1 Random signals and stochastic processes

Contents

1.1 Statistical estimation

1.1.1 Expected Value

To generate a 1000 random values in a vector in matlab we use the following code:

```
x = rand(1000,1);

a=0;
for i=1:1000
    a = a+x(i);
end
a=a/1000;
a
mean(x)
stem(x)
```

Some sample values obtained are:

mean(x)	Manually calculated
0.4958	0.4958
0.4948	0.4948
0.5118	0.5118
0.4984	0.4984
0.5127	0.5127
0.4867	0.4867

As we can see both values are consistent and to the same degree of accuracy. Increasing the decimal places to 20 also led to both functions to have the same accuracy and precision (i.e. they are the same number).

1.1.2 Sample standard deviation

To calculate the standard deviation we use the following script:

```
x = rand(1000,1);

a=0;
m=mean(x);
for i=1:1000
    a = a+(x(i)-m)^2;
end
a=a/999;
a=sqrt(a);
sprintf('%.17f',a) %17 digits is maximum accuracy in matlab
sprintf('%.17f',std(x))
```

Some values calculated for various sets include:

Again both the custom defined function and the internal matlab function return the same values to the same decimal accuracy (with matlab giving a maximum of 17 decimal places).

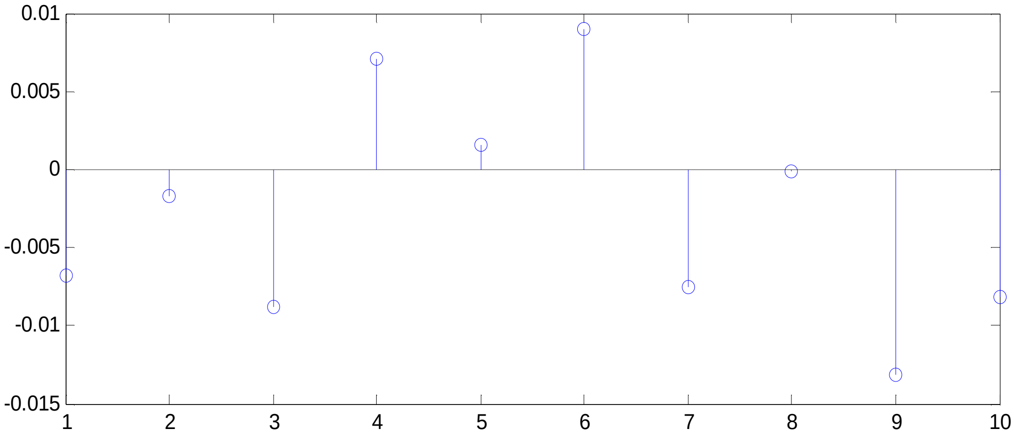
std(x)	calculated
0.284984	0.284984
0.279909	0.279909
0.289126	0.289126
0.288586	0.288586
0.287258	0.287258
0.286541	0.286541

1.1.3 Bias estimation

To calculate the bias we use the following script:

```
a=0;
for i=1:10
x = rand(1000,1);
m=mean(x);
a(i) = 0.5-m; %E(X) = 0.5 for N~U(0,1)
end
stem(a)
```

Which resulted, for a certain run of it, in the following graph:



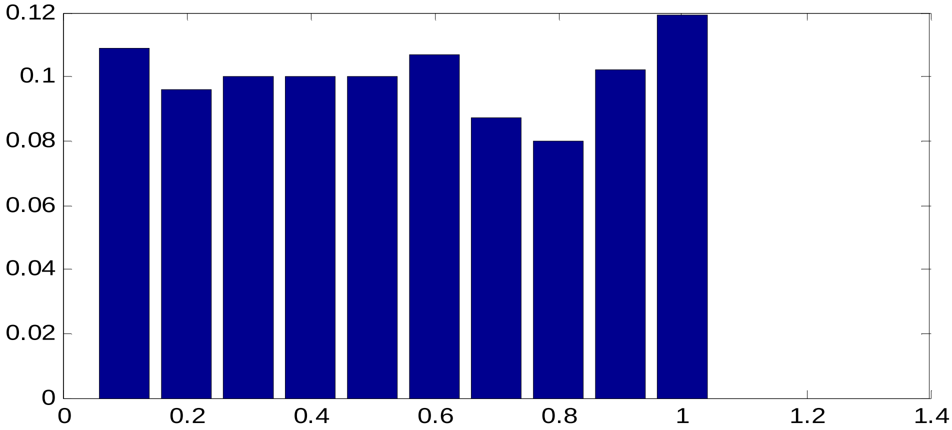
It can be seen that the randomly generated samples all tend towards 0, clustering around their theoretical values.

