ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Μάθημα: Συστήματα Αναμονής

Ονοματεπώνυμο: Ειρήνη Δόντη

Α.Μ: 03119839

$1^{η}$ Ομάδα Ασκήσεων

Αθήνα 2022

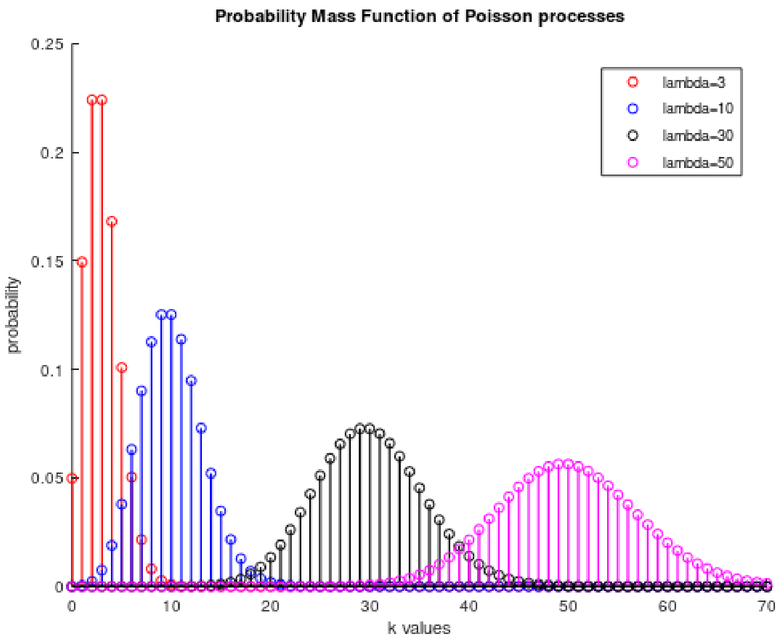
Κατανομή Poisson

Στην άσκηση αυτή, βγάζουμε τα συμπεράσματά μας από το δοθέν πρόγραμμα damo1a.m.

A)

Η συνάρτηση μάζας πιθανότητας μίας διακριτής τυχαίας μεταβλητής X η οποία ακολουθεί κατανομή Poisson είναι: $Pr(X = k) = \frac{e^{-\lambda}\lambda^{k}}{k!}$, όπου λ > 0 η παράμετρος της κατανομής και k=0,1,2,3….

Παρακάτω απεικονίζεται, σε κοινό διάγραμμα, η συνάρτηση μάζας πιθανότητας των κατανομών Poisson με παραμέτρους λ={3,10,30,50}.



Παρατηρούμε ότι, όσο αυξάνεται η παράμετρος λ, τόσο μειώνεται η κορυφή της γραφικής Poisson και αποκτά μεγαλύτερο πλάτος στον άξονα x.Αυτό συμβαίνει, διότι το άθροισμα των πιθανοτήτων της συνάρτησης μάζας πιθανότητας είναι ίση με τη μονάδα. Οπότε, αφού μειώνονται οι κορυφές καθώς αυξάνεται η παράμετρος λ, τότε η κατανομή Poisson πρέπει να αποκτήσει μεγαλύτερο πλάτος στον άξονα x (ανοίγει η διακύμανση της). Τέλος, παρατηρούμε ότι, σε όλες τις περιπτώσεις, η μέση τιμή των κατανομών βρίσκεται στο σημείο λ του άξονα x, το οποίο είναι λογικό καθώς η μέση τιμή της κατανομής Poisson είναι E[x]=λ.

Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

```
k = 0:1:70;
lambda = [3, 10, 30, 50];

for i = 1 : columns(lambda)
  poisson(i, :) = poisspdf(k, lambda(i));
endfor

colors = "rbkm";
figure(1);
hold on;
for i = 1 : columns(lambda)
  stem(k, poisson(i, :), colors(i), "linewidth", 1.2);
endfor
hold off;

title("Probability Mass Function of Poisson processes");
xlabel("k values");
ylabel("probability");
legend("lambda=3", "lambda=10", "lambda=30", "lambda=50");
```

B)

Η μέση τιμή της κατανομής Poisson μίας τυχαίας μεταβλητής X είναι:

$$\mu = E[X] = \sum_{k=0}^{\infty} ke^{-\lambda}\frac{\lambda^k}{k!} = e^{-\lambda}\lambda e^{\lambda} = \lambda$$

Η διακύμανση της κατανομής Poisson μίας τυχαίας μεταβλητής X είναι:

$$\sigma^2 = E[X^2] - \mu^2 = \lambda^2 + \lambda - \lambda^2 = \lambda$$

διότι: $E[X^2] = E[X(X-1)] + E[X] = \lambda^2 + \lambda$ με $E[X] = \lambda$ και

$$E[X(X-1)] = \sum_{k=0}^{\infty} k(k-1)e^{-\lambda}\frac{\lambda^k}{k!} = \lambda^2 \sum_{k=2}^{\infty} e^{-\lambda}\frac{\lambda^{k-2}}{(k-2)!} = e^{-\lambda}\lambda^2 e^{\lambda} = \lambda^2$$

Παρακάτω, απεικονίζεται το command window αφότου εκτελέσουμε το πρόγραμμα:

```
mean value of Poisson with lambda 30 is
mean_value =  30.000
Variance of Poisson with lambda 30 is
variance =  30.000
```

Από το command window, παρατηρούμε ότι η κατανομή Poisson έχει ίδια μέση τιμή με την τιμή της διακύμανσης και ισούται με την παράμετρο λ = 30. Το γεγονός αυτό επιβεβαιώνεται και από τη θεωρία.

Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

```
index = find(lambda == 30);
chosen = poisson(index, :);
mean_value = 0;
for i=0:(columns(poisson(index, :)) - 1)
  mean_value = mean_value + i .* poisson(index,i+1);
endfor

display("mean value of Poisson with lambda 30 is");
display(mean_value);

second_moment = 0;
for i = 0 : (columns(poisson(index, :)) - 1)
  second_moment = second_moment + i .* i .* poisson(index, i + 1);
endfor

variance = second_moment - mean_value .^ 2;
display("Variance of Poisson with lambda 30 is");
display(variance);
```

Γ)

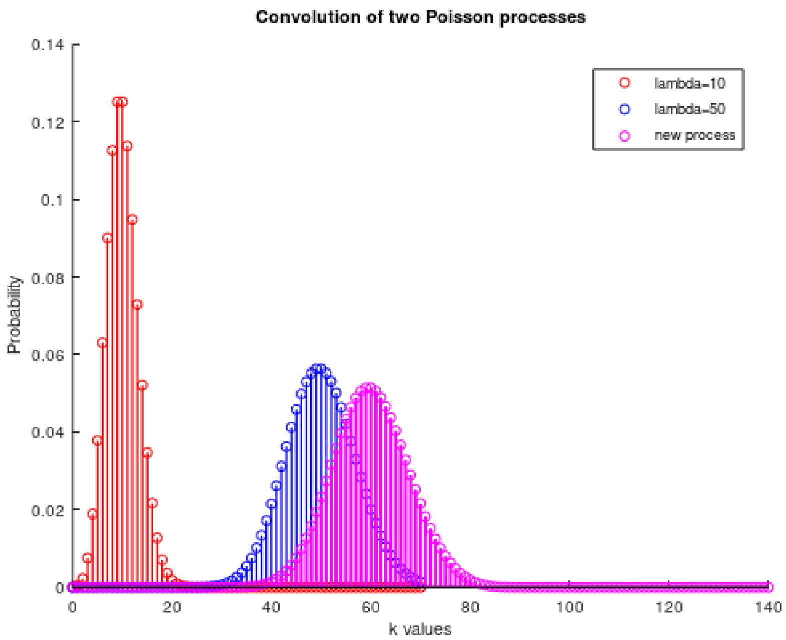
Έστω $P_X(x)$, $P_Y(y)$, $P_{X+Y}(z)$, οι συναρτήσεις μάζας πιθανότητας των *ανεξάρτητων* τυχαίων μεταβλητών X, Y και X+Y.

Για X=k, τότε Y = z-k.

Οπότε, $P_{X+Y}(z) = \sum_{x=0}^{z} P_X(x)P_Y(z-x) \; = \sum_{x=0}^{z} e^{-\mu}\frac{\mu^x}{x!}e^{-\lambda}\frac{\lambda^{z-x}}{(z-x)!} =$ $\frac{e^{-(\mu+\lambda)}}{z!}\sum_{x=0}^{z}\binom{z}{x}\mu^x\lambda^{z-x} = \frac{e^{-(\mu+\lambda)}}{z!}(\mu+\lambda)^z$, δηλαδή η τυχαία μεταβλητή X+Y ακολουθεί την κατανομή Poisson με παράμετρο (λ+μ).

Επιλέγουμε κατανομές Poisson με παραμέτρους λ=10 και λ=50. Η συνέλιξη των δύο αυτών κατανομών είναι ουσιαστικά μία νέα κατανομή Poisson με λ =10+50=60.

Οι τρεις κατανομές παρουσιάζονται σε κοινό διάγραμμα, το οποίο απεικονίζεται παρακάτω:



Παρατηρούμε ότι η κατανομή που προέκυψε είναι κατανομή Poisson με παράμετρο λ=60, όπως προβλέψαμε και προηγουμένως. Η παράμετρος αποτελεί, ουσιαστικά, το άθροισμα των παραμέτρων των επιμέρους κατανομών. Με άλλα λόγια, η υπέρθεση δύο κατανομών Poisson αποτελεί τη συνέλιξη των δύο κατανομών και έχει ως παράμετρο το άθροισμα των παραμέτρων των δύο επιμέρους κατανομών.

Η απαραίτητη προϋπόθεση είναι να υπάρχει *τυχαιότητα* σε αυτή την συνέλιξη, δηλαδή οι κατανομές να ορίζονται με πιθανοτικό τρόπο.

Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

```
first = find(lambda == 10);
second = find(lambda == 50);
poisson_first = poisson(first, :);
poisson_second = poisson(second, :);

composed = conv(poisson_first, poisson_second);
new_k = 0 : 1 : (2 * 70);

figure(2);
hold on;
stem(k, poisson_first(:), colors(1), "linewidth", 1.2);
stem(k, poisson_second(:), colors(2), "linewidth", 1.2);
stem(new_k, composed, "mo", "linewidth", 2);
hold off;
title("Convolution of two Poisson processes");
xlabel("k values");
ylabel("Probability");
legend("lambda=10", "lambda=50", "new process");
```

