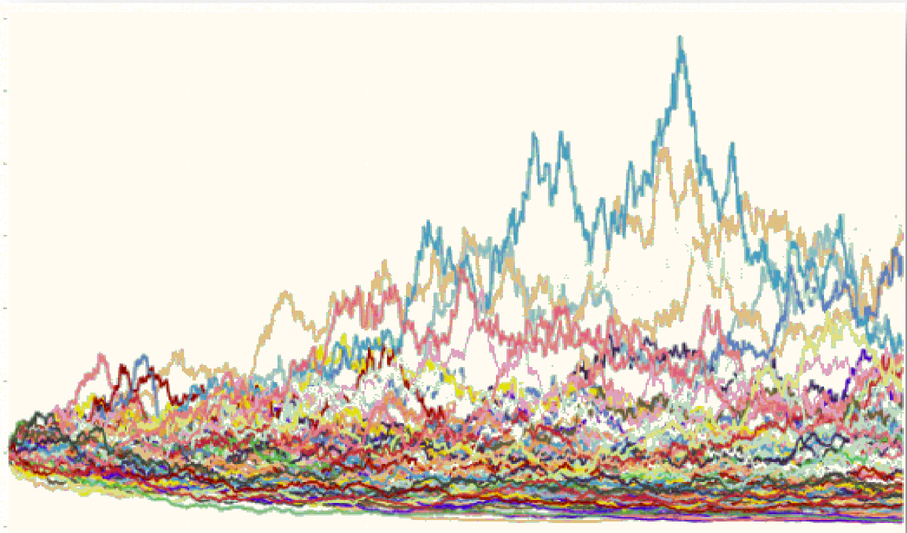
Introduction to Audio-visual Processing

Lab 2: Statistical Signal Modelling II

Stochastic Processes



Course 20-21/Q2

Contents

Introduction

The objective of this second Lab session is to consolidate the main concepts related with the statistical characterization of stochastic processes. The aim is to review concepts like realization of a stochastic process; mean, autocorrelation and variance of stochastic processes; stationary process; power spectral density; or ergodicity. In the Lab session, using your computers, you will synthetize and analyze different processes and random variables, that will allow you to understand, gradually, those concepts. During the Lab session, we will also present some random process models that are used frequently in applications.

Pre Lab activities:

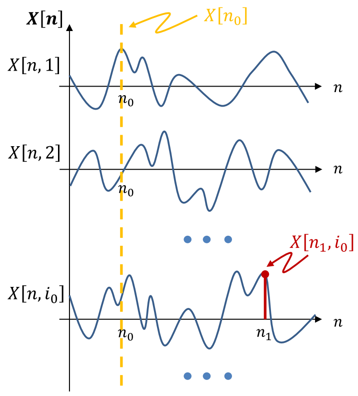
The aim of the Lab session is to review, consolidate and extend the basic concepts studied in class so, before this session, you should review Unit 1.3 Discrete Stochastic Processes.

1. Statistical characterization of stochastic processes

Different kinds of random processes are presented aiming to introduce the most common ones and their possible applications. Each activity proposes generating several realizations of a random process that will be saved in a matrix. Some results will be reused in other activities, so the recommendation is to name the matrices that model the random processes as SP# (with # the number of the process).

1.1. White Gaussian stochastic process (noise)

The objective of this part is twofold. First, to consolidate the concept of discrete stochastic (random) process and the definitions of *realization* of a random process, *random variable* associated to a time instant, and *instantaneous value*. Second, to introduce the concepts of *stationary process* and *pdf* of the process samples at a given time instant.



Process no. 1.

Using the function `randn()` MATLAB generates a sequence of normal (Gaussian) independent and identically distributed (i.i.d.) samples with zero-mean and unite variance. With this function, generate a vector of 16000 samples as follows: Noise1

```
n=[0:15999];
Noise1=randn(size(n));
```

The vector that you have generated could model 2 seconds of white Gaussian noise sampled at 8000Hz. Listen to the vector to understand that a realization of noise has been generated. Plot also the samples and, using the *Zoom In* tool, visualize the signal.

```
Fs=8000;
soundsc(Noise1,Fs);
plot(n/Fs,Noise1);
```

In vector `Noise1` you have a *realization* of the stochastic process that we could name: *zero-mean and unite variance white Gaussian noise*. Since white Gaussian noise is one of the most common types of noise (e.g.: in speech), a similar result could had been obtained if you had recorded 2 seconds of silence with your microphone. It is important to remark that once you get a *realization* of the noise (associated with a random experiment), you have a deterministic signal that can be processed using the tools that you learned in Signals and Systems course.