1.1.4 Probability density function

A pdf was approximated using the following function:

```
a=0;
x = rand(1000,1);
a = hist(x)./1000;
b=.1:0.1:1
bar(b,a)
```

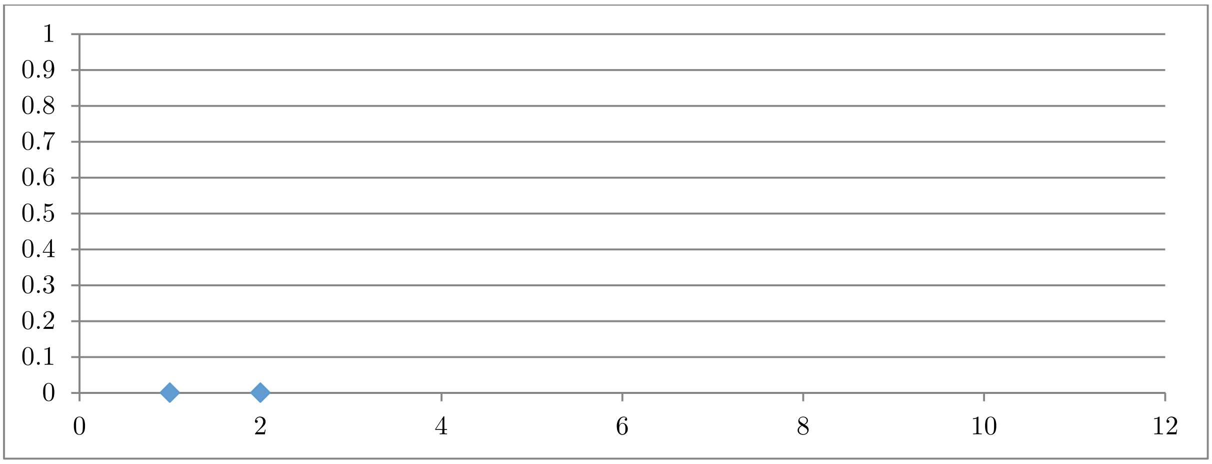
From this the following bar was obtained, which shows an expected bar graph, with each value having a probability of 0.1 of occurring.



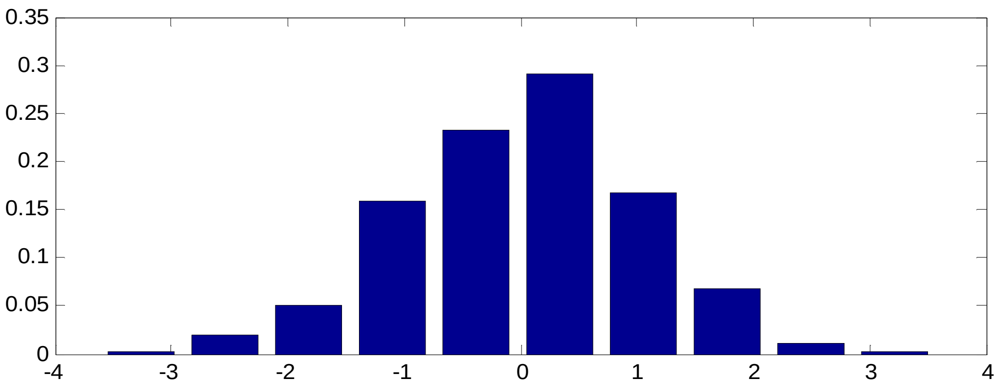
1.1.5 With Gaussian distributed random variable.

Repeating steps 1-4 for a Gaussian distributed variable we find the mean to be around 0, with observed values including: -0.020069, -0.027927 and 0.011566. The standard deviation is, as expected around 1, with observed values including: 0.965608, 0.978326 and 1.041055.

The following graph provides an insight into bias estimation of various iterations. As we can see the values gravitate towards the expected value, which is 0.



A probability density function can be estimated using the hist function and gives us a distribution similar to that of its theoretical Gaussian distribution function.