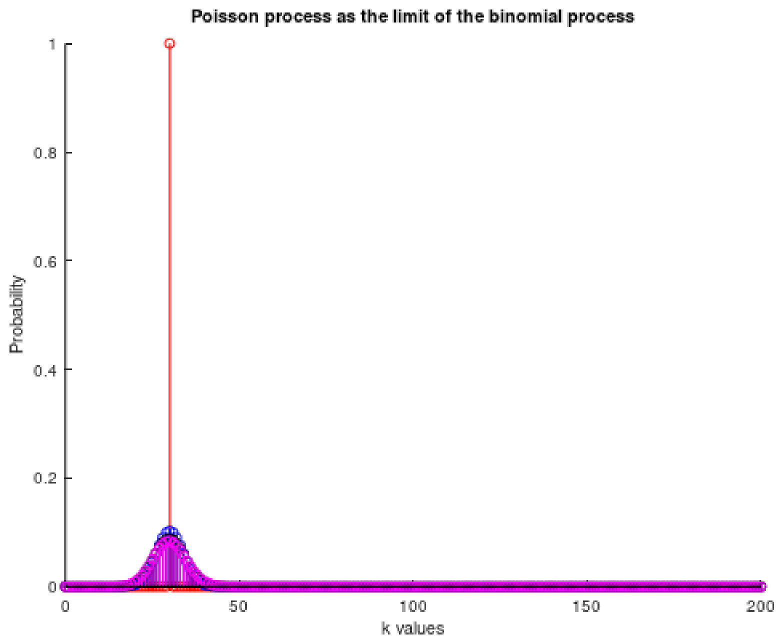
Δ)

Μια διακριτή τυχαία μεταβλητή X λέμε ότι ακολουθεί τη διωνυμική κατανομή με παραμέτρους N και p και έχει συνάρτηση μάζας πιθανότητας:

$$\Pr(\mathrm{X} = \mathrm{k}) = \binom{N}{k} p^k (1-p)^{N-k} \text{ για κάθε k= 0,1,…N με N≥1 και p∈(0,1)}$$

Μια διωνυμική κατανομή χαρακτηρίζεται από δύο παραμέτρους, τις N (τυχαία πειράματα) και p (η πιθανότητα εμφάνισης ενός ενδεχομένου). Μια κατανομή Poisson αποτελεί το όριο μίας διωνυμικής κατανομής, αρκεί ο αριθμός N να είναι πολύ μεγάλος, το p να τείνει στο 0 και το γινόμενο Np να είναι ίσο με την παράμετρο λ.

Κατασκευάζουμε μία κατανομή Poisson με παράμετρο λ=30 σημεία/sec και σχεδιάζουμε σε κοινό διάγραμμα την εξέλιξη μίας διωνυμικής κατανομής καθώς τείνει στην επιθυμητή κατανομή Poisson, όπως διακρίνεται παρακάτω.



Για μικρή τιμή του N υπάρχει μεγάλη απόκλιση από την κατανομή Poisson, όπως διακρίνεται στην κόκκινη γραφική παράσταση. Όσο αυξάνεται η τιμή του N, τόσο περισσότερο τείνει, η διωνυμική κατανομή, να ταυτιστεί με την κατανομή Poisson, όπως φαίνεται για παράδειγμα στην μπλε και μωβ γραφική παράσταση.

Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

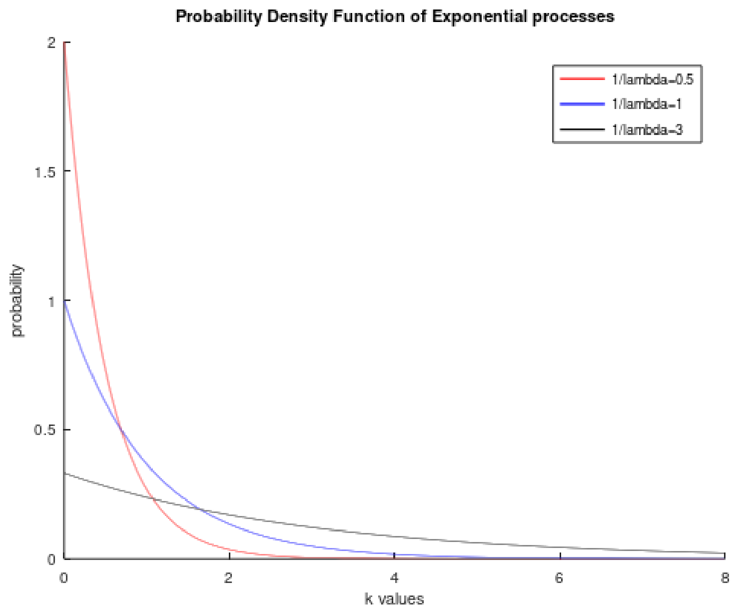
```
k = 0 : 1 : 200;
# Define the desired Poisson Process
lambda = 30;
i = 1 : 1 : 5;
n = lambda .* i;
p = lambda ./ n;

figure(3);
title("Poisson process as the limit of the binomial process");
xlabel("k values");
ylabel("Probability");
hold on;
for i = 1 : 4
  binomial = binopdf(k, n(i), p(i));
  stem(k, binomial, colors(i), 'linewidth', 1.2);
endfor
hold off;
```

<u>Εκθετική Κατανομή</u>

A)

Η συνάρτηση κατανομής πιθανότητας μίας συνεχούς τυχαίας μεταβλητής X η οποία ακολουθεί εκθετική κατανομή είναι: $f(x) = \begin{cases} 0, \ x < 0 \\ \frac{1}{\theta} e^{-\frac{x}{\theta}}, \ x \geq 0 \end{cases}$ με μέση τιμή E[X]=θ.

