Islamic University of Technology (IUT)

Organization of Islamic Cooperation (OIC)

Department of Electrical and Electronic Engineering (EEE)

Course No.	**:** Math 4422
Course Name	**:** Random Signals and Processes Lab
Experiment No.	**:** 03

❖ Permutations

The number of permutations of n objects taken k at a time is given by $\frac{n!}{(n-k)!}$

MATLAB has a built-in function.

```
>>perms([1 2 3])
```

What is the output that you get?

How can we execute ${}^{n}P_{k}$?

nchoosek(n,k)*factorial (k)

❖ The cumsum function

The cumsum function creates a vector in which each element is the cumulative sum of all the elements up to and including the comparable position in the original vector.

Example: Let x be a vector of values, sorted in ascending order, and p a vector of the probabilities associated with each of the corresponding values. Find the cumulative sum of the probabilities.

```
x=[10 20 30];
p=[0.20 0.30 0.50];
y=cumsum(p)
```

❖ Gaussian Probabilities

For a gaussian random variable X with mean m and standard deviation s, the cumulative distribution function is given by

$$F_X(x) = P\{X \le x\} = \frac{1}{2} + \frac{1}{2} erf\left(\frac{x-m}{s\sqrt{2}}\right)$$

where erf is the error function. MATLAB has a built-in error function erf defined by

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

The following results are important when evaluating gaussian probabilities:

$$P\{x_1 < X \le x_2\} = \frac{1}{2} erf\left(\frac{x_2-m}{s\sqrt{2}}\right) - \frac{1}{2} erf\left(\frac{x_1-m}{s\sqrt{2}}\right)$$

$$P\{X > x\} = 1 - P\{X \le x\} = \frac{1}{2} - \frac{1}{2} erf\left(\frac{x-m}{s\sqrt{2}}\right)$$

Problem:

A Gaussian voltage has a mean value of 5 and a standard deviation of 4.

1. Find the probability that an observed value of the voltage is greater than zero.
2. Find the probability that an observed value of the voltage is greater than zero but less than or equal to 5.

Solution:

```
clear all

% Answer to part (1)
m=5; s=4; x=0; % specify the mean, standard deviation, and x
z=(x-m)/(s*sqrt(2)); % define an intermediate variable z
P1=1/2-1/2*erf(z) % compute P(X>0)
```

❖ White noise

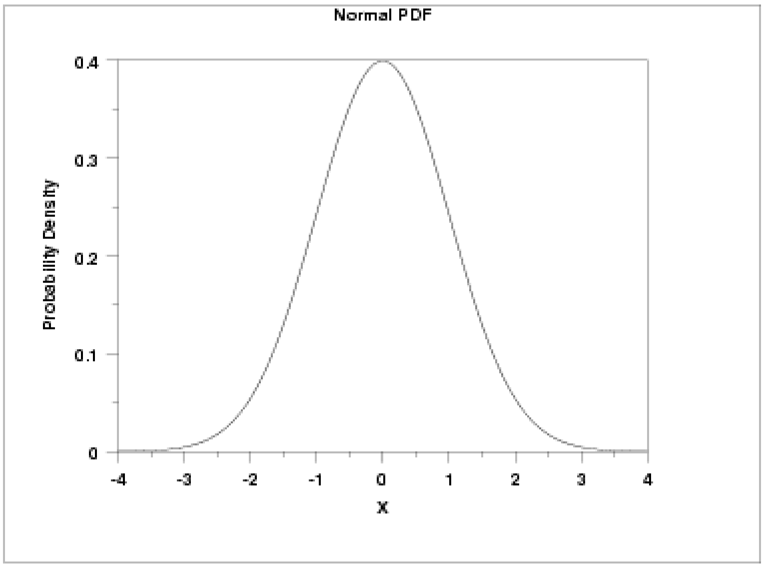
A uniform white noise is a sequence of independent samples with zero mean. The rand function generates a sequence of uniform pseudo numbers with mean of 0.5 and variance of 1/12. Similarly, the function randn provides a Gaussian sequence with zero mean and a variance of unity. Therefore, one can generate a white gaussian noise having an average power P via $\sqrt{P}$ randn.

Example:

