ECE 341: Random Processes

www.BisonAcademy.com

Instructor:	Jacob Glower
Office Location:	ECE 201A
Office Phone:	none at present
Class Hours:	Noon - 2:30pm Monday - Friday Also live streamed on Zoom
Office Hours	M-F, 7-8pm on Zoom
Text:	Bison Academy
References	Get a used text book on Statistics. They sell for $6 on Amazon

Catalog Description: Principles of probability. Application of probability and statistics to electrical and computer engineering problems. 3 lectures. Prereq: MATH 266. F, S

Topics Covered: Combinatorics, trees, apriori, aposteriori, conditional probabilties, pdf, cdf, expectiotaions, single and multi random variables, discrete and continuous functions of random variables, analysis via statistics, MATLAB, and Monte Carlo approaches.

Course Objectives: By the end of the semester, students should be able to

- Explain what a random process is and give examples,
- Determine the mean, variance, and expected value of a random process given its cumulative distribution function or its probability density function via calculus as well as Monte-Carlo (MATLAB) simulations.
- Determine what type of distribution applies to a random process including: binomial, Poisson, geometric, hyper-geometric, and normal distributions.
- Determine if two random processes have similar means using a t-test.
- Determine if a random process has a given distribution using a chi-squared test.

Grading

- Midterms: 1 unit each
- Homework 1 unit
- Total: Average of all above

Final Percentage:

- 100% - 90% A
- 89% - 80% B
- 79% - 70% C
- 69% - 60% D
- < 59% E

Policies: A student may take a makeup exam if he/she misses an exam due to an emergency, illness, or plant trip and notifies me in advance of the exam. Late homework will not be accepted once the solutions are posted online.

Homework: Homework is due by 8am the following day. (I'll start grading the homework at 8am and hopefully have them finished by 11am at the start of office hours).

Testing: All tests will be open-book, open-notes, calculators, matlab, the internet permitted. Individual effort though (i.e. no working together). Midterms serve to identify who put in the time solving the homework problems. My goal in writing tests is add new twists you haven't seen yet so that

- If you did the homework and are comfortable with the concepts and tools, you'll have a shot at the midterms.
- If you copied someone else's homework, you'll be lost.

The best way to study for the midterm is to make up your own midterm. There's only so many ways to ask a question.

The rules for tests are:

- Tests will be posted at 8am, due at 7am the next day.
- 2 hour time limit from when you start
- Each test will be different (each student generates random numbers for the test)
- All tests are open-book, open notes, calculators, Matlab all permitted
- Providing or receiving help from others prohibited
- Posting test on-line and/or using an on-line solution prohibited

Special Needs - Any students with disabilities or other special needs, who need special accommodations in this course are invited to share these concerns or requests with the instructor as soon as possible.

Academic Honesty - All work in this course must be completed in a manner consistent with NDSU University Senate Policy, Section 335: Code of Academic Responsibility and Conduct. Violation of this policy will result in receipt of a failing grade.

ECE Honor Code: On my honor I will not give nor receive unauthorized assistance in completing assignments and work submitted for review or assessment. Furthermore, I understand the requirements in the College of Engineering and Architecture Honor System and accept the responsibility I have to complete all my work with complete integrity.

Introduction

What is a random process?

A random process is a process whose outcome is not completely repeatable. For example, the function 'rand' in MATLAB generates a random number in the interval (0,1). If you call it several times, you get a different result each time:

```
rand()
 ans  =

    0.0826407

and()
 ans  =

    0.2014105
```

MATLAB is a matrix language. If you want 10 random numbers stored in a 10x1 array, simply type:

```
-->X = rand(10,1)
 X  =

    0.2113249
    0.7560439
    0.0002211
    0.3303271
    0.6653811
    0.6283918
    0.8497452
    0.6857310
    0.8782165
    0.0683740
```

You can also make this a discrete (integer) random variable. For example, if you roll ten six-sided dice, the results could be:

```
-->Die = ceil( 6 * rand(10,1) )
 Die  =

    2.
    5.
    1.
    2.
    4.
    4.
    6.
    5.
    6.
    1.
```

