Imperial College London

Imperial College London

Department of Electrical and Electronic Engineering

ELEC96002 Advanced Digital Signal Processing

Prof. Danilo P. Mandic

Coursework Report

submitted by

Costanza Gulli

CID 01352733

Contents

1 Random signals and stochastic processes

1.1 Statistical estimation

1.1.1 Uniform distribution

Figure 1 is a plot of 1000 random realisations of the uniform random variable $X \sim \mathcal{U}(0,1)$. The pdf of the random variable is described in Equation 1. From the samples, the sample mean $\hat{\mu}_X$ and sample standard deviation $\hat{\sigma}_X$ are obtained. These are calculated with the matlab functions `mean` and `std`, according to the formulas in Equations 2 and 3.

$$f_X(x) \sim \mathcal{U}(0,1) = \begin{cases} 1 & 0 \leq x \leq 1 \\ 0 & otherwise \end{cases} \tag{1}$$

$$\hat{\mu}_X = \frac{1}{N}\sum_{n=1}^{N} x[n] \tag{2}$$

$$\hat{\sigma}_X = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N}(x[n]-\mu_x)^2} \tag{3}$$

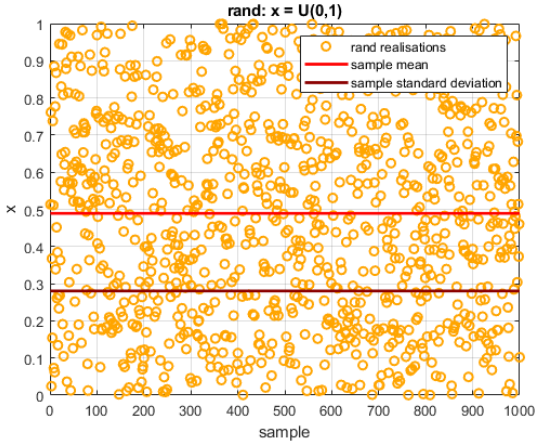

Figure 1: 1000 realisations of the random variable X;
their sample mean and standard deviation

After this, a set of 10 realisations, each of 1000 samples is obtained. For each realisation, the sample mean and sample standard deviation is calculated, and compared with the theoretical value. As it can be noticed from Figure 2, when considering the mean and standard deviation of different realisation, each comprising of a large enough number of samples (in this case N = 1000), the sample values are very close to the theoretical ones. In fact, it can be noticed from Figure 3 that the estimated values asymptotically converge to the theoretical ones as the sample size increases. In the plots, the error in sample mean estimation is calculated as $e = \mu_X - \hat{\mu}_X$, and the error in sample standard deviation estimation is calculated as $e = \sigma_X - \hat{\sigma}_X$.

The theoretical mean and standard deviation are calculated from the pdf, as follows in Equations 4 and 5.

$$\mu_X = \int_{-\infty}^{\infty} x\, f_X(x)\, \mathrm{d}x = \int_0^1 x\, \mathrm{d}x = \left[\frac{x^2}{2}\right]_0^1 = \frac{1}{2} \tag{4}$$

$$\sigma_X = \sqrt{\mathbb{E}\{X^2\} - \mathbb{E}\{X\}^2} \tag{5}$$

$$\mathbb{E}\{X^2\} = \int_{-\infty}^{\infty} x^2\, f_X(x)\, \mathrm{d}x = \int_0^1 x^2\, \mathrm{d}x = \left[\frac{x^3}{3}\right]_0^1 = \frac{1}{3}$$

$$\mathbb{E}\{X\}^2 = \mu_X^2$$

$$\sigma_X = \sqrt{\mathbb{E}\{X^2\} - \mathbb{E}\{X\}^2} = \sqrt{\frac{1}{3} - \frac{1}{2^2}} = \sqrt{\frac{1}{12}}$$

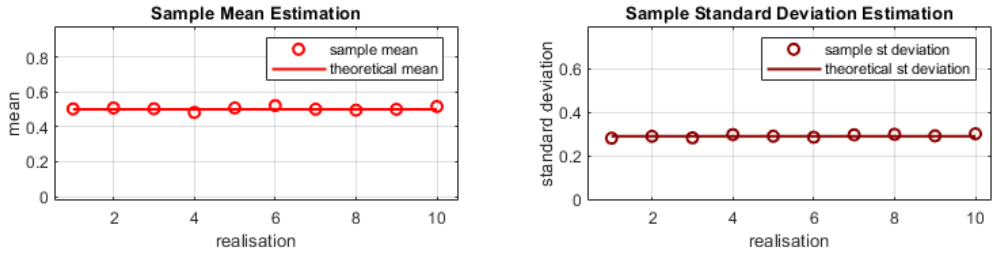

Figure 2: Sample mean and standard deviation estimation for 10 realisations of 1000 samples, compared with the theoretical value