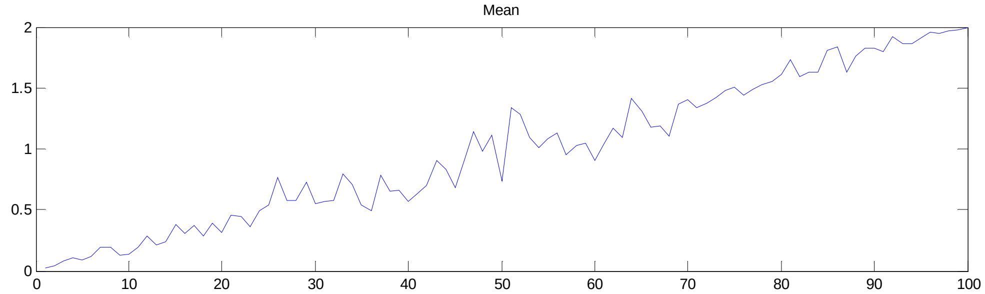
The code used to calculate it is the following:

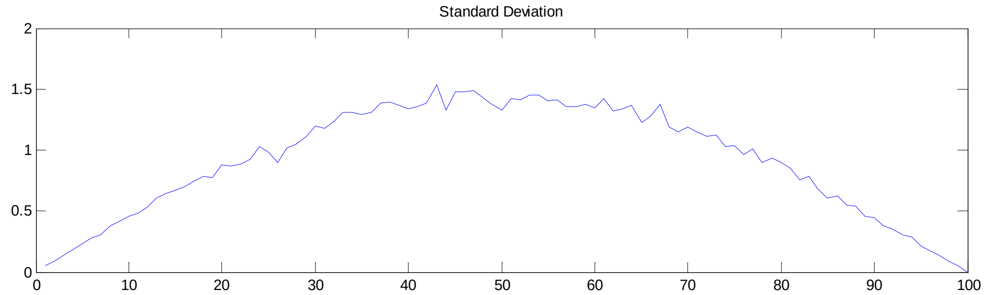
```
x = randn(1000,1);
[a,b] = hist(x);
a = a./1000;
sprintf('mean:%.6f',mean(x))
sprintf('std:%.6f',std(x))
bar(b,a)
```

1.2 Stochastic processes

1.2.1 Ensemble mean and standard deviation for M,N = 100

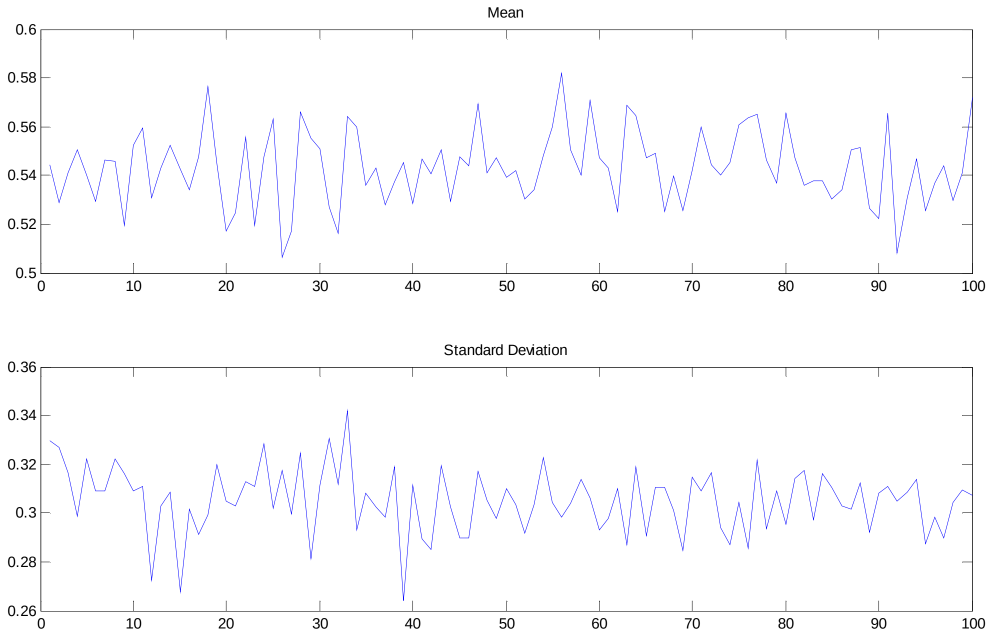
The following graph shows the evolution of the ensemble mean and standard deviation evolution over time for the rp1 stochastic process.





