

TMA4265 - Stochastic Processes

Semester Project

Candidate number
111111

October 29, 2012

Problem One

a.

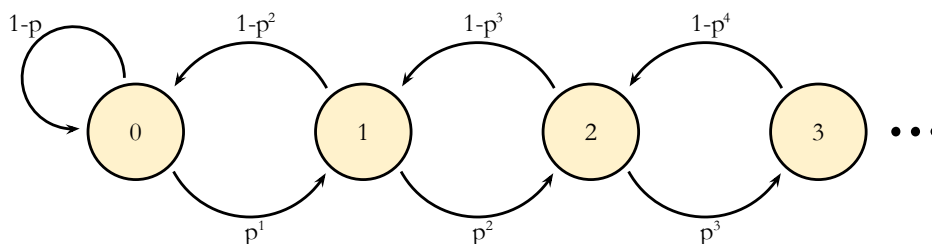


Figure 1: The Markov chain P_{ij} illustrated as a digraph.

The Markov chain P_{ij} in fig. 1 describes a random walk. Starting in any state, you can reach any other state - i.e. all states communicate. Thus, P_{ij} is **strongly connected**, has **one equivalence class** and is **irreducible**. Let X_n and X_{n+s} be two states such that there are $s - 1$ intermediate states between them. To get from state i to $j = i + s$ you have to visit all intermediate states at least once¹. Another trait of P_{ij} is that with growing i , the probability of moving from state i to $j = i + 1$ diminishes exponentially, while the probability of moving back to $j' = i - 1$ grows exponentially. Starting in state i , it is therefore reasonable to claim that it is always possible to return to state i in a finite number of steps. State i

¹Equivalently, s is the minimum number of steps you must perform to make the transition from state i to $i + s$

is then said to be positive recurrent, and since positive recurrence is a class property, the whole chain is **positive recurrent**.

Starting in state i , let n be the number that allows you to end up in state i after n transitions with the probability $P_{ii}^n > 0$. Let N_i be the set of all such numbers for our chain. The greatest common divisor of the elements in N_i is called the period d of state i . Since $P_{00}^n > 0$ for any n , $N_0 = 0, 1, 2, \dots$ and so P_{00} has a period of $d = 1$. Since periodicity is a class property, the chain as a whole has period one. A Markov chain with $d = 1$ is called **aperiodic**.

A positive recurrent aperiodic chain is called **ergodic**. Theorem 4.1 in [Ross 2010] states that for an irreducible ergodic Markov chain the limit $\pi_j = \lim_{n \rightarrow \infty} P_{ij}^n$, $j \geq 0$ exists,

$$\pi_j = \sum_{i=0}^{\infty} \pi_i P_{ij} \quad (1)$$

and

$$\sum_{j=0}^{\infty} \pi_j = 1. \quad (2)$$

In this case eq. (1) becomes

$$\pi_i = \begin{cases} \pi_{i-1}P_{i-1,i} + \pi_{i+1}P_{i+1,i} & = \pi_{i-1}p^i + \pi_{i+1}(1-p^{i+2}) & i > 0 \\ \pi_i P_{i,i} + \pi_{i+1}P_{i+1,i} & = \pi_0(1-p) + \pi_1(1-p) & i = 0 \end{cases} \quad (3)$$

The π 's are called the limiting probabilities that the process will be in state i at time n , and it can be shown that π_i also equals the long-run proportion of time that the process will be in state i . Below the limiting probabilities are calculated for the first 3 states.

$$\pi_1 = \frac{p}{1-p^2}\pi_0; \quad \pi_2 = \frac{p^2}{1-p^3}\pi_1; \quad \pi_3 = \frac{p^3}{1-p^4}\pi_2$$