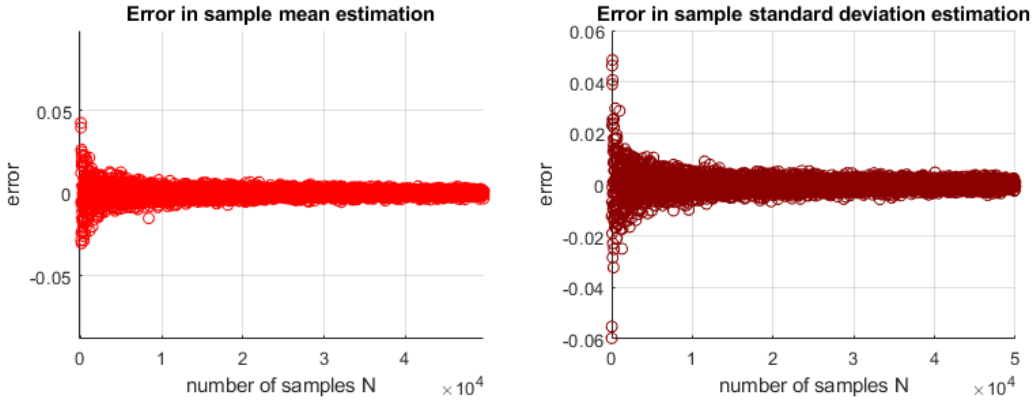

Figure 3: Error in mean and standard deviation estimation as a function of number of samples N

Afterwards, the samples are used to approximate the pdf from which the samples are drawn. This is done by using the `hist` matlab function. The result is plotted using `bins`. The pdf estimation is obtained varying both the number of bins and the number of samples. Two main phenomena can be noticed.

First of all, having more bins results in a better resolution in the pdf approximation. Secondly, the number of samples per bin has to be sufficient in order to obtain meaningful results. In subplot 2, there are 10 samples per bin. This leads to unexpected behaviour due to the random nature of the distribution from which the samples are drawn.

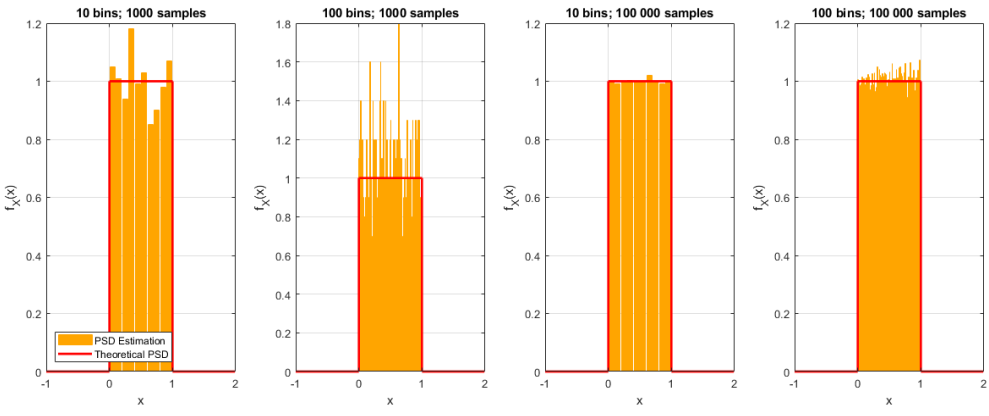

Figure 4: PDF estimation, varying the sample size and number of bins

1.1.2 Normal distribution

The same analysis is conducted with samples drawn from a standard normal distribution. The pdf is given in Equation 6. 1000 samples are drawn from this distribution. The plot is shown in Figure 5, together

with the sample mean and standard deviation. As above, these are obtained according to equations 2 and 3.

$$f_X(x) \sim \mathcal{N}(0,1) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} \tag{6}$$

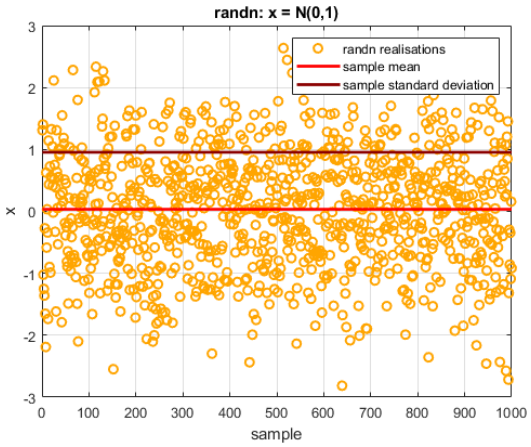

Figure 5: 1000 realisations of the random variable X;
their sample mean and standard deviation

Again, 10 realisations are obtained and their sample mean and standard deviations are compared with the theoretical values (Figure 6). The mean of a standard normal distribution is $\mu_X = 0$ and the standard deviation is $\sigma_X = 1$.