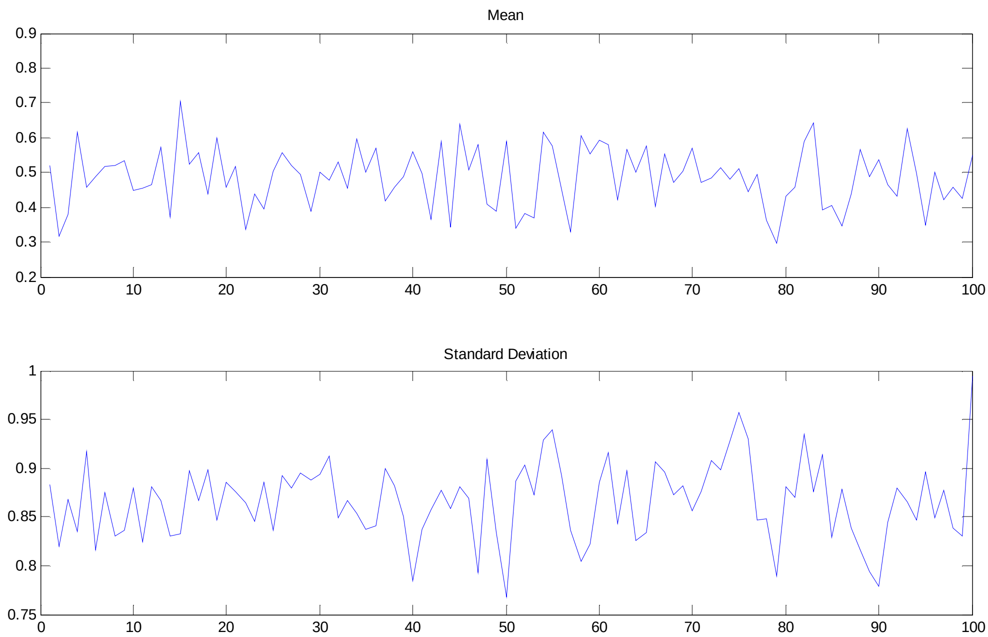
From this we can observe that the rp1 process is not stationary as its mean and standard deviation change when shifted in time. Indeed the mean increases linearly over time with a trend line $mean = \frac{time\ sample\ \#}{50}$ for this particular case. The standard deviation does not have a linear trend but rather a trend line best approximated by the following function: $s.d. = 1.44 \times \sin(\frac{time\ sample\ \#}{time\ sample\ total} \times \pi)$. Again this is not a constant and depends on time thus we conclude rp1 is a non-stationary process.

The following graph shows the ensemble mean and standard deviation plotted over time for the rp2 process:



We can notice that the rp2 process is stationary with a constant expected mean (in this case just above 0.5) and standard deviation of $\frac{1}{3}$.

The following graph shows the ensemble mean and standard deviation plotted over time for the rp3 process:



The rp3 process is also stationary with an expected mean of 0.5 and expected standard deviation of .86.

These graphs were made from the following code:

```
clc;
a = rp2(100,100);
```

```
subplot(2,1,1); plot(mean(a))
title('Mean')
subplot(2,1,2); plot(std(a))
title('Standard Deviation')
mean(mean(a))
mean(std(a))
```

1.2.2 Ergodicity of processes for M=4 and N=1000

In the following cases we are essentially taking 4 samples at 1000 different occurrences in time, i.e. we have 1000 different realisations of 4 samples.

For the rp1 process we can conclude that it is not ergodic due to the average of the ensembles not being the same as the time average, due to the nature of the trendline established earlier. Essentially a reading of many samples at one point in time will not help predict the full properties of this stochastic system.

For the rp2 process is not ergotic either. Indeed each set of occurrences and samples yield vastly different values. Thus observing one realisation of the system 1000 times will not provide us with all of the properties of the system, i.e. all the values that the system can take are not exhibited in one run and the starting value of the system determines the future instance of this system.

Iteration	**Expectation of Expectation of time**	**Expectation of Standard Deviation samples**
1	0.1195	0.207
2	0.4367	0.2713
3	0.5635	0.3172
4	0.4527	0.2134
5	0.2346	0.3048
6	0.7103	0.2522

The rp3 process is ergotic as the system's expected mean and standard deviation is constant and does not change over different iterations. Thus in just one instance of many time samples the properties of the system can be determined.

1.2.3 Mathematical representation

1.2.3.1 Process rp1

Rp1 can be represented by the function:

$$v_t = cb \times sin\left(\frac{t \times \pi}{T}\right) + at$$

Where:

a and b are constants and c is a uniformly distributed random variable between -0.5 and 0.5 with $E(c) = 0, \sigma_c = \sqrt{Var(c)} = \frac{1}{2\sqrt{3}}$

t is also a constant representing the iteration number with T representing the total number of iterations.