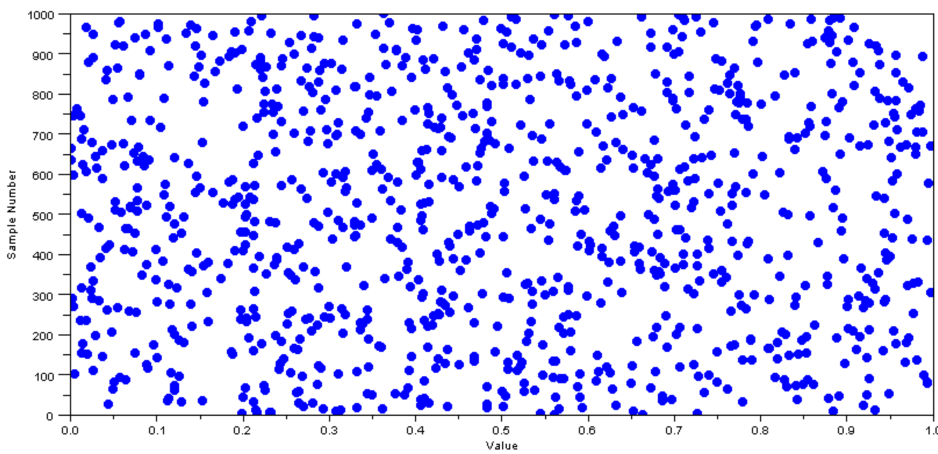
One of the objectives of this course is to provide tools for describing and analyzing such random events.

Here's a problem: how would you mathematically describe the rand() function in MATLAB? Or, more generally, how would you mathematically describe any random process?

A simple technique is called a Monte Carlo simulation. With this, you just taks a large number of samples and argue that if you let the sample size go to infinity, you fully describe the random process. Since infinity is a little unwieldy to work with, lets take 1000 samples:

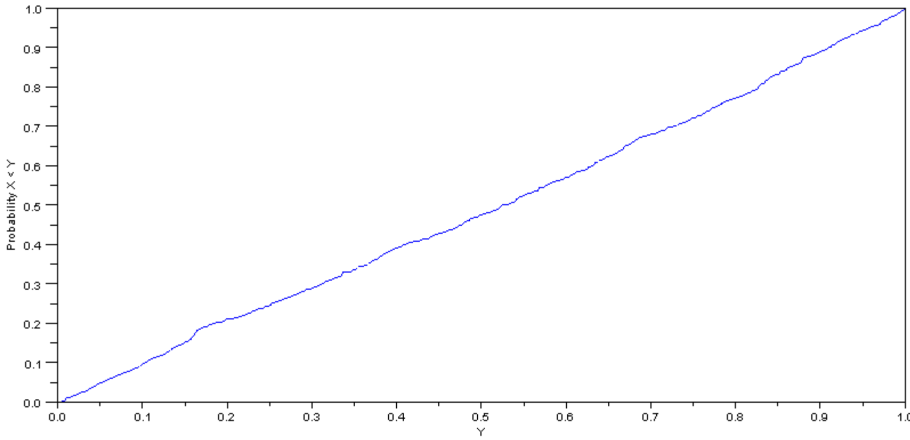
```
X = rand(1000,1);
Die = ceil( 6 * rand(1000,1) );
```

List 1000 numbers isn't very useful. Instead, lets plot the data:



Again, this doesn't tell you much. Let's sort the data and then plot it:

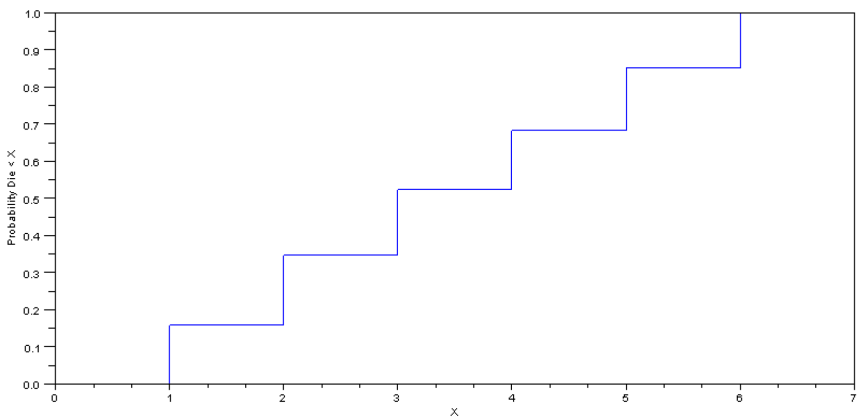
```
Xs = sort(X);

N = [1:1000]';
N2 = 1001 - N;

plot(Xs(N2), N/1000)

xlabel('Y');
ylabel('Probability X < Y')
```