It can be noticed that the values obtained from the samples are less precise than those of the uniform distribution in section 1.1.1. This is because for the normal distribution the distribution is more spread around the mean, leading to higher uncertainty on the output. To obtain more accurate results, we would need to calculate the values from a higher number of samples N.

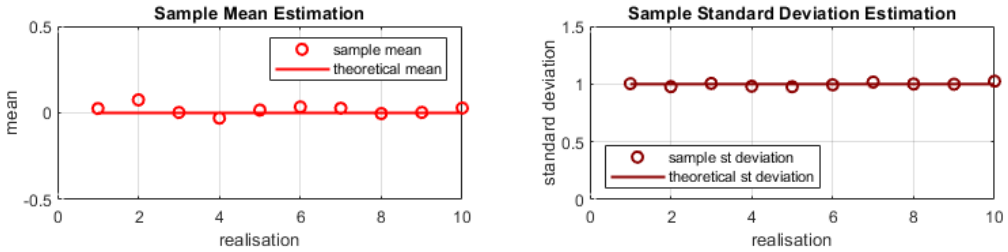

Figure 6: Sample mean and standard deviation estimation for 10 realisations of 1000 samples, compared with the theoretical value

When approximating the pdf of the distribution from which the samples are drawn, similar results to those obtained in section 1.1.1 are noticed (Figure 7). Subplot 4 shows that, more than in the uniform distribution, having more bins increases the resolution in approximating the pdf. However, subplot 2 shows that the number of samples per bin has to be higher than ~ 100 to avoid unpredictable behaviour. In fact, subplots 1 and 3 are really precise despite having very low resolution.

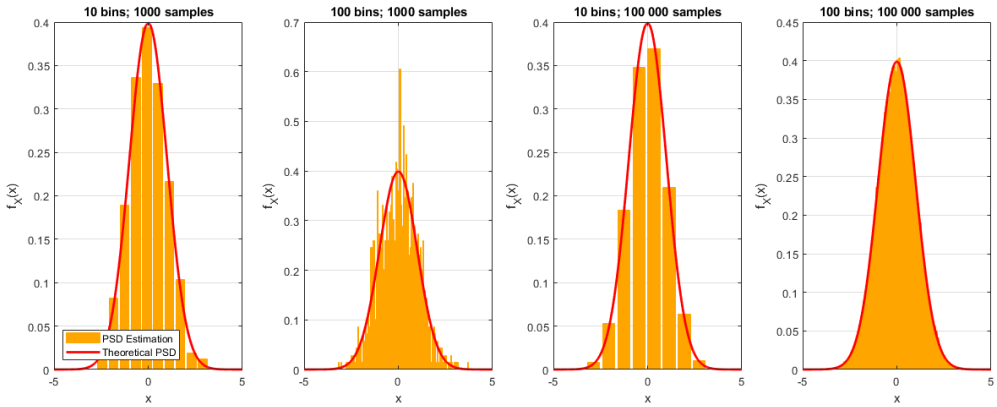

Figure 7: PDF estimation, varying the sample size and number of bins

1.2 Stochastic processes

This section analyses the difference between ensemble and time means and standard deviations. These characteristics are than used o discuss the stationarity and ergodicity of random processes. In the following sections, signals drawn from 3 probability distributions are analysed.

1.2.1 Random process 1

The first signal is drawn from *random process 1*, defined by the following matlab function.

```
function v=rp1(M,N)
a=0.02;
b=5;
Mc=ones(M,1)*b*sin((1:N)*pi/N);
Ac=a*ones(M,1)*[1:N];
v=(rand(M,N)-0.5).*Mc+Ac;
```

The output is a MxN matrix, where M is the number of realisations and N is the number of samples per realisation.

1.2.1.1 Ensemble mean and standard deviation

M = 100 realisations of N = 100 samples of the random process are are obtained. The ensemble mean and standard deviation are obtained by averaging the values for all realisation at a given time. The results are plotted as a function on time in Figure 8. More accurate values are obtained by averaging M = 100,000 realisations (dotted lines).

Both mean and standard deviation vary as a function of time. Therefore, the signal is **not stationary**.