A stochastic process cannot be understood, neither analyzed, from a single realization. Multiple realizations (ideally infinite) of the stochastic process are needed to statistically characterize the process.

A first step to obtain more information of the random process could be to share with your colleagues the noise realization that you have generated. Alternatively, if you repeat the generation of the `Noise1` vector several times, you can also obtain independent[1] realizations of that process.

Implement this second strategy and generate M=2000 realizations of a white Gaussian noise process and save the different realizations in a matrix (`SP1` matrix), which will contain the information of the stochastic process. Saving each realization in a different row, `SP1` will result into an M×N matrix (M=2000, N=16000) where the i-th column will be a random variable associated to the value of the random process at time n_i. SP1

After generating the `SP1` matrix, answer the following questions:

- Using the `subplot()` function, plot four different realizations of the stochastic process. Describe, qualitatively, the aspect of the random process (range of values, fluctuations…). 1.1a
- Select an arbitrary time instant ni of the process and, using the MATLAB code given below, plot the histogram of the 2000 realizations of the random variable X[ni] using 100 bins, overlap the exact pdf $f_x(x;n)$ and compare the histogram and the pdf. Does the histogram fit the pdf?

```
figure
hist(SP1(:,ni),100);
DeltaX=(max(SP1(:,ni))-min(SP1(:,ni)))/100;
K1=DeltaX*size(SP1,1);
GaussPDF=normpdf([-4:0.01:4],0,1); % Exact pdf
K2=0.01*sum(GaussPDF);
hold on
plot([-4:0.01:4],GaussPDF*K1/K2,'r')
```

  - Does the histogram fit the pdf?
  - Review the MATLAB code and justify why K1 and K2 constants are calculated and why the exact pdf `GaussPDF` is scaled by K1/K2. 1.1b

Process no. 2.

Let us now generate a different Gaussian process using the following MATLAB code: Noise2

```
n=[0:15999];
A=exp(-(n/8000-1).^2/0.1);
Noise2=sqrt(A).*randn(size(A));
```

As you did for the first process, listen `Noise2` vector and plot its samples.

Which is the main characteristic of this new process and the main difference with respect to the previous one? 1.1c

---

1 *If you synthetize a stochastic process with your computer, it is important to be sure that the random number generator that your software uses generates independent random numbers. Otherwise we cannot affirm that we are generating independent realizations of the process.*

Generate M=2000 realizations of this new process and save them in a matrix (`SP2` matrix). After generating the `SP2` matrix, answer the following questions: SP2

- Using the `subplot()` function, plot four different realizations of the stochastic process. Describe, qualitatively, the aspect of the random process (range of values, fluctuations…). 1.1d
- Select the 5000th sample of the process and plot its histogram using 100 bins `hist(SP2(:,5000),100)`. As you did for process 1, overlap the pdf $f_x(x; n)$ using the same MATLAB code. Does the histogram fit the pdf? 1.1e

As we have seen in class, the expected value $E\{\cdot\}$ is used to calculate the first and second order moments of a stochastic process (mean, variance, correlation and covariance).

- Justify that, when multiple realizations of the process are available, the statistical operator $E\{\cdot\}$ can be calculated numerically computing a long-run average value of repetitions of the same experiment. 1.1f
- According to the previous sentence, use the MATLAB functions `mean(SP#(:,ni))` and `var(SP#(:,ni))` to calculate, numerically, the mean and variance of the process at the time instant ni. Specifically, calculate numerically both moments for the two processes at two different time instants. Calculate, also, the theoretical values and compare the results. 1.1g

	Process no. 1		**Process no. 2**	
	Numerical	**Theoretical**	**Numerical**	**Theoretical**
Mean n=100				
Mean n=5000				
Variance n=100				
Variance n=5000				

- Are the mean and the variance constant with respect to time? Relate this result with the stationarity of the process. 1.1h
- A stochastic process is ergodic if the temporal mean (or autocorrelation) equals the statistical mean (or autocorrelation). Calculate the mean and variance of an arbitrary realization and say if the processes are ergodic. 1.1i

1.2. Colored Gaussian stochastic process (noise)

In some cases, given a stochastic process, another stochastic process can be obtained after filtering the initial one through an LTI system with impulse response h[n]. As it has been shown in class, an LTI system modifies the autocorrelation of the initial process. In consequence, this strategy can be used to generate stochastic processes with a certain autocorrelation. The objective of this part is to manipulate (filter) the white Gaussian noise process generated in section 1.1 (`SP1`), to consolidate the concepts of autocorrelation and memory of a stochastic process, and to introduce the concept of power spectral density and the idea of predictability of a random process.