By induction this yields the general formula (recurrence relation)

$$\begin{aligned}\pi_{i+1} &= \frac{p^{i+1}}{1-p^{i+2}}\pi_i = \frac{p^{i+1}p^i p^{i-1} \dots p^2 p^1}{(1-p^{i+2})(1-p^{i+1})(1-p^i) \dots (1-p^3)(1-p^2)}\pi_0 \\ &= \frac{p^{1+2+\dots+(i-1)+i+(i+1)}}{\prod_{k=2}^{k=i+2}(1-p^k)}\pi_0 = \frac{p^{\frac{1}{2}(i+1)(i+2)}}{\prod_{k=2}^{k=i+2}(1-p^k)}\pi_0\end{aligned}$$

or, equivalently,

$$\pi_i = \pi_0 \cdot \prod_{k=1}^{k=i+1} \frac{p^k}{1-p^{k+1}} \quad (4)$$

Now eq. (2) becomes

$$\begin{aligned}\sum_i \pi_i &= \pi_0 \left(1 + \sum_i \cdot \prod_{k=1}^{k=i+1} \frac{p^k}{1-p^{k+1}} \right) = 1 \\ \implies \pi_0 &= \left(1 + \sum_i \cdot \prod_{k=1}^{k=i+1} \frac{p^k}{1-p^{k+1}} \right)^{-1}\end{aligned} \quad (5)$$

Substituting eq. (5) into eq. (4) gives an expression for the limiting probabilities

$$\pi_i = \frac{\prod_{k=1}^{k=i+1} \frac{p^k}{1-p^{k+1}}}{1 + \sum_i \cdot \prod_{k=1}^{k=i+1} \frac{p^k}{1-p^{k+1}}} \quad (6)$$

The distribution is positive so the limiting probabilities exist and the states are indeed positive recurrent.

b.

Let the set ξ contain the first 10 states, and let T_i be the time spent to reach state 10, starting in state $X_0 = i \in \xi$.

$$\begin{aligned}\mu_i &= E[T_i] \\ &= E[T_i|X_1 = i-1]P(X_1 = i-1) + E[T_i|X_1 = i+1]P(X_1 = i+1) \\ &= (1 + E[T_{i-1}])(1-p^{i+1}) + (1 + E[T_{i+1}])p^{i+1} \\ &= 1 + \mu_{i-1} + (\mu_{i+1} - \mu_{i-1})p^{i+1}\end{aligned} \quad (7)$$

Note the special cases for $\mu_0 = 1 + \mu_0 + (\mu_1 - \mu_0)p$, and $\mu_9 = 1 + \mu_8(1 - p^{10})$

eq. (7) constitutes a set of 10 equations that can be solved with MATLAB. In table 1 the solutions are listed for $p = 0.75$ and $p = 0.90$.

i	$E[T_i]$	
	p = 0.75	p = 0.90
0	$3.504 \cdot 10^6$	100.679
1	$3.504 \cdot 10^6$	99.568
2	$3.504 \cdot 10^6$	98.073
3	$3.504 \cdot 10^6$	96.145
4	$3.504 \cdot 10^6$	93.610
5	$3.504 \cdot 10^6$	90.159
6	$3.503 \cdot 10^6$	85.235
7	$3.503 \cdot 10^6$	77.772
8	$3.501 \cdot 10^6$	65.577
9	$3.485 \cdot 10^6$	43.711

Table 1: Mean values of reaching state 10 from any state $i = 0, 1, \dots, 9$

The exponential decay for the probabilities for moving on to the next state clearly manifests itself for the case of $p = 0.75$; no matter what state you start in, you'll have to wait for a very long time to reach state 10. The transition from state 9 to 10 is taking very long time compared to the other one-step transitions.

c.

The probability for ever reaching state 100 is 1, no matter what state you start in, since all limiting probabilities are positive. If you start in state $X_0 < 100$, you will have to wait for a very long time to reach state 100, and if $X_0 > 100$ you will only have to wait close to $X_0 - 100$ steps (at least if not $p \approx 1$).

d.

table 2 draws a comparison between the analytical and simulated values for the limiting probabilities discussed in **a**. Our estimates are pretty close to the analytical values and comfortably within the confidence intervals in the column to the left. This suggests that our estimation was good.