From this we calculate the expected value:

$$E(v_t) = E\left(cb \times sin\left(\frac{t \times \pi}{T}\right) + a \times t\right) = E(c) \times E\left(b \times sin\left(\frac{t \times \pi}{T}\right)\right) + E(at) = 0 \times E\left(b \times sin\left(\frac{t \times \pi}{T}\right)\right) + E(a \times t)$$
$$= at$$

And the standard deviation:

$$\sigma_{v_t}^2 = Var(v_t) = E(v_t^2) - E(v_t)^2 = E\left(c^2 b^2 \sin^2\left(\frac{t \times \pi}{T}\right) + 2 \times atcb \times sin\left(\frac{t \times \pi}{T}\right) + a^2t^2\right) - a^2t^2$$
$$= E\left(c^2 b^2 \sin^2\left(\frac{t \times \pi}{T}\right)\right) + E\left(2 \times atcb \times sin\left(\frac{t \times \pi}{T}\right)\right) + a^2t^2 - a^2t^2$$
$$= E\left(c^2 b^2 \sin^2\left(\frac{t \times \pi}{T}\right)\right) + E(c) \times E\left(2 \times atb \times sin\left(\frac{t \times \pi}{T}\right)\right) = E(c^2) \times b^2 \sin^2\left(\frac{t \times \pi}{T}\right)$$
$$= \frac{1}{12} b^2 \times \sin^2\left(\frac{t \times \pi}{T}\right)$$

$$\sigma_{t_v} = \frac{1}{\sqrt{12}} b \times sin\left(\frac{t \times \pi}{T}\right)$$

We know that the parameters where $a = \frac{1}{50}$ and $b = 5$, thus we can calculate the ideal values for expected value and standard deviation.

$E(v_t) = \frac{t}{50}$, which fits in perfectly with the found values.

$\sigma_{t_v} = \frac{1}{\sqrt{12}} b \times sin\left(\frac{t \times \pi}{T}\right) = \frac{5}{\sqrt{12}} \times sin\left(\frac{t \times \pi}{T}\right) \cong 1.443 \times sin\left(\frac{t \times \pi}{T}\right)$, which also fits in perfectly with the obtained standard deviation.

1.2.3.2 Process rp2

Rp2 can be represented by the function:

$$v_t = X_n + Y_n \times Z_{t,n}$$

Where:

n and t represent what the random variables are random to, for example X_n and Y_n are both random with regards to the sample instance but not the number of realisations and $Z_{t,n}$ is random with respect to both sample instance and realisation.

X_n and Y_n are constants with respect to time uniformly distributed constants between 0 and 1 with $E(X_n) = E(Y_n) = 0.5, \sigma_{X_n} = \sigma_{Y_n} = \sqrt{Var(X_n)} = \sqrt{Var(Y_n)} = \frac{1}{2\sqrt{3}}$

$Z_{t,n}$ is a uniformly distributed random variable with $E(z_n) = 0, \sigma_{z_n} = \sqrt{Var(Z_n)} = \frac{1}{2\sqrt{3}}$

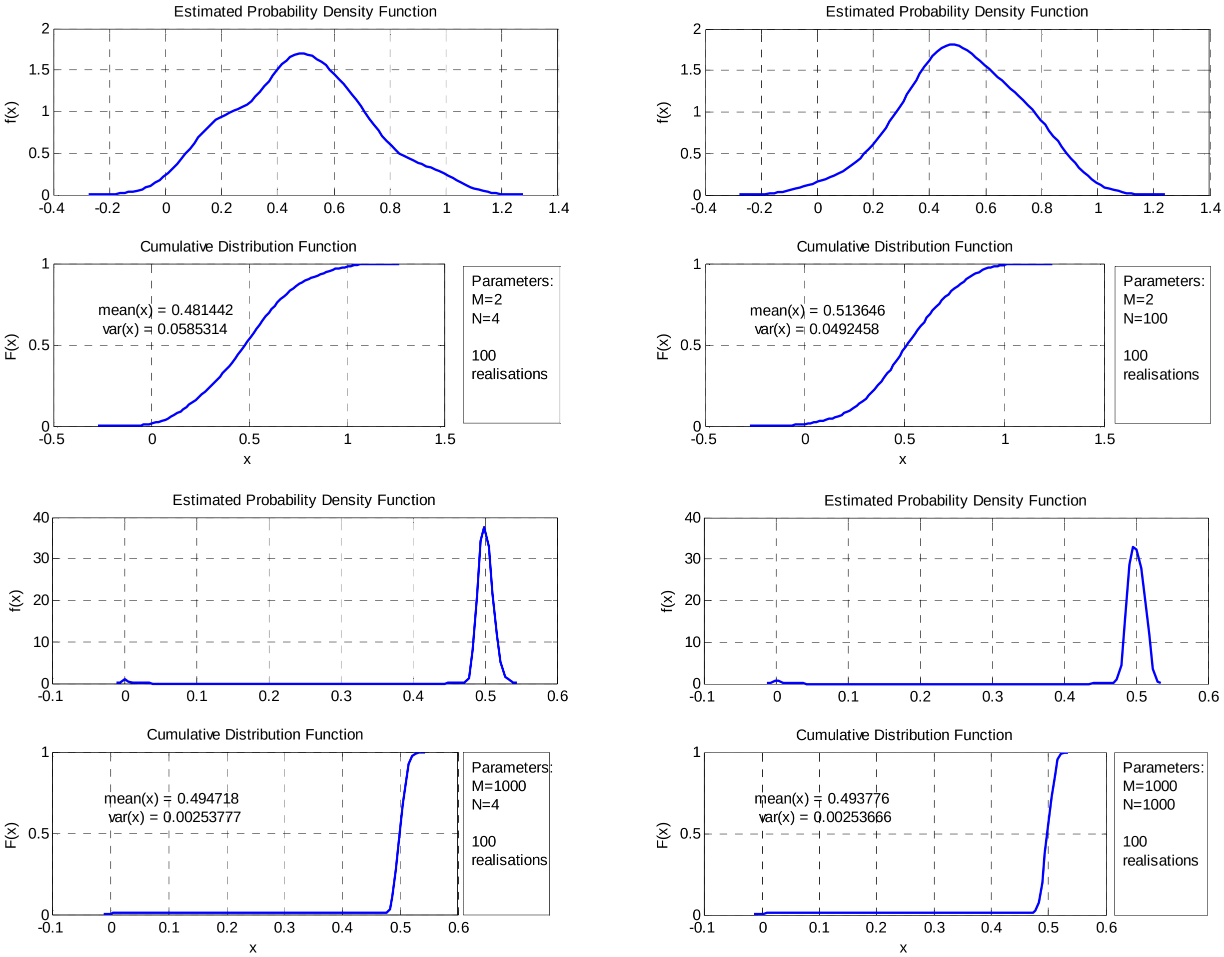
Thus the expectation with respect to time.