Process no. 3.

As you did in Section 1.1, generate a realization of a zero-mean and unite variance white Gaussian random process (`NoiseX`) and filter this realization with an LTI system with impulse response `h1` that averages 10 consecutive samples of the input signal, generating a new random process (`NoiseY`). Listen the vector to appreciate the effect of filtering the initial noise (you can listen the noise before and after filtering). Plot the samples and, using the *Zoom In* tool, visualize the signal. SP3

```
h1=ones(1,10);
h1=h1./(sqrt(h1*h1'));

n=[0:15999];
NoiseX=randn(size(n));

NoiseY=conv(NoiseX,h1,'same');

Fs=8000;
soundsc(NoiseY,Fs);
plot(n/Fs,NoiseY);
```

Generate M=2000 realizations of this process, save them in a matrix (`SP3` matrix) and answer the following question: SP3

- Using the `subplot()` function, plot four different realizations of the filtered stochastic process. Describe, qualitatively, the aspect of the random process. Pay special attention to the evolution of the process and its variation and compare it with the white Gaussian process (`SP1`). 1.2a

The procedure to estimate the autocorrelation function of a random process will be studied in a posterior unit of the course. So, we skip now this part and we provide the MATLAB function `[Rx Tau]=IPALab2_XCorr(SPn,Lags)` that, given an M×N matrix with M realizations of a random process, estimates the autocorrelation of the process. Vector `Rx` contains the estimation of the autocorrelation $r_x(l)$ and `Tau` the lags in the interval `(-Lags,Lags)` where the autocorrelation has been calculated. To plot the autocorrelation of the process use `plot(Tau,Rx)`. Rx

Note: We denote the lag of the autocorrelation as the index m of the autocorrelation function $r_x[m]$.

- Using the previous MATLAB function, estimate the autocorrelation of the processes `SP1` and `SP3` for `Lags=127` and plot the results. Calculate also the modulus of their Fourier transforms (using `abs(fft(.))`). As it has been studied in class, the Fourier transform of the autocorrelation function is the *power spectral density*. 1.2b
  Compare the autocorrelation of the `SP1` and `SP3` processes. Compare also the power spectral density plots. Relate the narrowness or width of the power spectral density with the conclusions that you have drawn with respect to the evolution and variability of each process.

Generate again M=2000 realizations of a filtered process but now with a longer filter `h1=ones(1,100)`, save them in a matrix (`SP4` matrix) and visualize four arbitrary realizations of this process. SP4

- Using the MATLAB function `IPALab2_XCorr()` estimate and plot the autocorrelation of the new process (for `Lags=127`) and its power spectral density. Repeat the discussion done for SP3 for this new process. Could you calculate and plot (overlap) the theoretical expression of the power spectral density? 1.2c

- At the end of the course we will see some strategies to predict the future samples of a stochastic process using the previous ones. Some qualitative conclusions can now be drawn. In view of the results that you have obtained, could you stablish a link between the degree of predictability of a random process and the information (shape) given by the autocorrelation and power spectral density functions? 1.2d

After analyzing the power spectral density you should understand why the random process in Section 1.1 was named white Gaussian noise whereas we name the process in this Section as colored Gaussian noise[2].

2. Analysis of a real stochastic process

2.1. Analysis of power energy consumption

Electrical load measures in residential buildings is an example of stochastic process that has some interesting features. Extracted from a data set that contains hourly load profile data for hundreds of residential buildings in the US[3], the file `PowerConsumption.mat` presents information of 8 selected buildings with similar characteristics and located in the same area (Arizona State).

> *Source: https://openei.org/datasets/dataset/commercial-and-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-states*

The 8 realizations for this stochastic process were saved in an M×N matrix (M=8, N=8760) where each row is a realization of the process (a specific building) and each column is a random variable associated to a given hour of the year (power consumption during one year and with a sampling period of 1 hour was measured).