Αρχικά, σχεδιάζουμε τη συνάρτηση πυκνότητας πιθανότητας των εκθετικών κατανομών με μέσους όρους θ = 1/λ = [0.5,1,3] και προσεγγίζουμε τη συνάρτηση αυτή ως μία διακριτή με πολύ μικρό σφάλμα. Επιλέγουμε τις τιμές 0 έως 8 για τον οριζόντιο άξονα με την εντολή k:0:0.0001:8. Παρακάτω, απεικονίζονται οι ζητούμενες γραφικές παραστάσεις σε κοινό διάγραμμα:



Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

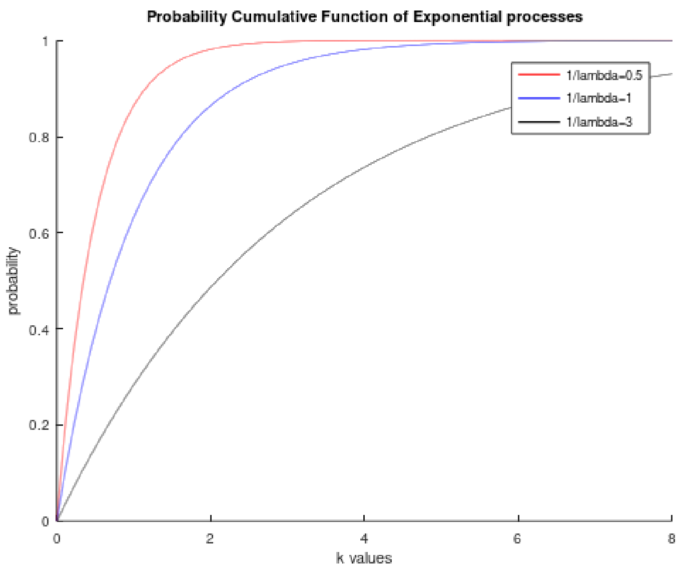
```
#A Task
k = 0:0.0001:8;
lambda = [0.5, 1, 3];

for i = 1 : columns(lambda)
  exp(i, :) = exppdf(k, lambda(i)); #Probability Density Function - Exponential
endfor

colors = "rbkm";
figure(1);
hold on;
for i = 1 : columns(lambda)
  plot(k, exp(i, :), colors(i), "linewidth", 1.2);
endfor
hold off;

title("Probability Density Function of Exponential processes");
xlabel("k values");
ylabel("probability");
legend("1/lambda=0.5", "1/lambda=1", "1/lambda=3");
```

B)

Η συνάρτηση κατανομής της εκθετικής συνάρτησης είναι ίση με:

$$F(x) = \begin{cases} Pr(X \le x) \; = \; 0, & x < 0 \\ Pr(X \le x) \; = \; \displaystyle\int_{-\infty}^{x} f(x')dx' \; = \int_{-\infty}^{x} \frac{1}{\theta} e^{-\frac{x'}{\theta}} dx' \; = \; 1 - e^{-\frac{x}{\theta}}, & x \ge 0 \end{cases}$$

Σχεδιάζουμε την αθροιστική συνάρτηση κατανομής των εκθετικών κατανομών του προηγούμενου ερωτήματος σε κοινό διάγραμμα, χρησιμοποιώντας τη συνάρτηση expcdf() του Octave. Οι ζητούμενες γραφικές παραστάσεις, απεικονίζονται παρακάτω:



Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

```
#B Task

# same k, lambda
for i = 1 : columns(lambda)              #Same as before
  exp(i, :) = expcdf(k,lambda(i)); #Cumulative Distribution Function - Exponential
endfor

colors="rbkm";
figure(2);
hold on;
for i = 1 : columns(lambda)
  plot(k,exp(i, :), colors(i), "linewidth", 1.2);
endfor
hold off;

title("Probability Cumulative Function of Exponential processes");
xlabel("k values");
ylabel("probability");
legend("1/lambda=0.5","1/lambda=1","1/lambda=3");
```

Γ)

Το υποερώτημα στηρίζεται στη χρήση της ιδιότητας έλλειψης μνήμης της εκθετικής συνάρτησης. Για κάθε z,y θετικοί αριθμοί: Pr(X ≥ z + y | X ≥ z) = Pr(X ≥ y), διότι:

$$\Pr(X \geq z + y \mid X \geq z) = \frac{Pr(\{X\geq z+y\}\wedge\{X\geq z\})}{Pr(X\geq z)} = \frac{Pr(X\geq z+y)}{Pr(X\geq z)} = \frac{1-Pr(X<z+y)}{1-Pr(X<z)} = \frac{1-\mathrm{F}(z+y)}{1-F(z)} =$$

$$\frac{e^{\frac{-(z+y)}{\theta}}}{e^{\frac{-z}{\theta}}} = e^{\frac{-y}{\theta}} = 1 - F(y) = \Pr(X \geq y)$$

Κάνουμε χρήση της συνάρτησης κατανομής πιθανότητας της εκθετικής κατανομής για 1/λ = 2.5 sec. Ισχύουν ότι:

$\Pr(X>30000) = 1-\Pr(X\leq 30000) = 1 - F(30000)$ και

$\Pr(X > 30000+20000 \mid X > 20000) = \frac{1-\mathrm{F}(50000)}{1-F(20000)} = 1 - F(30000)$ όπως αποδείξαμε.

Οι δύο αυτές πιθανότητες ορίστηκαν στο πρόγραμμα ως p1 και p2 αντίστοιχα.

```
P(X>30000)= p1 =  0.30121
P(X>50000|X>20000)= p2 =  0.30119
```

Παρατηρούμε ότι, στο command window, τυπώθηκαν τα ίδια αποτελέσματα και για τις δύο πιθανότητες.

Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

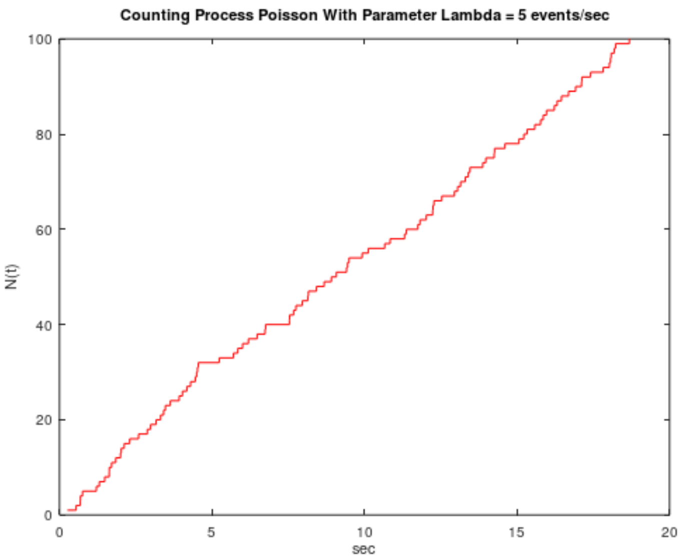
```
#C Task

# k same  with lambda = 2.5 sec
F = expcdf (k,2.5);
p1=1 - F(30000); # 1 - P(X <= 30000)
fprintf('P(X>30000)= '); p1 # print the probability P(X>30000)
p2=(1-F(50000))./(1-F(20000)); # (1-P(X<=50000))/(1-P(X<=20000))
fprintf('P(X>50000|X>20000)= '); p2 # print the probability P(X>50000|X>20000)
```

<u>Διαδικασία Καταμέτρησης Poisson</u>

A)

Οι χρόνοι που μεσολαβούν ανάμεσα στην εμφάνιση δύο διαδοχικών γεγονότων Poisson ακολουθούν, κατά τα γνωστά, εκθετική κατανομή με παράμετρο 1/λ.

Χρησιμοποιούμε την εντολή exprnd() με σκοπό να δημιουργήσουμε έναν πίνακα με 100 τυχαία γεγονότα, τα οποία ακολουθούν εκθετική κατανομή με μέση τιμή 1/λ = 0.2sec (για λ = 5 γεγονότα ανά sec). Έπειτα, με τη βοήθεια της συνάρτησης stairs(), σχεδιάζουμε μία διαδικασία καταμέτρησης poisson, όπως φαίνεται παρακάτω:



Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

```
# Exercise 3
#A Task
X = 100;               #100 successively randomly
y = ones(X,1);         #100 by 1 array of ones.
r = exprnd(0.2,1,X);   #Generate a random 1 by 100 array of exponential
                       #distribution with mean LAMBDA = 1/lambda = 0.2 sec.
for i = 1:(X-1)
  r(i+1) = r(i+1) + r(i);
  y(i+1) = y(i+1) + y(i);
endfor
figure(1);
stairs(r,y,color='r'); # the inputs r must be a row vector and y must a matrix
                        # with length(r) rows
hold on;
xlim("auto")            # set x axis limits automatically
title("Counting Process Poisson With Parameter Lambda = 5 events/sec");
xlabel("sec");
ylabel("N(t)");
```

B)

Ο αριθμός γεγονότων σε ένα χρονικό παράθυρο ΔΤ = t1-t2 ακολουθεί, κατά τα γνωστά, κατανομή Poisson με μέσο αριθμό γεγονότων λΔΤ, όπου λ είναι ο μέσος ρυθμός (rate) εμφανίσεων (γεγονότων ανά μονάδα του χρόνου) και μπορεί να προσεγγιστεί ως το πηλίκο του πλήθους των γεγονότων ως προς το συνολικό διάστημα που καταγράφηκαν.

Υπολογίζουμε το μέσο αριθμό γεγονότων στη μονάδα του χρόνου για (0)100, (i) 200, (ii) 300, (iii) 500, (iv) 1000 και (v) 10000 διαδοχικά τυχαία γεγονότα.

Υπολογίζουμε την προσέγγιση της τιμής του μέσου αριθμού γεγονότων στη μονάδα του χρόνου λ με τον τύπο: $\frac{\alpha\rho\iota\theta\mu\acute{o}\varsigma\ \gamma\varepsilon\gamma o\nu\acute{o}\tau\omega\nu}{\chi\rho\acute{o}\nu o}$.

(0)

