INF4480 Project I: Random processes

Roy Edgar Hansen

January 17, 2021

Distribution date	January 15, 2021
Deadline	January 26, 2021, 10:15
Deliver presentation in	`Devilry`

Introduction

Remember to read the manual pages on the matlab commands before using them.

1 Random variables

1.1 Random variables - uniform PDF

Generate a long vector x of samples of a uniform random variable using the matlab function `rand`. Use several thousand samples.

Estimate the probability density function (PDF), the first order moment (or the mean value) $m_x = E\{x\}$, and the second order central moment (or the variance) $\sigma_x^2 = E\{(x - m_x)^2\}$ using the matlab functions `histogram` or `histcounts`, `mean` and `std`. Do not use the `histogram`-function with `'Normalization'` set to `'pdf'` or `'probability'`.

Make sure that the estimated PDF is properly normalized - check that the properties of the PDF is fulfilled (see lecture notes 1). Plot the estimated PDF. Plot the theoretical PDF on top in the same figure in a different colour. Use the matlab command `hold on` to put several plots in the same figure.

Derive the theoretical values for the mean m_x and the variance σ_x^2. That is, solve the integral formula for expectation of the random variable and the second order central moment. Use the fact that matlabs function `rand` produces a uniform density in the range $[0, 1]$.

Repeat the numerical experiment several times and observe. What happens to to the estimates of the pdf, m_x and σ_x^2 ?

1.2 Random variables - Gaussian PDF

Generate a long vector x of samples of a Gaussian random variable using the matlab function `randn`. Use several thousand samples.

Estimate the probability density function (PDF), the first order moment $m_x = E\{x\}$ and the second order central moment (or the variance) $\sigma_x^2 = E\{(x - m_x)^2\}$ using the same approach as in exercise 1.1. Make sure that the estimated PDF is properly normalized. Compare the estimated values with the true value of the mean and the variance.

Plot the estimated PDF. Plot the theoretical PDF is the same figure in a different colour. The formula for the Gaussian PDF is known:

$$f_x(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left(-\frac{(x - m_x)^2}{2\sigma_x^2}\right)$$

1.3 The Central Limit Theorem

Use `rand(12,N)` to produce 12 independent uniform random vectors of length N. Average over each column using the matlab command `mean`. The result is a single random vector of length N.

Estimate the PDF from this new random variable using the histogram technique as in exercise 1.1 and 1.2. Remember normalization. Also estimate the mean and the variance.

Derive the theoretical values for the mean and the variance for the new random variabe (based on the sum). Note that the matlab function `rand` produces a uniformly distributed random variable with nonzero mean value. Hint: Assume statistical independence to simplify.

Compare the estimated mean and variance from exercise with the theoretical values derived. Plot the estimated PDF. Plot the theoretical PDF for a Gaussian random variable superimposed.

2 Stationarity and ergodicity

In this exercise we consider three random processes. The following code generates M realizations and N time samples of each random process.

```
function [ v ] = rp1(M,N)    %<<------ RANDOM PROCESS #1
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;

function [ v ] = rp2(M,N)    %<<------ RANDOM PROCESS #2
Ar = rand(M,1)*ones(1,N);
```

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

function [ v ] = rp3(M,N)   %<<------ RANDOM PROCESS #3
a = 0.5;
m = 3;
v = ( rand(M,N)-0.5 )*m + a;
```

2.1 Stationary or Ergodic?

Use the code above to generate four members $M = 4$ of each process of length $N = 100$. Display each of the random processes `rp1`, `rp2` and `rp3` in a matlab figure using subplot. Overlay each realization (M variable) in different colours using the matlab command `hold on`.

Decide by inspection whether the each process is ergodic and/or stationary.

2.2 Ensemble Averages

Compute the ensemble mean and standard deviation for each of the processes. Use a fairly large number of realizations $M = 80$, and use the matlab commands `mean` and `std` to estimate the mean and the standard deviation.

Plot the results versus time N. Use subplot to generate two plots - one for the mean value and one for the standard deviation. Overlay plots for each of the random processes `rp1`, `rp2` and `rp3` in different colours.

2.3 Time Averages

Estimate the time averages of four different realizations (or members) $M = 4$ of each process. Use a fairly large number of time samples $N = 1000$.

Decide which of the processes are ergodic (again). Hint: A nonstationary process cannot be ergodic.

Plot all data from each random process as a twodimensional image using the matlab command `imagesc`. Let the x-axis be the time axis and the y-axis be the realization axis. Can this plot be used to decide if the random process is stationary and/or ergodic? If so, how?