$$E(v_t) = E(X_n) + E(Y_n) \times E(Z_{t,n}) = E(X_n) + Y_n \times 0 = E(X_n) = X_n \rightarrow 0.5$$

Which backs up the findings that the rp2 process had a random expectation which in turn, over many realisations, has an average of 0.5. To illustrate this we have the graphs of the pdf of the mean of rp1 over many separate realisations. The code to make the following graphs is:

```
for i = 1:100;
a = rp2(M,N);
b(length(b)+1) = mean(mean(a));
end
```

```
gkdeb(b) %function written by Yi Cao (Cranfield University) to estimate pdf of input
%http://www.mathworks.co.uk/matlabcentral/fileexchange/19121-probability-density-function-pdf-estimator-v3-2/content/gkdeb.m
```



As we can see that while the mean is always around 0.5 the distribution is very different according to the number of instances per time (M).
The standard deviation of rp2 is:

$$Var(v_t) = E(v_t^2) - E(v_t)^2 = E\left(X_n^2 + 2X_nY_n \times Z_{t,n} + Y_n^2 \times Z_{t,n}^2\right) - X_n^2 = E(X_n^2) + E(Y_n^2)E\left(Z_{t,n}^2\right) - X_n^2$$
$$= \frac{1}{3} + \frac{1}{3} \cdot \frac{1}{12} - \frac{1}{4} = \frac{1}{9}$$

Note: to find $E(X_n^2)$ and $E(Y_n^2)$ we can use $Var(X_n) = E(X_n^2) - E(X_n)^2 \to \frac{1}{12} + \frac{1}{4} = \frac{1}{3} = E(X_n^2)$

$$\sigma_{v_t} = \sqrt{Var(v_t)} = \frac{1}{\sqrt{9}} = \frac{1}{3}$$

Which is exactly what is observed.

1.2.3.3 Process rp3

We define the rp3 process as
$v_t = m \cdot X + a$, where $X \sim \mathcal{U}(-\frac{1}{2}, \frac{1}{2})$ and $m = 3$ and $a = 0.5$
The expectation is

$$E(v_t) = m \cdot E(X) + a = a = \frac{1}{2}$$

And the standard deviation is:

$$Var(v_t) = E(v_t^2) - E(v_t)^2 = E(m^2X^2 + a^2 + 2 \cdot amX) - \frac{1}{4} = \frac{9}{12}$$
$$\sigma_{v_t} = \sqrt{Var(v_t)} = \frac{3}{2\sqrt{3}} \cong 0.86$$

Which is what is observed.