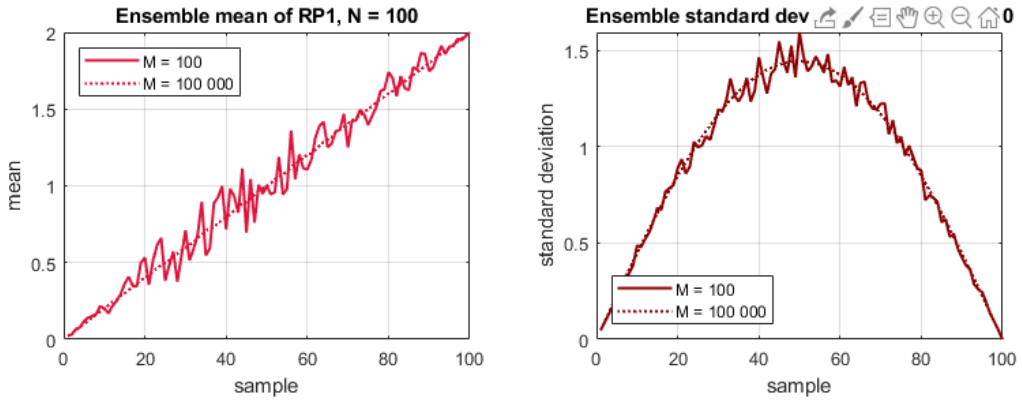

Figure 8: Ensemble mean and standard deviation of RP1

1.2.1.2 Mean and standard deviation for each realisation

The mean and standard deviations are also obtained by averaging the values over time for each realisation. The time mean and standard deviation for each realisation are plotted in Figure 9. The values are constant over time, but the signal is non-stationary, therefore is also **non-ergodic**.

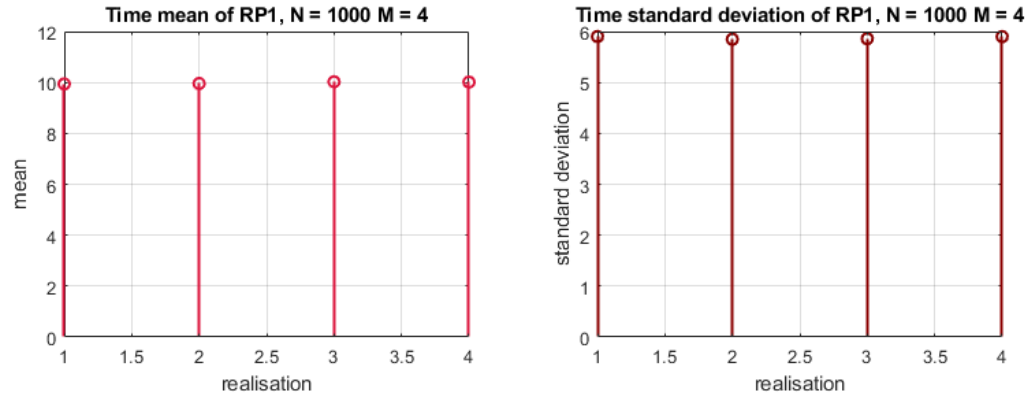

Figure 9: Time mean and standard deviation of RP1

1.2.1.3 Theoretical mean and variance

The mathematical description for signal A, drawn from RP1, is given below.

$$f_A(n) = (u(n) - 0.5) * b * \sin(\frac{n\pi}{N}) + a * n = u_1(n) * \sin(\frac{n\pi}{N}) + 0.02n$$

$$u_1(n) \sim \mathcal{U}(-2.5, 2.5)$$

The expectation is calculated as follows:

$$\mu_A = \mathbb{E}\{u_1(n) * \sin(\frac{n\pi}{N})\} + \mathbb{E}\{0.02n\} = 0.02n$$

It can be noticed that the result matches the data in Figure 8. In fact, the mean is a straight line through the origin, with value of 2 at n = 100.

The variance is calculated from the signal's equation:

$$\sigma_A^2 = \mathbb{V}ar\{f_A(x)\} = \mathbb{E}\{(f_A(x) - \mu_A)^2\} = \mathbb{E}\{((u(n) - 0.5)5\sin(\frac{n\pi}{N}) + 0.02n - 0.02n)^2\} =$$

$$= \mathbb{E}\{(u(n) - 0.5)^2 25\sin^2(\frac{n\pi}{N})\} = 25\sin^2(\frac{n\pi}{N})\mathbb{E}\{(u(n) - 0.5)^2\} = 25\sin^2(\frac{n\pi}{N})\mathbb{V}ar\{u(n)\} = \frac{25}{12}\sin^2(\frac{n\pi}{N})$$

The standard deviation is therefore $\sigma_A = \frac{5}{\sqrt{12}}\sin(\frac{n\pi}{N})$. This matches the estimated standard deviation in Figure 8. In fact, this is a sinusoidal signal with period $\approx 2N$ and peak at ≈ 1.44, which is the approximate value of $\frac{5}{\sqrt{12}}$.