MATLAB is here used to illustrate some random walks.

j	π_j	$\bar{\pi}_j$	C.I.
0	.1669	0.1677	[.1666, .1688]
1	.2861	0.2867	[.2856, .2877]
2	.2789	.2774	[.2767, .2782]
3	.1718	.1708	[.1699, .1717]
4	.0713	.0714	[.0706, .0721]
5	.0206	.0205	[.0201, .0210]
6	.0042	.0042	[.0040, .0044]
7	.0006	.0006	[.0005, .0006]
8	.0001	.0001	[.0000, .0001]
9	0	0	[0 , 0]
10	0	0	[0 , 0]

Table 2: $p=0.75$: The analytical and simulated limiting probabilities for the first 11 states are shown together with the corresponding 95% confidence interval.

```

hold on;
%%% A ten-step illustrated random walks, p=0.90
% Starting in state 0
plot(randwalk(20,0,0.9), "1", \
randwalk(20,0,0.9), "2", randwalk(20,0,0.9), "3" )
% Starting in state 5
plot(randwalk(20,5,0.9), "1", \
randwalk(20,5,0.9), "2", randwalk(20,5,0.9), "3" )
% Starting in state 9
plot(randwalk(20,9,0.9), "1", \
randwalk(20,9,0.9), "2", randwalk(20,9,0.9), "3" )

%%% p=0.75
% Starting in state 0
plot(randwalk(20,0,0.75), "1", \
randwalk(20,0,0.75), "2", randwalk(20,0,0.75), "3" )
% Starting in state 5
plot(randwalk(20,5,0.75), "1", \
randwalk(20,5,0.75), "2", randwalk(20,5,0.75), "3" )
% Starting in state 9
plot(randwalk(20,9,0.75), "1", \
randwalk(20,9,0.75), "2", randwalk(20,9,0.75), "3" )

```

Notice the difference between $p = 0.90$ and $p = 0.75$ in figs. 2 and 3. Walking with $p = 0.75$ does indeed pull you down towards state 0.