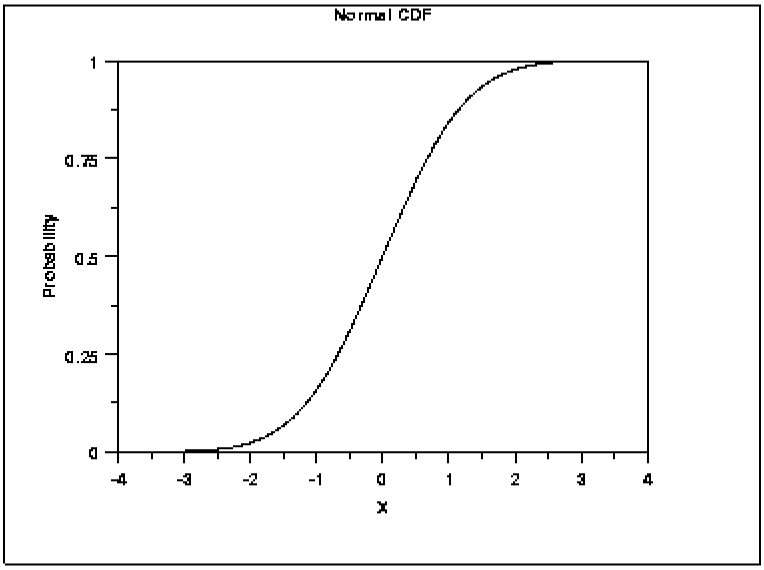
```
clear all
close all
%Signal-to-noise ratio=2
t=[0:512]/512; %define a time vector
signal=sqrt(2)*cos(2*pi*5*t); %define a signal sequence (average
power=1 W)
noise=sqrt(0.5)*randn(1,length(t)); %define a noise sequence
(average power=0.5 W)
sn=signal+noise; %compute the signal+noise sequence
subplot(2,1,1);
plot(t,sn);
legend('snr=2');
grid %plot the signal+noise sequence
%Signal-to-noise ratio=20
signal2=sqrt(20)*cos(2*pi*5*t); %define a signal sequence (average
power=10 W)
sn2=signal2+noise; %computer the signal+noise sequence
subplot(2,1,2);
plot(t,sn2);
grid %plot the signal+noise sequence
legend('snr=20');
```

❖ Probability Density Function (PDF) and Cumulative Distribution Function (CDF)

In probability theory, a probability density function (PDF), or density of a continuous random variable, is a function whose value at any given sample (or point) in the sample space (the set of possible values taken by the random variable) can be interpreted as providing a *relative likelihood* that the value of the random variable would equal that sample.



The cumulative distribution function (CDF) is the probability that the variable takes a value less than or equal to x.



Built in Matlab codes for calculating PDFs and CDFs

```
clear all

piglu=1;
giglu=2;
tmp=randi([1 1000],1,100); % generate 100 random numbers
%pd_tmp(piglu)=fitdist(tmp','logn'); %lognormal
%pd_tmp(piglu)=fitdist(tmp','weibull'); %weibull
pd_tmp(piglu)=fitdist(tmp','normal'); %normal
%pd_tmp(piglu)=fitdist(tmp','poisson'); %poisson
%pd_tmp(piglu)=fitdist(tmp','gamma'); %gamma
%pd_tmp(piglu)=fitdist(tmp','exponential'); %exponential

%Calculating PDFs

%s_tmp_pdf(piglu,:)=lognpdf(tmp,pd_tmp(piglu).mu,pd_tmp(piglu).sigma);%lognormal
%s_tmp_pdf(piglu,:)=wblpdf(tmp,pd_tmp(piglu).A,pd_tmp(piglu).B); %weibull
s_tmp_pdf(piglu,:)=normpdf(tmp,pd_tmp(piglu).mu,pd_tmp(piglu).sigma); %normal
%s_tmp_pdf(piglu,:)=poisspdf(tmp,pd_tmp(piglu).lambda); %poisson
%s_tmp_pdf(piglu,:)=gampdf(tmp,pd_tmp(piglu).a,pd_tmp(piglu).b); %gamma
%s_tmp_pdf(piglu,:)=exppdf(tmp,pd_tmp(piglu).mu); %exponential

figure(piglu)
plot(tmp,s_tmp_pdf(piglu,:),'o');

%Calculating CDFs

%tmp_prob(giglu,:)=logncdf(tmp,pd_tmp(piglu).mu,pd_tmp(piglu).sigma) %lognormal
%tmp_prob(giglu,:)=wblcdf(tmp,pd_tmp(piglu).A,pd_tmp(piglu).B); %weibull
tmp_prob(giglu,:)=normcdf(tmp,pd_tmp(piglu).mu,pd_tmp(piglu).sigma); %normal
%tmp_prob(giglu,:)=poisscdf(tmp,pd_tmp(piglu).lambda);%poisson
%tmp_prob(giglu,:)=gamcdf(tmp,pd_tmp(piglu).a,pd_tmp(piglu).b); %gamma
%tmp_prob(giglu,:)=expcdf(tmp,pd_tmp(piglu).mu); %exponential
figure(giglu)
plot(tmp,tmp_prob(giglu,:),'o');
```

❖ Cross-Correlation

In general, correlation describes the mutual relationship which exists between two or more things. The same definition holds good even in the case of signals. That is, correlation between signals indicates the measure up to which the given signal resembles another signal. Cross correlation is a kind of correlation, in which the signal in-hand is correlated with another signal so as to know how much resemblance exists between them.

a=[1 2 3 0]   b=[7 8 10 12]   - Example

```
            1   2   3  0     = 12      LAG -3
7   8  10  12

        1   2   3   0        = 34      LAG -2
7   8  10  12

    1   2   3   0            = 64      LAG -1
7   8  10  12

1   2   3   0                = 53      LAG 0 (one centered on the other)
7   8  10  12

1   2   3   0                = 38      LAG +1
    7   8  10  12

1   2   3   0                = 21      LAG +2
        7   8  10  12

1   2   3   0                = 0       LAG +3
            7   8  10  12
```

```
clear all
a=[1 2 3 0];
b=[7 8 10 12];
[cc,lag]=xcorr(a,b)
plot(lag,cc)
```

- Home task: Find **5** problems similar to the contents of today's lab and solve those. You can include the solutions if they are available online but all the codes should be properly explained using comments after each line.