1.2.2 Random Process 2

The signal B is drawn from *random process 2*, defined by the following matlab function.

```
function v=rp2(M,N)
Ar=rand(M,1)*ones(1,N);
Mr=rand(M,1)*ones(1,N);
v=(rand(M,N)-0.5).*Mr+Ar;
```

1.2.2.1 Ensemble mean and standard deviation

Figure 10 shows that the mean and standard deviation are constants over time. The signal is **stationary**.

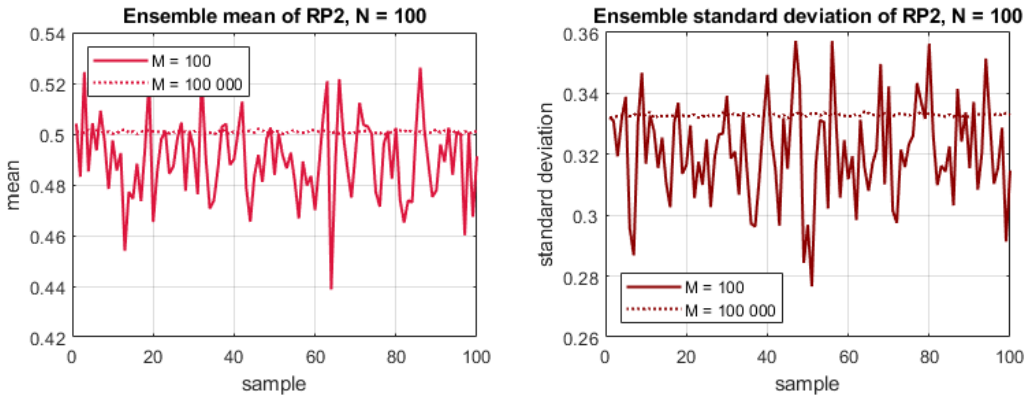

Figure 10: Ensemble mean and standard deviation of RP2

1.2.2.2 Mean and standard deviation for each realisation

It can be noticed from Figure 11 that the time mean and standard deviation varies across realisations. Therefore the signal is **non-ergodic**.

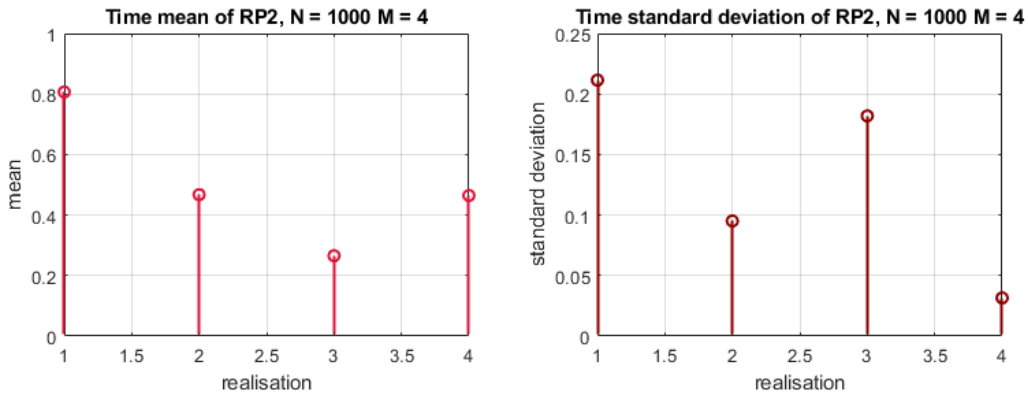

Figure 11: Time mean and standard deviation of RP2

1.2.2.3 Theoretical mean and variance

The signal's equation is as follows:

$$f_B(n) = u(n) * (u(n) - 0.5) + u(n) = u(n) * u_2(n) + u(n)$$

$$u_2(n) \sim \mathcal{U}(-0.5, 0.5)$$

The theoretical mean and variance are obtained from the above equation. Note that the calculations exploit the fact that $u(n), u_2(n)$ and $u(n)$ are independent random variables.