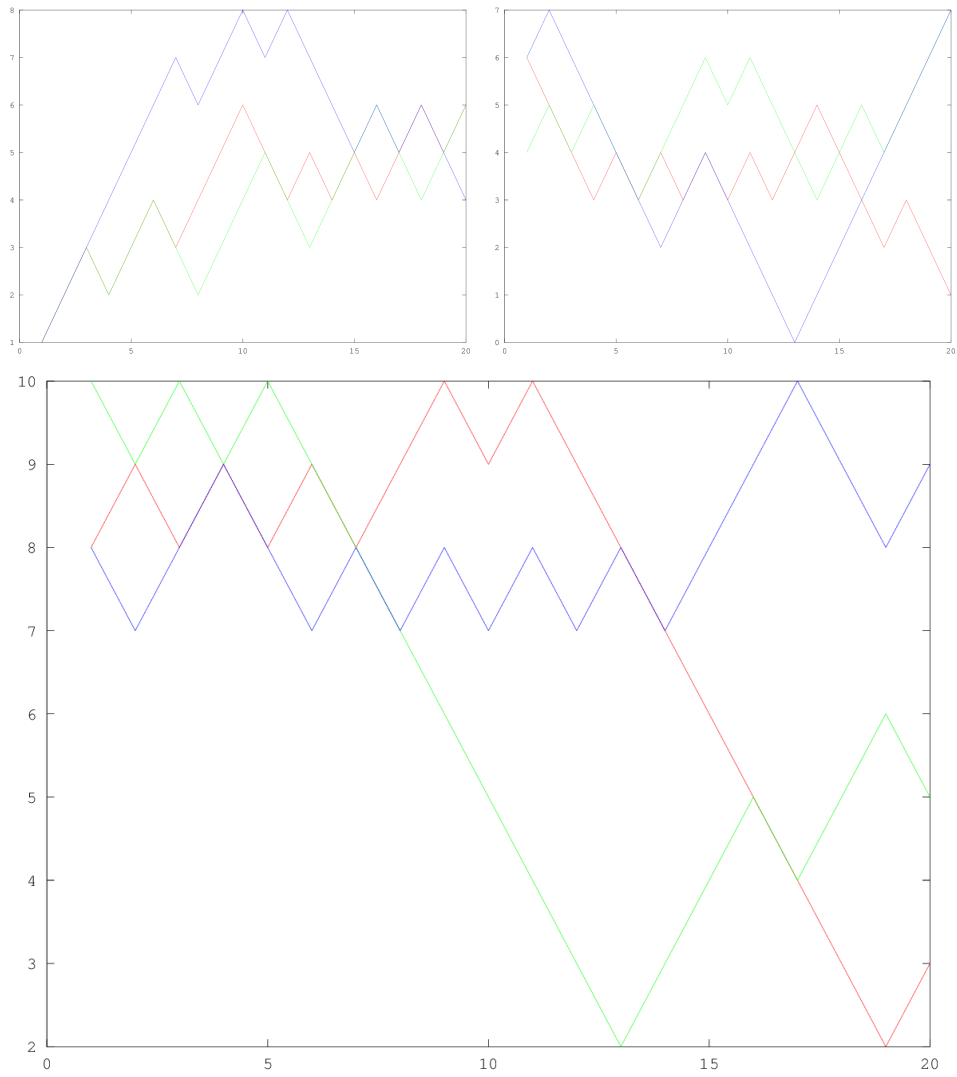


Figure 2: A 20-step random walk w. $p = 0.90$ starting with state 0, 5 and 9, respectively.

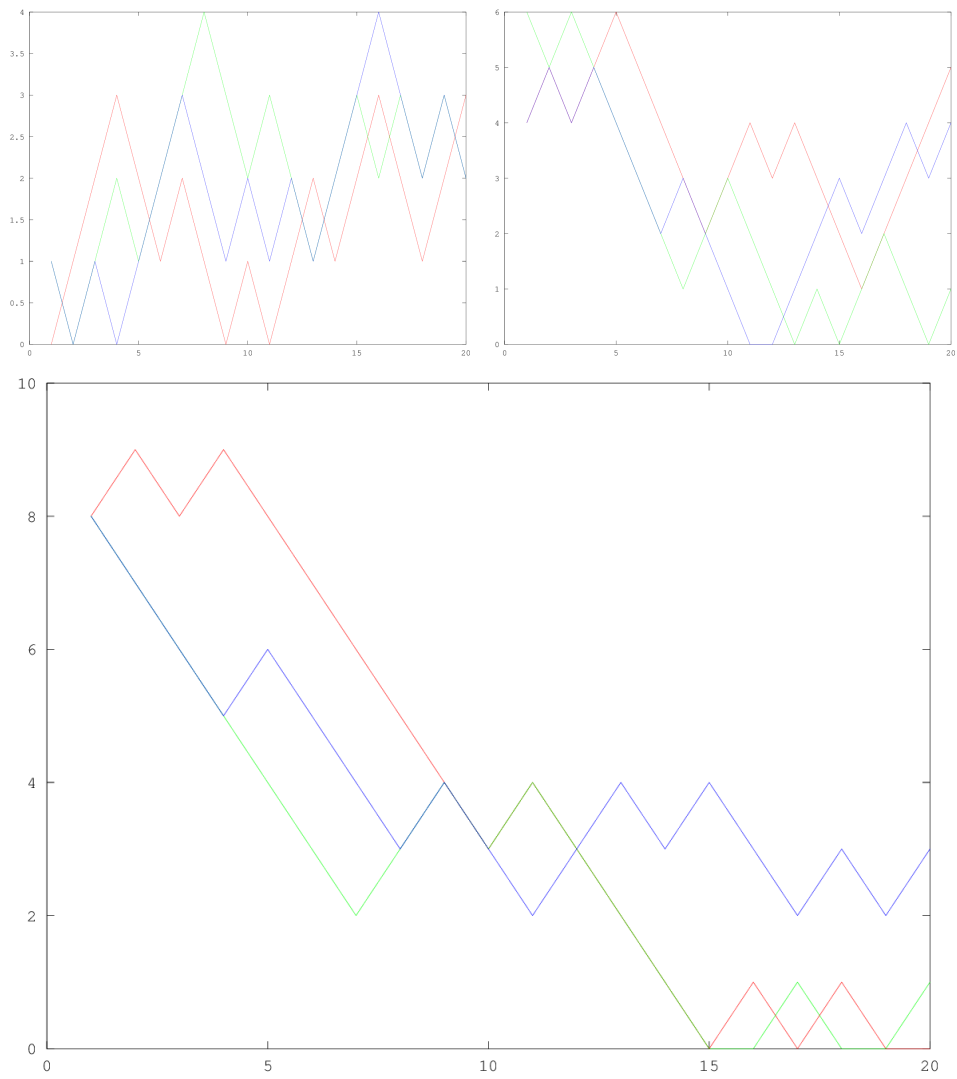


Figure 3: A 20-step random walk w. $p = 75$ starting with state 0, 5 and 9, respectively.

