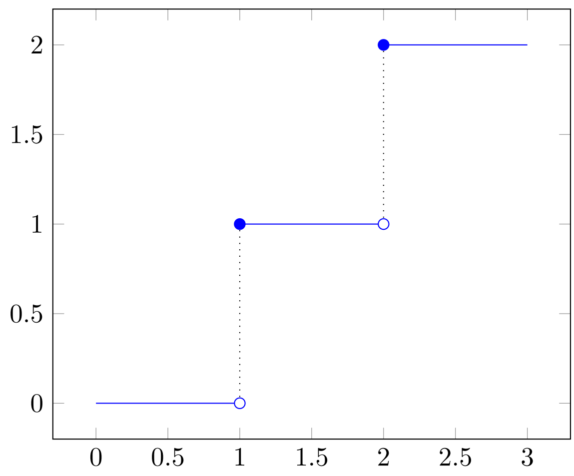
Stochastic Processes

Kevin Chang

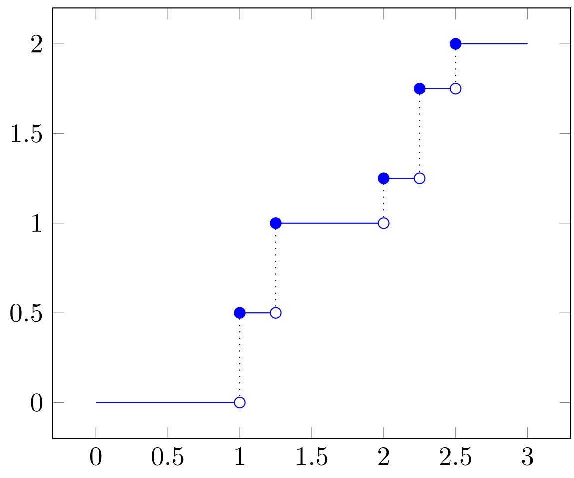
February 18, 2022

1

- Question: For $t \in (0,3)$, plot by hand $\mathbb{E}[N(t)]$ for each of the following renewal processes:
- (a) $X_j = 1$



- (b) $X_j = \begin{cases} 1 & \text{w/ prob } \frac{1}{2} \\ \frac{5}{4} & \text{w/ prob } \frac{1}{2} \end{cases}$



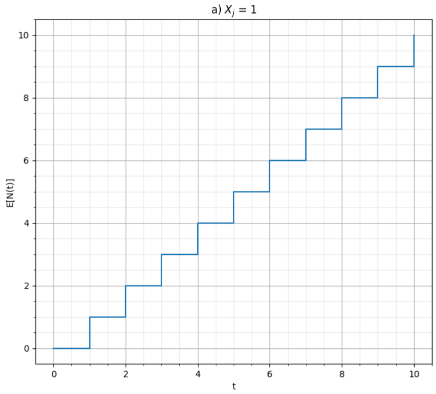
2

- Question: Now write code to estimate $\mathbb{E}[N(t)]$ based on many trials, and plot the estimate versus time for $t \in (0,10)$ for each of the following renewal processes:
- Code

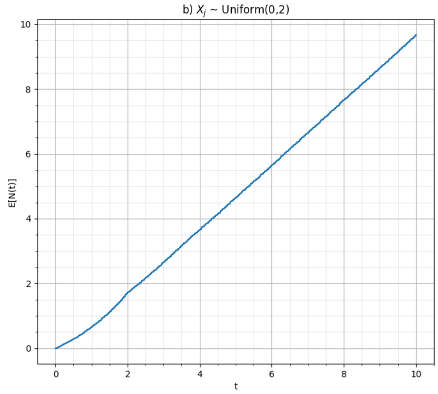
```python
import numpy as np
import matplotlib
import matplotlib.pyplot as plt

tmax = 10
n = int(30 * tmax)
trials = 10000

def myplot(average, title, file_name):
    x = np.linspace(0, tmax, n+1)
    plt.figure(figsize = (8,8))
    plt.title(title)
    plt.xlabel('t')
    plt.ylabel('E[N(t)]')
    plt.minorticks_on()
    plt.grid(True, which='major')
    plt.grid(True, which='minor', color='#999999', linestyle = '-', alpha=0.2)
    plt.plot(x, average, drawstyle='steps-post')
    plt.savefig("{}.png".format(file_name))
    #  plt.show()

def estimateENt(interarrivalTime, tmax, n, trials):
    res = np.zeros((n + 1))
    for trial in range(trials):
        current_num = 0
        current_time = 0
        next_interval = interarrivalTime()
        for i in range(n + 1):
            t = i * tmax / n
            if t >= current_time + next_interval:
                current_num = current_num + 1
                current_time = current_time + next_interval
                next_interval = interarrivalTime()

            res[i] += current_num
    res = res / trials
    return res


myplot(estimateENt(lambda: 1, tmax, n, trials), 'a) $X_j$ = 1', "a")
myplot(estimateENt(lambda: 2*np.random.random(), tmax, n, trials), 'b) $X_j$ ~ Uniform(0,2)', "b")
myplot(estimateENt(lambda: 0.5 + np.random.random(1), tmax, n, trials), 'c) $X_j$ ~ Uniform($\\frac{1}{
2},\\frac{3}{2}$)', "c")
myplot(estimateENt(lambda: np.random.exponential(1), tmax, n, trials), 'd) $X_j$ ~ Exp($\lambda$ = 1)',
 "d")
myplot(estimateENt(lambda: 1 if np.random.random() < 0.75 else np.random.exponential(1), tmax, n, trial
s), r'e) $X_j = 1$ w/ prob $\frac{3}{4}$' "\n" r'~ Exp($\lambda$ = 1) w/ prob $\frac{1}{4}$', "e")
myplot(estimateENt(lambda: 0.5 if np.random.random() < 0.5 else 1.5, tmax, n, trials), r'f) $X_j = \fra
c{1}{2}$ w/ prob $\frac{1}{2}$' "\n" r'= $\frac{3}{2}$ w/ prob $\frac{1}{2}$', "f")
myplot(estimateENt(lambda: 5/6 if np.random.random() < 6/7 else 2, tmax, n, trials), r'g) $X_j = \frac{
5}{6}$ w/ prob $\frac{6}{7}$' "\n" r'2 w/ prob $\frac{1}{7}$', "g")
myplot(estimateENt(lambda: 1/3 if np.random.random() < 0.461263 else np.pi / 2, tmax, n, trials), r'h)
$X_j = \frac{1}{3}$ w/ prob 46.1263%' "\n" r'$\frac{\pi}{2}$ w/ prob 53.8737%', "h")
```