$$\mu_B = \mathbb{E}\{u(n) * u_2(n) + u(n)\} = \mathbb{E}\{u(n)\} * \mathbb{E}\{u_2(n)\} + \mathbb{E}\{u(n)\} = 0.5 * 0 + 0.5 = 0.5$$

$$\sigma_B^2 = \mathbb{V}ar\{u(n) * u_2(n) + u(n)\} = \mathbb{V}ar\{u(n) * u_2(n)\} + \mathbb{V}ar\{u(n)\} =$$
$$= \mathbb{E}\{u^2(n)\}\mathbb{E}\{u_2^2(n)\} - \mathbb{E}\{u(n)\}^2\mathbb{E}\{u_2(n)\}^2 + \mathbb{E}\{u^2(n)\} - \mathbb{E}\{u(n)\}^2 =$$
$$= \int_0^1 x^2 \, \mathrm{d}x \; * \int_{-0.5}^{0.5} x^2 \, \mathrm{d}x - 0.5^2 * 0^2 + \int_0^1 x^2 \, \mathrm{d}x - 0.5^2 =$$
$$\left[\frac{x^3}{3}\right]_0^1 * \left[\frac{x^3}{3}\right]_{-0.5}^{0.5} + \left[\frac{x^3}{3}\right]_0^1 - 0.5^2 = \frac{1}{9}$$

The value of the mean is confirmed by the data in Figure 10. Note that the value asymptotically tends towards the theoretical as the number of realisations increases. The standard deviation is obtained as the square root of the variance, $\sigma_B = \sqrt{\sigma_b^2} = \frac{1}{3} \approx 0.33$. This is confirmed by the value obtained empirically in Figure 10.

1.2.3 Random Process 3

The last signal is drawn from *random process 3*, defined by the following matlab function.

```
function v=rp3(M,N)
a=0.5;
m=3;
v=(rand(M,N)-0.5)*m + a;
```

1.2.3.1 Ensemble mean and standard deviation

The ensemble mean and standard deviation are both constant over time (Figure 12). The signal is therefore **stationary**.

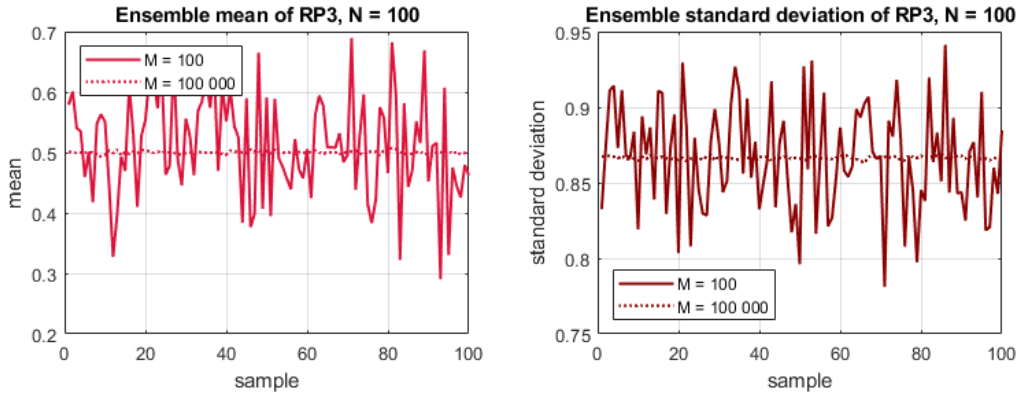

Figure 12: Ensemble mean and standard deviation of RP3

1.2.3.2 Mean and standard deviation for each realisation

Figure 13 shows that the time mean and standard deviation do not vary across realisations. Therefore, the signal is **ergodic**.

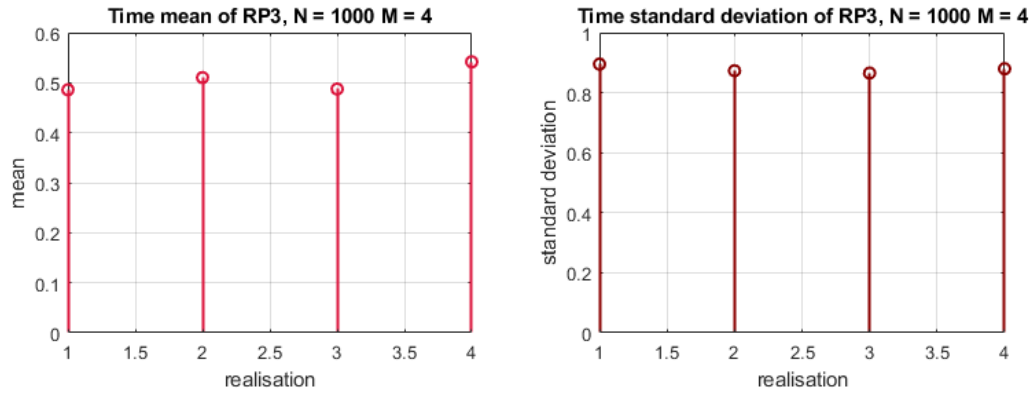

Figure 13: Time mean and standard deviation of RP3

1.2.3.3 Theoretical mean and variance

Signal C is a random signal uniformly distributed $s_C(n) \sim \mathcal{U}(-1, 2)$.