Problem Two

In the following X_n is the number of individuals in the n 'th generation, X_0 is the initial population, $\mu = E[\text{children per individual}]$, $p_j = P\{\text{an individual has } j \text{ offspring}\}$. The distributions that are analyzed are listed in table 3

	p_0	p_1	p_2	p_3
I	0.6	0.05	0.15	0.2
II	0.25	0.60	0.10	0.05

Table 3: The probability distributions given in Problem Two.

Chapter 4.7 in [Ross 2010] presents some results that are used in this problem. Below is some properties of branching processes discussed.

Generally, the mean number, and the variance, of offspring of a single individual is

$$\mu = \sum_{j=0}^{\infty} jP_j \quad \text{and} \quad \sigma^2 = \sum_{j=0}^{\infty} (j - \mu)^2 P_j \quad (8)$$

respectively.

For both our distributions $\mu = 0.95$, and since $\mu < 1$ the population will eventually die out. Defining Z_i to be the number of offspring of the i 'th individual for the $(n - 1)$ th generation, one can find

$$X_n = \sum_{i=1}^{X_{n-1}} Z_i \quad (9)$$

in the edge case where $X_0 = 1$. One can then obtain

$$E[X_n] = E[E[X_n|X_{n-1}]] = E \left[E \left[\sum_{i=1}^{X_{n-1}} Z_i | X_{i-1} \right] \right] = E[X_{n-1}]\mu \quad (10)$$

which leads to the result

$$E[X_1] = \mu, \quad E[X_2] = \mu E[X_1] = \mu^2, \quad \dots, \quad E[X_n] = \mu^n. \quad (11)$$

It can then be shown that the variance is

$$Var(X_n) = \begin{cases} \sigma^2 \mu^{n-1} \cdot \frac{1-\mu^n}{1-\mu}, & \mu \neq 1 \\ n\sigma^2, & \mu = 1 \end{cases} \quad (12)$$

When doing simulations, the estimated mean value and standard deviation is given by

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n X_i, \quad S^2 = \frac{1}{n-1} \sum_{i=1}^n X_i^2 - \bar{X}^2, \quad (13)$$

respectively.

a.

The mean and standard deviation is found analytically from eq. (8), eq. (11) and eq. (12), and is presented in table 4.

n	$E[X_n]$	$SD_I[X_n]$	$SD_{II}[X_n]$
10	0.5987	2.7977	2.7692
100	0.0059	0.4379	0.0678
1000	5.2912e-23	4.1529e-11	2.4697e-11

Table 4: Analytical values of the mean $E[X_n]$ and the standard deviation, $SD (= Var^{\frac{1}{2}}[X_n])$ for distribution I and II.

The branching process was simulated a hundred thousand times for each of the probability distributions. The values in table 5 were computed by running

```
% No. of simulations
N = 100000;
% Size of the first population
init = 1;
% Distribution I = p1 and II = p2
p1 = [0.60 0.05 0.15 0.20];
p2 = [0.25 0.60 0.10 0.05];
% Run the simulations for I and II
branchtrials(N, init, p1);
branchtrials(N, init, p2);
```

The simulated values is indeed very close to the ones found analytically for both distributions. Given that the MATLAB rand()-function is uniform, the

n	$E_I[X_n]$	$E_{II}[X_n]$	$E_{III}[X_n]$	$SD_I[X_n]$	$SD_{II}[X_n]$	$SD_{III}[X_n]$
10	0.6007	0.5989	0.0554	2.8099	1.6708	0.4662
100	0.0060	0.0050	0	0.4583	0.2392	0
1000	0	0	0	0	0	0

Table 5: Simulated values of the mean $E[X_n]$ and the standard deviation, SD ($= Var^{\frac{1}{2}}[X_n]$) for distribution I and II.

simulated values would probably become even more accurate if the number of simulations is increased. No simulation ever reached $n = 1000$, but this is only natural considering the low probability for that to happen.