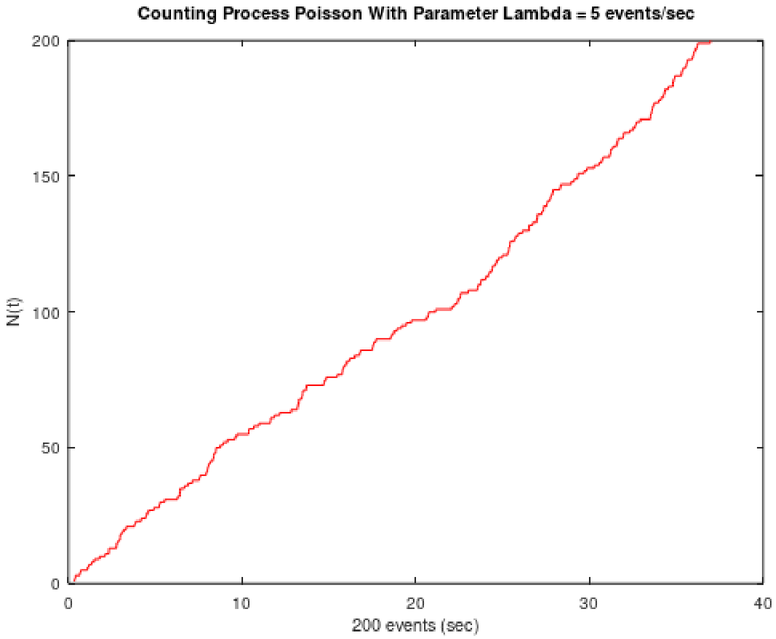
Το διάγραμμα που προκύπτει για τη διαδικασία καταμέτρησης Poisson N(t) σε αυτή την περίπτωση, είναι το ίδιο με εκείνου του προηγούμενου ερωτήματος. Η προσέγγιση της τιμής του μέσου αριθμού γεγονότων στη μονάδα του χρόνου λ είναι:

```
approximation of lambda (100 events) = l =  4.3314
```

Παρατηρούμε ότι υπάρχει αρκετή απόκλιση της θεωρητικής και πειραματικής τιμής λ.

(i)

Το διάγραμμα που προκύπτει για τη διαδικασία καταμέτρησης Poisson N(t) σε αυτή την περίπτωση είναι το παρακάτω:



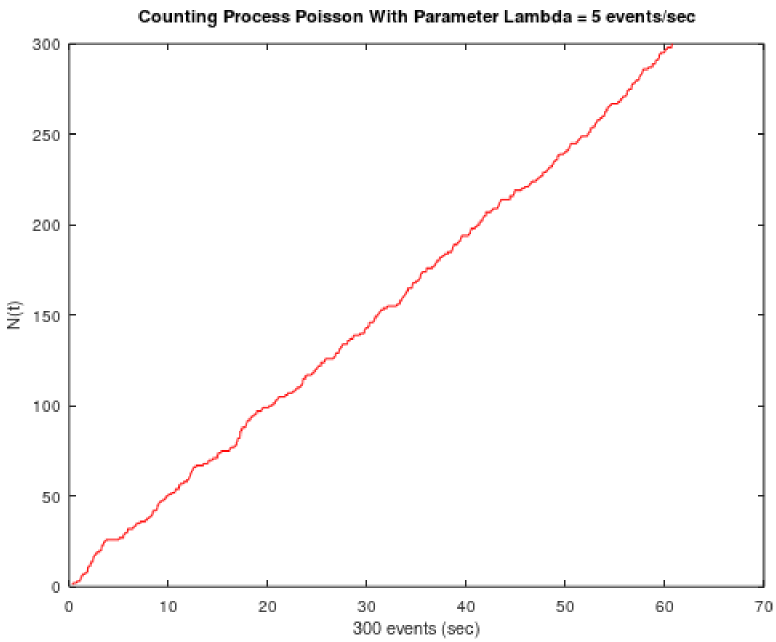
Η προσέγγιση της τιμής του μέσου αριθμού γεγονότων στη μονάδα του χρόνου λ είναι:

```
approximation of lambda (200 events) = l =  4.6157
```

Παρατηρούμε ότι υπάρχει αρκετή απόκλιση της θεωρητικής και πειραματικής τιμής λ, αλλά λιγότερη σε σχέση με εκείνη των 100 γεγονότων.

(ii)

Το διάγραμμα που προκύπτει για τη διαδικασία καταμέτρησης Poisson N(t) σε αυτή την περίπτωση είναι το παρακάτω:



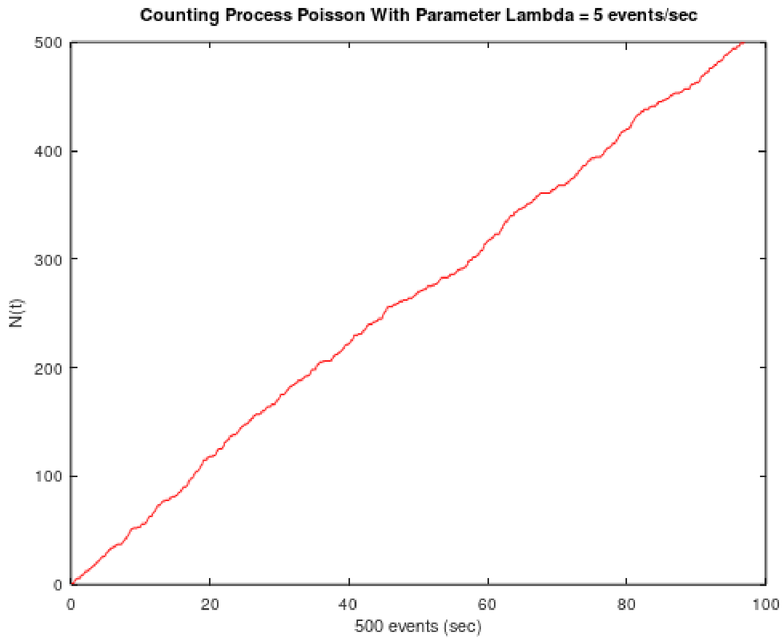
Η προσέγγιση της τιμής του μέσου αριθμού γεγονότων στη μονάδα του χρόνου λ είναι:

```
approximation of lambda (300 events) = l =  5.2623
```

Παρατηρούμε ότι υπάρχει απόκλιση της θεωρητικής και πειραματικής τιμής λ, αλλά λιγότερη σε σχέση με εκείνη των 200 γεγονότων.