Cumulative Distribution Function for a Uniform Distribution

or for the dice:

```
plot(Die2(N2), N/1000);
xlabel('X');
ylabel('Probability Die < X');
```



Cumulative Distribution Function for a 6-sided Die

What we've been plotting is termed the Cumulative Distribution Function (CDF) - which is the probability that the result of the random process is less than x:

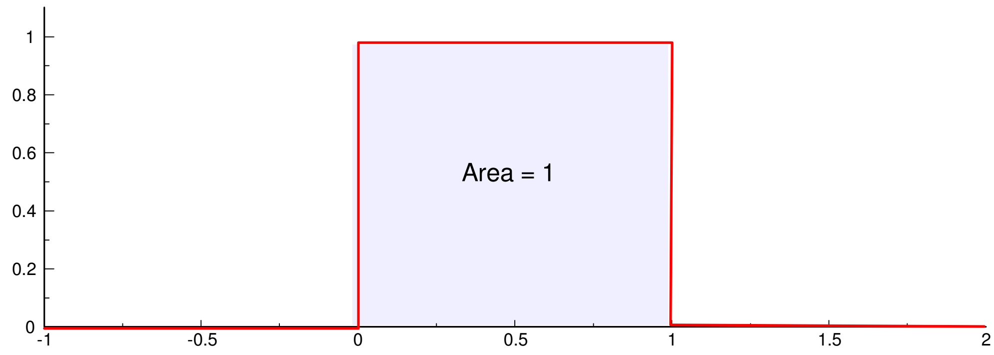
$$f(x) = P(y < x)$$

Note that for a CDF,

- It starts at zero (the probability that the result is less than minus-infinity is zero. It has to be something.)
- It goes to one (the probability that some number results is one.)

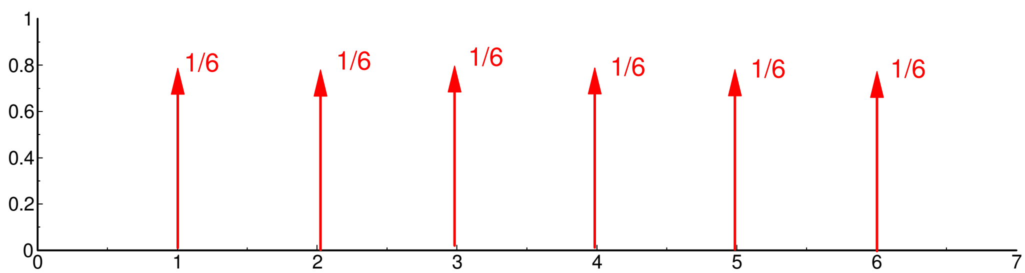
Another useful way of looking at the data is to take the derivative of the Cumulative Distribution Function. This is called the Probability Density Function.

For the function rand(), the derivative should look like the following:



Probability Density Function for a uniform distribution

For the dice, it looks like a bunch of delta functions:



Probability Density Function for a 6-sided die

This is the first concept to get across in this class: A 'good' way to describe a random process is to give its cumulative distribution function and its probability density functions. With this, we'll look at some standard distributions:

- Bernoulli Trial: Flip a coin
- Binomial Distribution: Flip N coins
- Geometric Distribution: Flip a coin until you get a heads
- Pascal Distribution: Flip a coin until you get N heads
- Hyper Geometric: Geometric without replacement
- Poisson: Number of events in a time interval
- Normal (Gaussian) Binomial distribution where N goes to infinity

Next, suppose we would like to describe a random process with a single number. That's the same as describing each of the above two graphs with a single number. You can't do it.

A second concept to get across is that simply giving the mean of a random process doesn't tell you much. You need more information. Ideally, you'd like to use the CDF or PDF. If you have to a few numbers, some measures, such as average, spread, etc. are needed. This is what the sample, mean (average), standard deviation (spread) tell you. The general term are the moments of a function:

- M_1 = average(X) 1st moment
- M_2 = average(X^2) 2nd moment
- M_3 = average(X^3) 3rd moment
- etc.

Third, suppose you have a random process and you'd like to test a hypothesis, such as

- Does Y have a mean that's greater than X?
- Is X a uniform distribution?

The first question results in comparing the means of the samples with the spread (standard deviation) taken in to account. This is a t-test.

The second question results in comparing the sample PDF or CDF's and results in a chi-squared distribution.