09-Random Numbers

text: Chapter 3.4-3.7

ECEGR 101
Engineering Problem Solving with Matlab
Professor Henry Louie

Overview

- Pseudo Random Numbers
- Random Integers
- Probability Density Functions

Pseudo Random Numbers

- Matlab can be used to generate "random" numbers
- Computers do not generate truly random numbers, rather they generate pseudo random numbers that "appear" random according to an algorithm
- Before we generate random numbers, we need to define the parameters of the random number generation

Random Integers

- Assume we want to simulate rolling a six-sided die
  - Equal chance of getting 1, 2, …, 6
  - Random integer generation

R = randi(IMAX,N) returns an N-by-N matrix containing pseudorandom integer values drawn from he discrete uniform distribution on 1:IMAX

Random Integers

- Example

```
>> randi(6,1)

ans =

     5

>> randi(6,1)

ans =

     6

>> randi(6,1)

ans =

     1
```

sequence: 5, 6, 1, and so on

Random Integers

- The numbers are psuedo random and you can control the sequence that occurs
- Use "rng(X)" to "seed" the psuedo random number generator
- Example:

```
>> rng(782);randi(6,1)
ans =
     6
>> randi(6,1)
ans =
     3
>> randi(6,1)
ans =
     4
```

An arbitrary seed

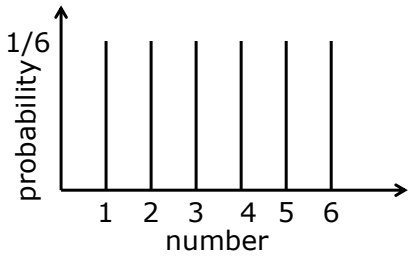
```
>> rng(782);randi(6,1)
ans =
     6
>> randi(6,1)
ans =
     3
>> randi(6,1)
ans =
     4
>> rng(782);randi(6,1)
ans =
     6
>> randi(6,1)
ans =
     3
>> randi(6,1)
ans =
     4
```

Random sequence repeats

Random Integers

- Do not need to use rng unless you want to repeat the randomness
  - Why would you ever want to do this?
- What about continuous random variables?

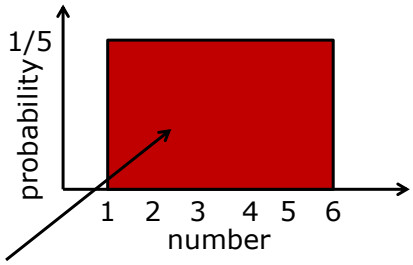
Probability Density Functions



Probability density functions (pdfs) describe the probability of outcomes of random number generators. Integral of PDF is always 1.

Probability Density Functions

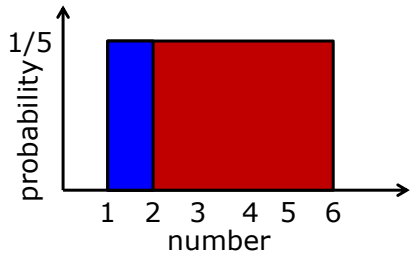
Continuous PDF. Equal probability of any number between 1 and 6



Area = 1

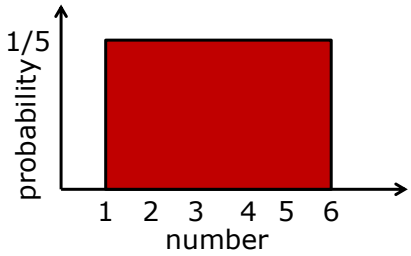
This is a "Uniform" PDF

Probability Density Functions



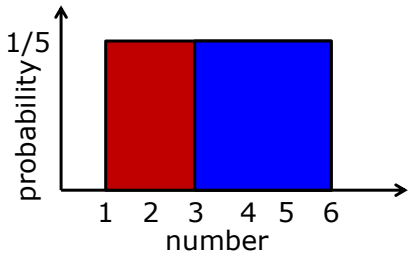
Probability of a number between 1 and 2 is 20% (the blue area)

Probability Density Functions



What is the probability of a number between 3 and 6?

Probability Density Functions



What is the probability of a number between 3 and 6?