1.3 Estimation of probability distributions

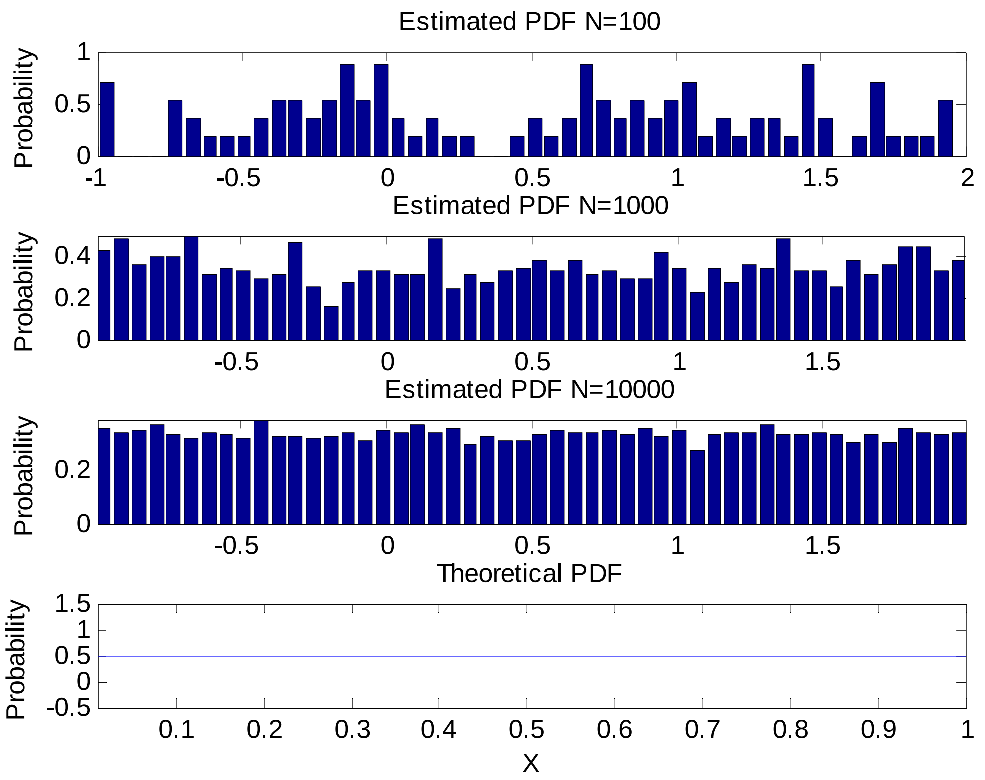
1.3.1 Matlab PDF estimation

The following function was written to estimate the probability density function of input samples:

```
N=1000;
v=randn(1,N);
[a,b]=hist(v,50);
figure(1)
a=a/trapz(b,a);%more accurate than dividing by N
%as dividing by N is true only if the bin size is small
%relative to the variance of the data
bar(b,a);
xlabel('X')
ylabel('Probability')
title('Estimated PDF')
```

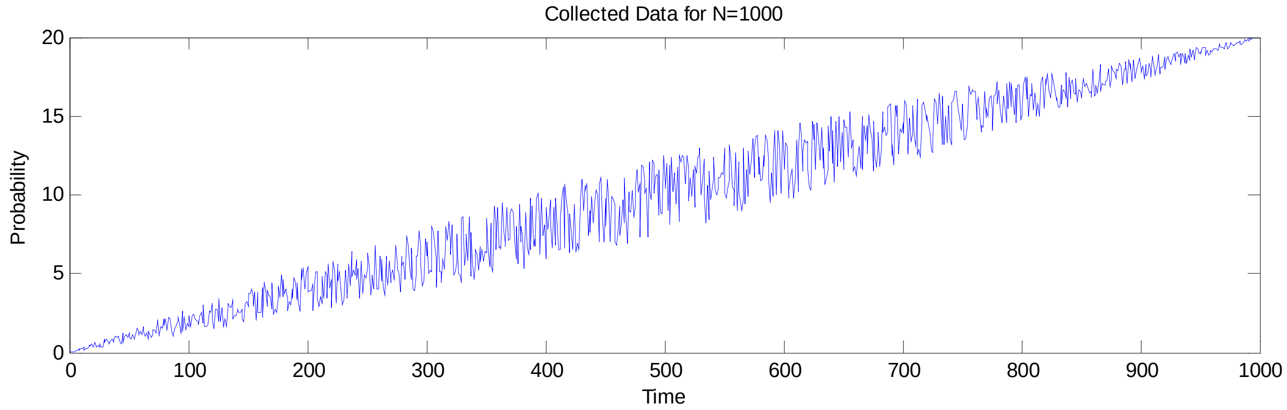
1.3.2 Comparison of theoretical and obtained PDFs