Case (i), (ii) and (iii) are presented in table 6 for both distributions.

Case	Distribution I		Distribution II		Distribution III	
	$E[\cdot]$	$SD[\cdot]$	$E[\cdot]$	$SD[\cdot]$	$E[\cdot]$	$SD[\cdot]$
(i)	4.5197	7.1409	8.2460	10.8820	3.5185	2.5587
(ii)	19.9991	109.4967	19.9830	65.9630	3.9787	7.0344
(iii)	2.9699	5.4466	2.3411	2.9429	1.5572	1.1611

Table 6: Simulated values of the mean $E[X_n]$ and the standard deviation, SD ($= Var^{\frac{1}{2}}[X_n]$) for distribution I and II.

b.

```
% Defining a third distribution
p3 = [0.50 0.30 0.15 0.05];
% Run the simulations for III
branchtrials(N, init, p3);
```

The three different cases are presented in figs. 4 to 6 as histograms, grouped by probability distribution.

References

[Ross 2010] Sheldon M. Ross (2010), Introduction to Probability Models.

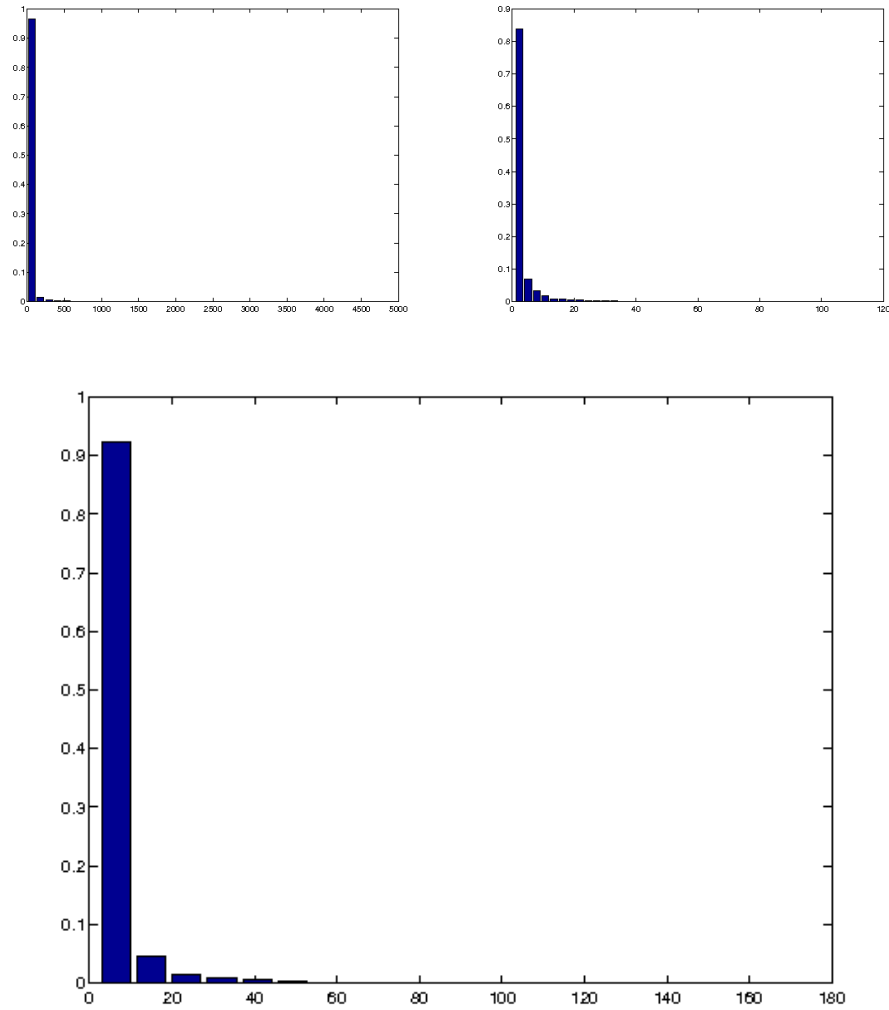


Figure 4: Case (i), (ii) and (iii) for distribution I

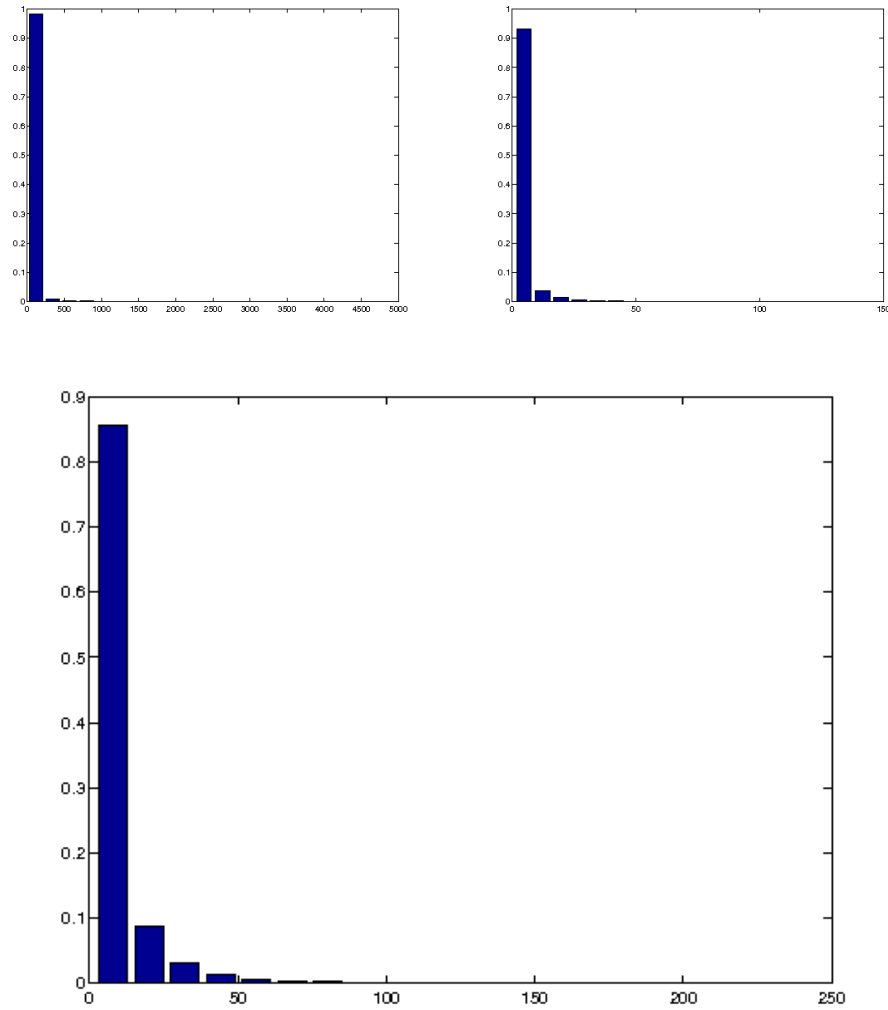


Figure 5: Case (i), (ii) and (iii) for distribution II

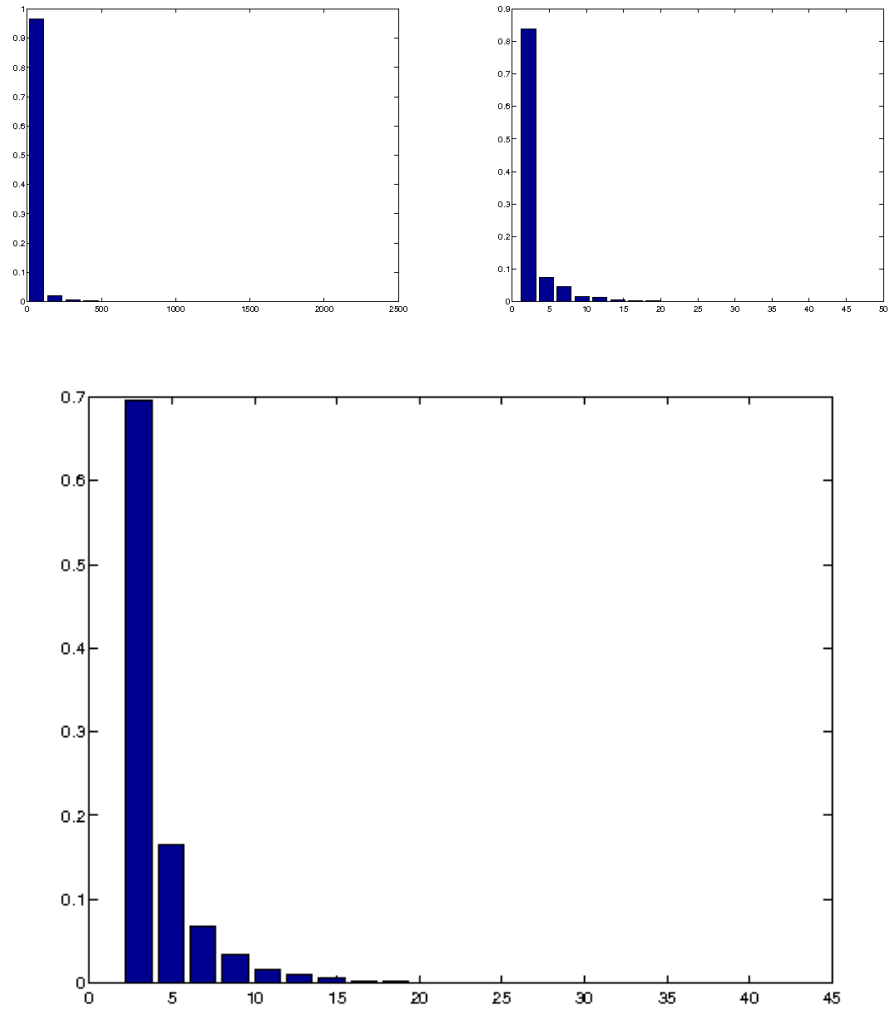