As expected, this random process is non-stationary since the power consumption is not constant over time. However, we can expect some consumption patterns that are cyclic along time. A stochastic process is said to be a *cyclostationary process* if it exhibits periodicity in their mean, correlation, or spectral characteristics. Specifically, a process is (wide-sense) cyclostationary if there exists an integer N such that:

$$E\{x[n]\} = E\{x[n+kN]\} \quad \text{and} \quad r_X(n;\tau) = r_X(n+kN;\tau)$$

---

2 *According to the power spectral density shape, there are different types of colored noise with significantly different properties, leading to different types of sounds and different applications. The name is inspired in an analogy to the colors of light. To listen different types of colored noise see e.g., https://mynoise.net/NoiseMachines/whiteNoiseGenerator.php or Wikipedia entries.*

Load `PowerConsumption.mat` file and do the following activities:

- Select three arbitrary rows of the matrix and using the `subplot()` function plot three different realizations of the random process. Describe, qualitatively, the aspect of the random process and identify its cyclic behavior. Explain the variations of the random process over time. 2.1

2.2. /a/ vowel

Record yourself pronouncing the vowel /a/ during approximately two seconds. Eliminate the transitory (first and last samples) and the save the record in a file.

Using the `Audio_Read` function load the vowel,

```
[y Fs]=Audio_Read(FileName);
```

calculate its autocorrelaction using the MATLAB function `IPALab2_XCorr()` for `Lags=2047`, and plot the power spectral density as you did in section 1.2.

Next figure illustrates the model of the human speech production system (this scheme was already presented in SIS course). Taking into account that the /a/ vowel is a voiced sound, justify the shape of the power spectral density that you have obtained and match the result with the pulse train generator and the digital filter. 2.2

Notice that you have obtained the conclusions using only a realization of the stochastic process /a/ vowel. So, in order to statistically characterize the stochastic process we need multiples realizations of the process. In order to obtain those realizations we ask you to deliver in Atenea the file that you recorded and we will try to use it in the future.

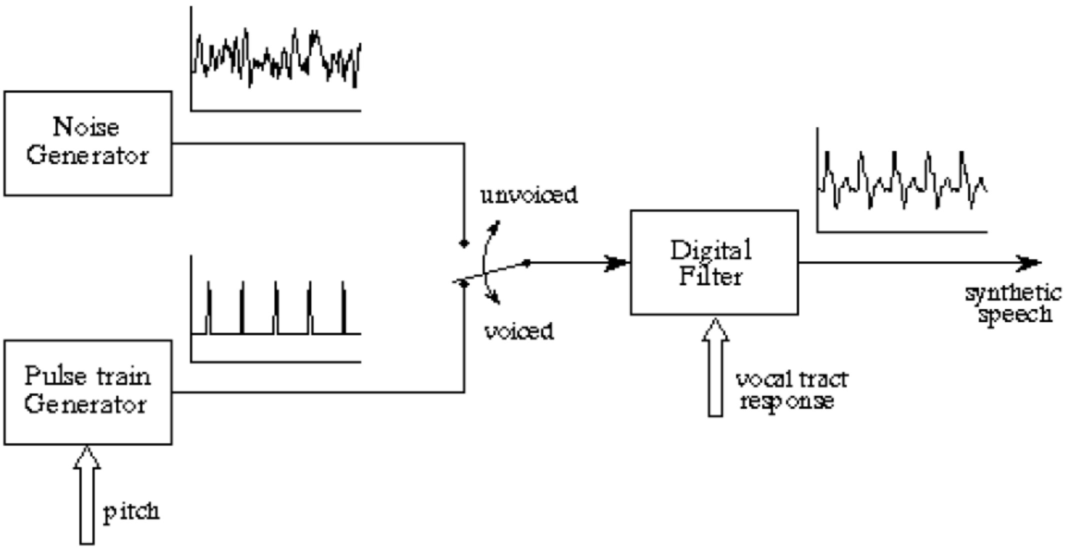


Figure. Model of the human speech production system.

RECORD a sound using MATLAB

```
Fs=44100;
Nbits=8;
r = audiorecorder(Fs, Nbits, 2);
record(r);
......................................
% Say /a/ during at least two seconds
pause(r);
stop(r);
p = getaudiodata(r, 'double');
plot(p);
.........................................
% Set IniSample and EndSample to eliminate the transitory
aVowel=p(IniSample:EndSample,:);
audiowrite('FILENAME.m4a',aVowel,Fs);
```

2.3. Identify a stochastic process

As it has been done for the power energy consumption, we ask you to choose a quotidian real stochastic processes and obtain (through internet) some realizations for that process. Plot several realizations of the process, comment the aspect of the process and discuss about the concepts developed in this lab session (stationarity, predictability, autocorrelation, power spectral density…). 2.3