$$f_C(n) = (u(n) - 0.5) * m + a = u_3(n) \sim \mathcal{U}(-1, 2)$$

Its mean and variance are calculated as follows:

$$\mu_C = \mathbb{E}\{u_3(n)\} = \int_{-\infty}^{\infty} x\, f_C(x)\, \mathrm{d}x = \int_{-1}^{2} x \frac{1}{3}\, \mathrm{d}x = \left[\frac{x^2}{6}\right]_{-1}^{2} = \frac{1}{2} = 0.5$$

$$\sigma_C^2 = \mathbb{V}ar\{(u(n) - 0.5) * m + a\} = m^2 \mathbb{V}ar\{u(n)\} = \frac{3^2}{12} = \frac{3}{4}$$

The standard deviation is therefore $\sigma_C = \sqrt{\frac{3}{4}} \approx 0.866$.

Notice that the theoretical predictions of mean and standard deviation match the data obtained both from the ensemble and the time data. This confirms that the signal is stationary and ergodic.

1.3 Estimation of probability distributions

The code shown below is a matlab function implemented to estimate PDFs. The input to the function is `x`: this is a vector containing the samples of the signal. The function will estimate the PDF from which these samples were drawn.

The outputs of the function are two vectors: `y` and `h_centers`. The pair of values at the same index in y and h_centers give the value and the relative probability of that value occurring.

```
function [y, h_centers] = pdf(x)
l = length(x);

if l ≥ 10000
    n_bins = 100;
elseif l ≤ 1000
    n_bins = 10;
else
    n_bins = round(l/100);
end

[h_counts, h_centers] = hist(x, n_bins);
y = h_counts./trapz(h_centers, h_counts);
end
```

The PDF estimation is done with the objective of finding a trade off between high resolution and high accuracy. Figure 14 shows why this is a relevant consideration to be taken into account:

- subplot 1 : if the number of bins is too low, the resolution is very poor;
- subplot 2: if the average number of samples per bin (10 in this case) is low, the accuracy of the estimate is very poor;
- subplot 3: if the number of bins is `n_bins` ≥ 100 and the average number of samples per bin is ≥ 100, the estimate is really close to the theoretical PDF;
- subplot 4: as the number of samples per bin increases, the estimate asymptotically converges towards the theoretical PDF.

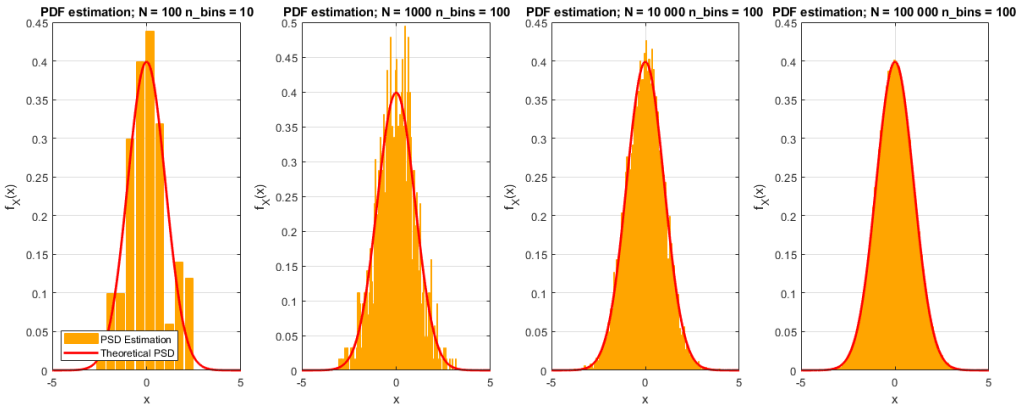

Figure 14: PDF estimation accuracy, varying N and n_bins

As a consequence, the code calculates the number of bins in the estimate `n_bins` based on the number of samples (`length(x)`): this is the desired resolution for the estimation.

Note that in the code (line 13), the number of samples counted in each interval is normalised so that the total area under the curve is equal to 1.

Figure 15 shows that the function `pdf` successfully estimates the PDF of a standard normal random variable for any number of samples, asymptotically providing a better estimate as N increases.