(iii)

Το διάγραμμα που προκύπτει για τη διαδικασία καταμέτρησης Poisson N(t) σε αυτή την περίπτωση είναι το παρακάτω:



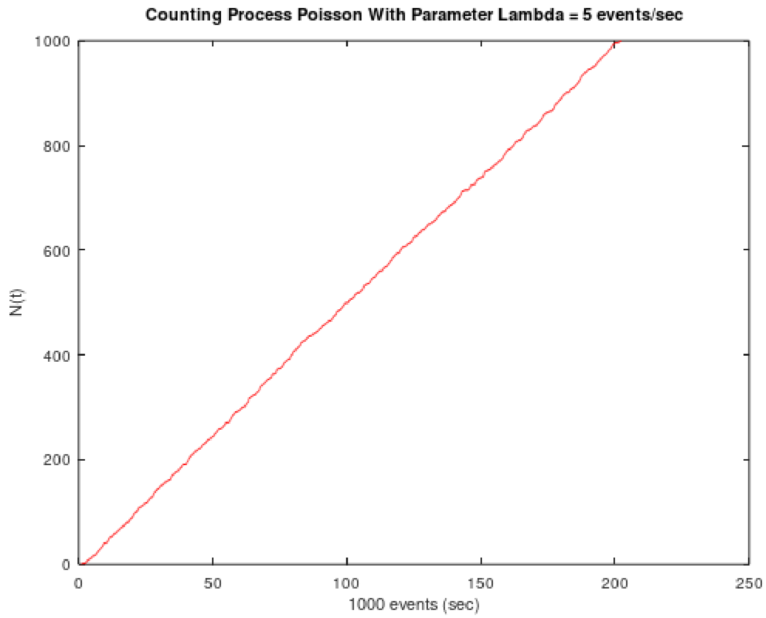
Η προσέγγιση της τιμής του μέσου αριθμού γεγονότων στη μονάδα του χρόνου λ είναι:

```
approximation of lambda (500 events) = l =  4.7867
```

Παρατηρούμε ότι υπάρχει απόκλιση της θεωρητικής και πειραματικής τιμής λ, αλλά λιγότερη σε σχέση με εκείνη των 300 γεγονότων.

(iv)

Το διάγραμμα που προκύπτει για τη διαδικασία καταμέτρησης Poisson N(t) σε αυτή την περίπτωση είναι το παρακάτω:



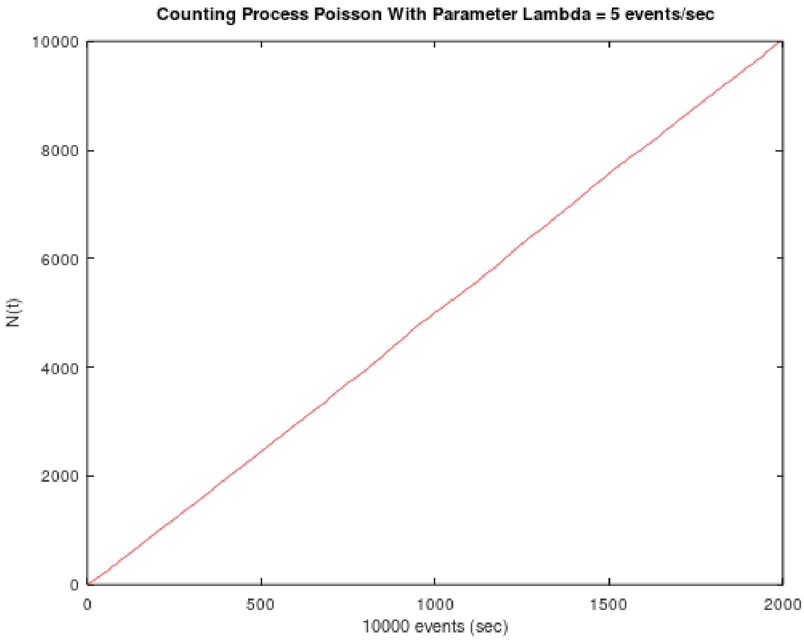
Η προσέγγιση της τιμής του μέσου αριθμού γεγονότων στη μονάδα του χρόνου λ είναι:

```
approximation of lambda (1000 events) = l =  5.1080
```

Παρατηρούμε ότι υπάρχει απόκλιση της θεωρητικής και πειραματικής τιμής λ, αλλά λιγότερη σε σχέση με εκείνη των 500 γεγονότων.

(v)

Το διάγραμμα που προκύπτει για τη διαδικασία καταμέτρησης Poisson N(t) σε αυτή την περίπτωση είναι το παρακάτω:



Η προσέγγιση της τιμής του μέσου αριθμού γεγονότων στη μονάδα του χρόνου λ είναι:

```
approximation of lambda (10000 events) = l =  5.0329
```

Παρατηρούμε ότι υπάρχει ελάχιστη απόκλιση της θεωρητικής και πειραματικής τιμής λ, αλλά λιγότερη σε σχέση με εκείνη των 1000 γεγονότων.