Figure 6: Case (i), (ii) and (iii) for distribution III

Problem One

```
1 function W = randwalk(T,i,p)
2
3 % ----// randwalk.m //--
4 %
5 % Simulates a random walk
6 %
7 %
8 % Input:
9 %   T - int, number of transitions
10 %   i - int, initial state
11 %   p - double, 0<p<1, probability.
12 %
13 % Output:
14 %   w - vector, with transtions.
15 %
16 % -----
17
18 W = zeros(1,T);
19 W(1) = i;
20 s = i; % current state s
21
22 for t=1:T
23     r=rand(1);
24     % Walk to the right
25     if r <= p^(s+1)
26         W(t) = s+1;
27         s = s+1;
28     % Walk to the left
29     elseif s>0
30         W(t) = s-1;
31         s=s-1;
32     % Return to state 0
33     end
34 end
35 end
```

Problem Two

```
1 function [popu, n, tot_people, largest_gen] = branchsim(init, dist)
2
```

```

3  % ----// branchsim.m ----
4  %
5  % Simulates a branching process
6  %
7  %
8  % Input:
9  %   init - int, initial size of the population
10 %   dist - vector, offspring probability
11 %           distribution [p0, ..., pn]
12 %
13 % Output:
14 %   popu - vector, population for each generation.
15 %   n     - int, number of generations
16 %   tot_people - int, total number of people lived.
17 %   largest_gen - int, size of the largest generation.
18 %
19 % -----
20
21 popu = zeros(1,2000);
22 % The population of the 1st generation
23 popu(1) = 1;
24 % Accumulated probabilities to be used w. rand(1)
25 cumdist = cumsum(dist);
26 len = length(dist);
27 tot_people = init;
28 largest_gen = init;
29
30 % Generation number n
31 n = 1;
32 while popu(n) > 0
33     popu(n+1) = 0;
34     for j=1:popu(n)
35         r = rand(1);
36         if r < cumdist(1)
37             popu(n+1) = popu(n+1) + 0;
38         elseif r < cumdist(2)
39             popu(n+1) = popu(n+1) + 1;
40         elseif r < cumdist(3)
41             popu(n+1) = popu(n+1) + 2;
42         else
43             popu(n+1) = popu(n+1) + 3;
44         end
45     end
46     tot_people = tot_people + popu(n+1);

```

```
47         if largest_gen < popu(n+1)
48             largest_gen = popu(n+1);
49         end
50         n=n+1;
51     end
52 end
```