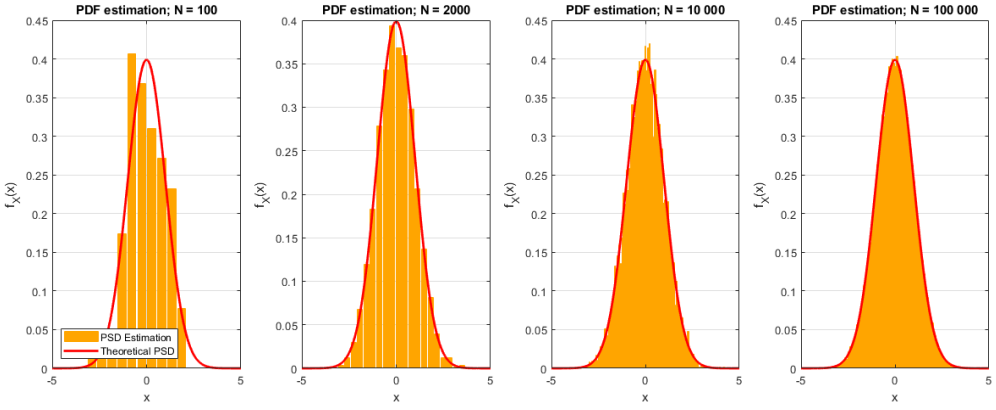

Figure 15: PDF estimation with the matlab function pdf, varying N

1.3.1 Estimation of stationary ergodic signals

The function is tested by estimating the PDF of the *random process 3* from section 1.2.3. Figure 16 shows that the function is indeed able to asymptotically provide an estimate of the distribution as N increases. The estimated PDF converges towards the theoretical, whose mathematical expression follows:

$$f_C(x) \sim \mathcal{U}(-1,2) = \begin{cases} \frac{1}{3} & -1 \leq x \leq 2 \\ 0 & otherwise \end{cases}$$

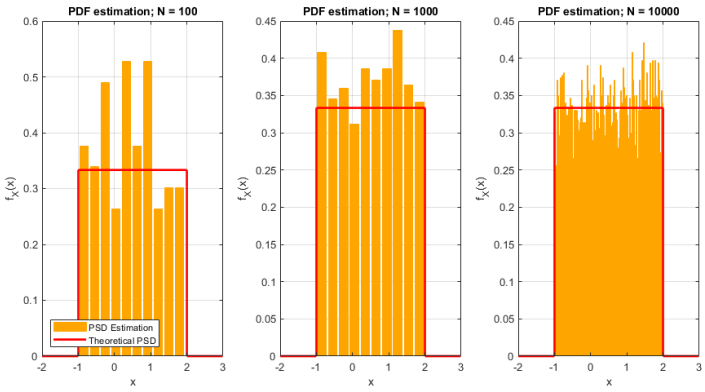

Figure 16: PDF estimation of RP3, varying N

1.3.2 Estimation of non-stationary signals

The ability of the function to estimate the PDF of a non-stationary process is tested. The function is applied with an input vector `v`. `v(1:500)`$\sim \mathcal{U}(-0.5, 0.5)$, `v(501:1000)`$\sim \mathcal{U}(0.5, 1.5)$. As it can be noticed, the function fails to estimate the PDF. This is because the matlab function `hist` inherently uses a time average of the signal to produce an estimate. This compromises the ability of the function to discriminate between different distributions.

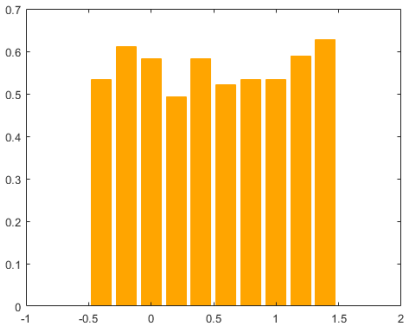

Figure 17: Wrong PDF estimation of the non-stationary signal

The correct PDF could be computed by separating the signal into v_1 = `v(1:500)` and v_2 = `v(501:1000)`, and separately estimating the PDFs. As it can be seen from Figure 18, this enables to correctly identify hte distributions. This is because the non-stationary signal `v` is separated into two stationary signals v_1 and v_2.

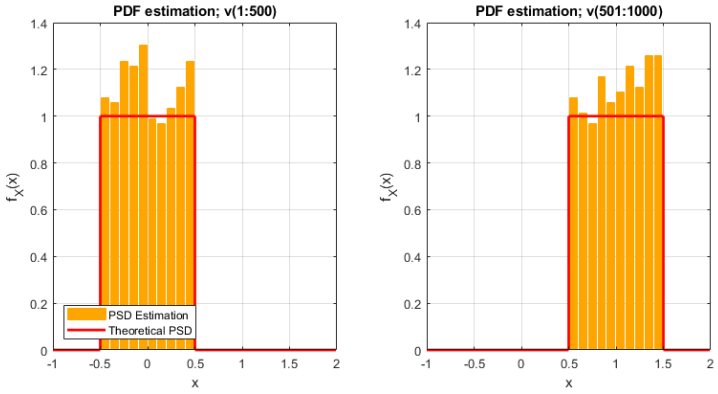

Figure 18: Correct PDF estimation of the non-stationary signal