Παρακάτω, απεικονίζεται το command window στο τέλος της εκτέλεσης του προγράμματος:

```
approximation of lambda (100 events) = l =  4.3314
approximation of lambda (200 events) = l =  4.6157
approximation of lambda (300 events) = l =  5.2623
approximation of lambda (500 events) = l =  4.7867
approximation of lambda (1000 events) = l =  5.1080
approximation of lambda (10000 events) = l =  5.0329
```

Παρατηρούμε ότι, αυξάνοντας τα διαδοχικά τυχαία γεγονότα, η προσέγγιση του λ τείνει με περισσότερη ακρίβεια στη θεωρητική τιμή του λ = 5 γεγονότα/sec. Στη συγκεκριμένη περίπτωση, για 10000 γεγονότα, η προσέγγιση της τιμής του μέσου αριθμού γεγονότων στη μονάδα του χρόνου λ πλησιάζει σε μεγάλο βαθμό τη θεωρητική τιμή του λ = 5 γεγονότα/sec. Τέλος, παρατηρούμε ότι, καθώς αυξάνονται τα γεγονότα, τόσο περισσότερο τείνει να γίνει ευθεία το διάγραμμα σκάλας που προκύπτει για τη διαδικασία καταμέτρησης Poisson N(t).

Παρακάτω, παρατίθεται ο κώδικας που χρησιμοποιήθηκε για αυτό το ερώτημα:

```
#B Task
#(0) 100 events
l = 100/r(100);
fprintf('approximation of lambda (100 events) = '); l

#(i) 200 events
X = 200;              #200 successively randomly
y = ones(X,1);        #200 by 1 array of ones.
r = exprnd(0.2,1,X);  #Generate a random 1 by 200 array of exponential
                      #distribution with mean LAMBDA = 1/lambda = 0.2 sec.
for i = 1:(X-1)
  r(i+1) = r(i+1) + r(i);
  y(i+1) = y(i+1) + y(i);
endfor
figure(2);
stairs(r,y,color='r'); # the inputs r must be a row vector and y must a matrix
                       # with length(r) rows
hold on;
xlim("auto")           # set x axis limits automatically
title("Counting Process Poisson With Parameter Lambda = 5 events/sec");
xlabel("200 events (sec)");
ylabel("N(t)");
hold on;
l = 200/r(200);
fprintf('approximation of lambda (200 events) = '); l

#(ii) 300 events
X = 300;              #300 successively randomly
y = ones(X,1);        #300 by 1 array of ones.
r = exprnd(0.2,1,X);  #Generate a random 1 by 300 array of exponential
                      #distribution with mean LAMBDA = 1/lambda = 0.2 sec.
for i = 1:(X-1)
  r(i+1) = r(i+1) + r(i);
  y(i+1) = y(i+1) + y(i);
endfor
figure(3);
stairs(r,y,color='r'); # the inputs r must be a row vector and y must a matrix
                       # with length(r) rows
hold on;
xlim("auto")           # set x axis limits automatically
title("Counting Process Poisson With Parameter Lambda = 5 events/sec");
xlabel("300 events (sec)");
ylabel("N(t)");
hold on;
l = 300/r(300);
fprintf('approximation of lambda (300 events) = '); l
```

```
#(iii) 500 events
X = 500;                #500 successively randomly
y = ones(X,1);          #500 by 1 array of ones.
r = exprnd(0.2,1,X);    #Generate a random 1 by 500 array of exponential
                        #distribution with mean LAMBDA = 1/lambda = 0.2 sec.
for i = 1:(X-1)
  r(i+1) = r(i+1) + r(i);
  y(i+1) = y(i+1) + y(i);
endfor
figure(4);
stairs(r,y,color='r');  # the inputs r must be a row vector and y must a matrix
                        # with length(r) rows
hold on;
xlim("auto")            # set x axis limits automatically
title("Counting Process Poisson With Parameter Lambda = 5 events/sec");
xlabel("500 events (sec)");
ylabel("N(t)");
hold on;
l = 500/r(500);
fprintf('approximation of lambda (500 events) = '); l


#(iv) 1000 events
X = 1000;               #1000 successively randomly
y = ones(X,1);          #1000 by 1 array of ones.
r = exprnd(0.2,1,X);    #Generate a random 1 by 1000 array of exponential
                        #distribution with mean LAMBDA = 1/lambda = 0.2 sec.
for i = 1:(X-1)
  r(i+1) = r(i+1) + r(i);
  y(i+1) = y(i+1) + y(i);
endfor
figure(5);
stairs(r,y,color='r');  # the inputs r must be a row vector and y must a matrix
                        # with length(r) rows
hold on;
xlim("auto")            # set x axis limits automatically
title("Counting Process Poisson With Parameter Lambda = 5 events/sec");
xlabel("1000 events (sec)");
ylabel("N(t)");
hold on;
l = 1000/r(1000);
fprintf('approximation of lambda (1000 events) = '); l


#(v) 10000 events
X = 10000;              #10000 successively randomly
y = ones(X,1);          #10000 by 1 array of ones.
r = exprnd(0.2,1,X);    #Generate a random 1 by 10000 array of exponential
                        #distribution with mean LAMBDA = 1/lambda = 0.2 sec.
for i = 1:(X-1)
  r(i+1) = r(i+1) + r(i);
  y(i+1) = y(i+1) + y(i);
endfor
figure(6);
stairs(r,y,color='r');  # the inputs r must be a row vector and y must a matrix
                        # with length(r) rows
hold on;
xlim("auto")            # set x axis limits automatically
title("Counting Process Poisson With Parameter Lambda = 5 events/sec");
xlabel("10000 events (sec)");
ylabel("N(t)");
hold on;
l = 10000/r(10000);
fprintf('approximation of lambda (10000 events) = '); l
```