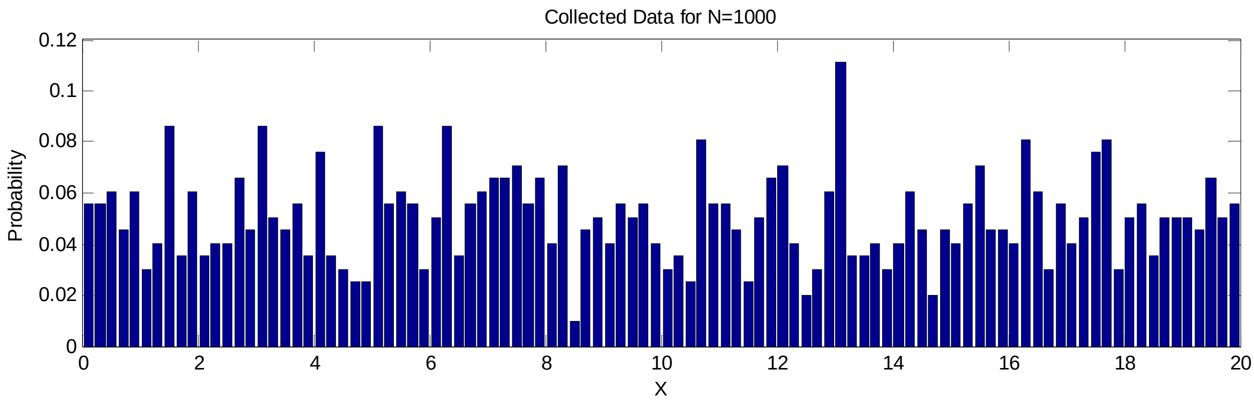
The only stationary and ergodic process is rp3 and as such will be the only investigated.



1.3.3 Estimation of non-stationary process

Estimating a nonstationary process such as rp1 using the hist function is difficult. This is what is obtained if this is attempted on rp1:





As we can see the PDF implies that the function amplitude is randomly generated between 0 and 20. However from the graph above we know that this is not the case due to this not being a stationary process. As such using a hist function removes this critical information and causes a loss of information. (This is information which would not be contained in a stationary process explain why such processes are uniquely defined by a time-invariant pdf.).

Appendix

Matlab Code

Part 1

```
clc;
a=0;
x = randn(1000,1);
[a,b] = hist(x);
a = a./1000;
%b=-.9:0.2:.9;
sprintf('mean:%.6f',mean(x))
sprintf('std:%.6f',std(x))
bar(b,a)
```

Part 1.2

```
clc;
a = rp3(200,400);

subplot(2,1,1); plot(mean(a))
title('Mean')
subplot(2,1,2); plot(std(a))
title('Standard Deviation')
```

```
b=0;
c=0;
d=0;
for i = 1:1000;
a = rp3(200,400);
b(length(b)+1) = mean(mean(a));
c(length(c)+1) = mean(std(a));
end
figure(2)
gkdeb(c) %function written by Yi Cao (Cranfield University) to estimate pdf of input
%http://www.mathworks.co.uk/matlabcentral/fileexchange/19121-probability-density-function-pdf-estimator-v3-
2/content/gkdeb.m
```

Part 1.3

```
clc;
N=100;
%v=randn(1,N);
v=rp3(1,N);
u=rp3(1,N*10);
w=rp3(1,N*100);
figure(1)
gkdeb(v);
[a,b]=hist(v,50);
figure(2)
a=a/trapz(b,a);%more accurate than dividing by N
%as dividing by N is true only if the bin size is small
%relative to the variance of the data
subplot(4,1,1)
axis tight
bar(b,a);
ylabel('Probability')
title('Estimated PDF N=100')
%%%%%%%%%%%
subplot(4,1,2)
[a,b]=hist(u,50);
a=a/trapz(b,a);
bar(b,a);
axis tight
ylabel('Probability')
title('Estimated PDF N=1000')
%%%%%%%%%%%
subplot(4,1,3)
[a,b]=hist(w,50);
a=a/trapz(b,a);
axis tight
bar(b,a);
axis tight
ylabel('Probability')
title('Estimated PDF N=10000')
%%%%%%%%%%%
subplot(4,1,4)
x=0.01:0.01:1;
y = ones(1,100).*1/2;
plot(x,y)
axis tight
xlabel('X')
ylabel('Probability')
title('Theoretical PDF')
```

```
clc;
%%%%%%%%%
v=rp1(1,1000);
plot(v)
[a,b] = hist(v,100);
a = a./trapz(b,a);
%bar(b,a);
xlabel('Time')
ylabel('Probability')
title('Collected Data for N=1000')
%%%%%%%%%
% subplot(2,1,2)
% v=rp1(1,1000);
% [a,b] = hist(v,100);
% a = a./trapz(b,a);
% bar(b,a);
% xlabel('X')
% ylabel('Probability')
% title('Collected Data for N=1000')
```