- Simulation Result

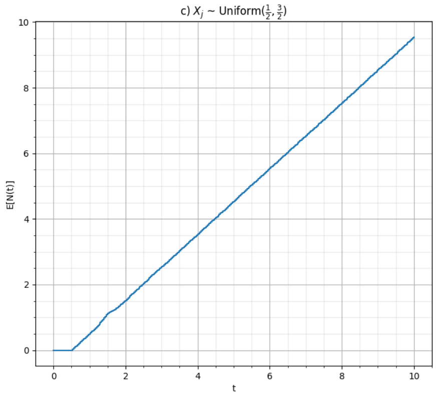
a) $X_j = 1$



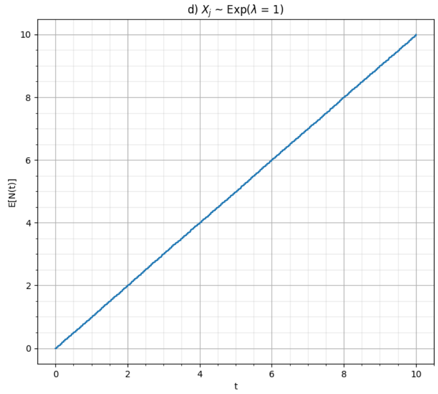
b) $X_j \sim$ Uniform(0,2)



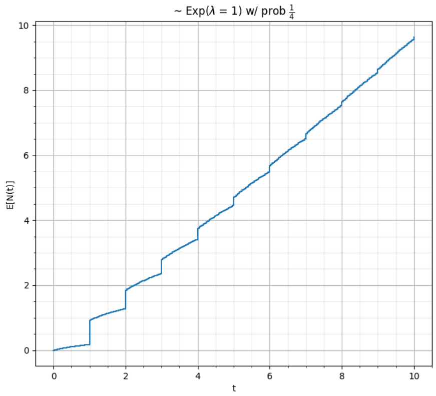
c) $X_j \sim$ Uniform($\frac{1}{2}, \frac{3}{2}$)



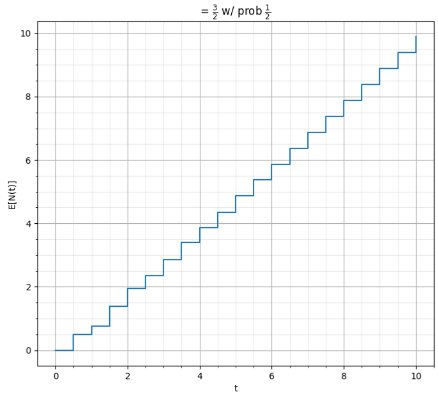
d) $X_j \sim$ Exp(λ = 1)



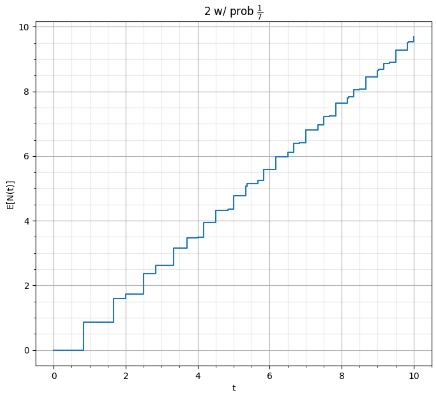
e) $X_j = 1$ w/ prob $\frac{3}{4}$
$\sim$ Exp(λ = 1) w/ prob $\frac{1}{4}$



f) $X_j = \frac{1}{2}$ w/ prob $\frac{1}{2}$
$= \frac{3}{2}$ w/ prob $\frac{1}{2}$



g) $X_j = \frac{5}{6}$ w/ prob $\frac{6}{7}$
2 w/ prob $\frac{1}{7}$



h) $X_j = \frac{1}{3}$ w/ prob 46.1263%
$\frac{\pi}{2}$ w/ prob 53.8737%