60%

Random Numbers: Uniform Distribution

- rand function generates uniformly distributed numbers between 0 and 1.
- rand(m,n) generates an mxn matrix of random numbers between 0 and 1.
- rand generates one random number.
- randi generates random integer

```
>> x1 = rand
x1 =
   0.6154
>>
>> x2 = rand
x2 =
   0.7919
```

```
>> y = rand(2,3)
y =
   0.8913    0.4565    0.8214
   0.7621    0.0185    0.4447
```


Random Numbers: Uniform Distribution

- How can we generate random numbers that are uniformly distributed in an interval (a,b)?
  - (b-a)*rand + a
- Example: (6,8)

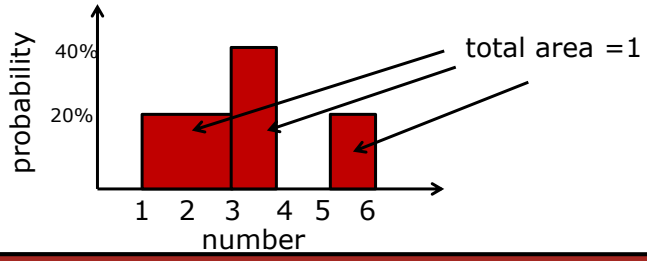
```
>> 2*rand+6
ans =
   7.4764
>>
>> 2*rand+6
ans =
   6.3525
```

```
>> 2*rand+6
ans =
   6.8114
>>
>> 2*rand+6
ans =
   7.8709
```

```
>> x = 2*rand(2,2)+6
x =
   6.8205    6.1158
   7.7873    6.7057
>>
>> x = 2*rand(2,2)+6
x =
   7.6263    6.2778
   6.0197    6.4055
```

Probability Density Functions

- Consider the following probabilities
  - 20% between 1 and 2
  - 20% chance between 2 and 3
  - 40% chance between 3 and 4
  - 0% chance between 4 and 5
  - 20% chance between 5 and 6



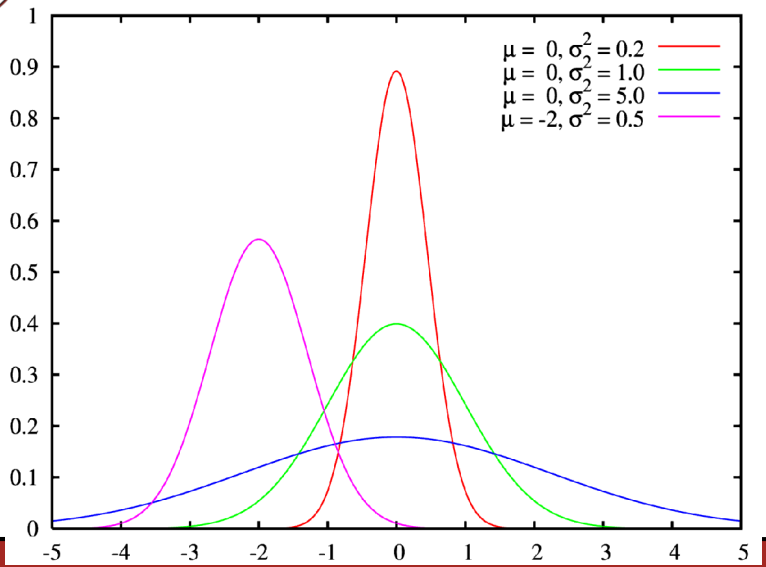
Probability Density Functions

- We can bias our results by changing the PDF from which the random variables are generated from
- Common to use "families" of PDFs and change parameters to adjust their shape

Normal Distribution

- The normal distribution, also called Gaussian distribution is a probability distribution of great importance in many fields.
- The standard normal distribution is the normal distribution with a mean of zero and a variance of one.
- It is often called the bell curve because the graph of its probability density resembles a bell.

Normal Distribution



Random Numbers: Normal Distribution

- randn function generates normally distributed numbers with mean 0 and standard deviation 1.

```
>> randn
ans =
   -0.4326
>>
>> randn
ans =
   -1.6656
```

```
>> randn
ans =
    0.1253
>>
>> randn
ans =
    0.2877
```

```
>> randn(5,2)
ans =
   -1.1465    0.1746
    1.1909   -0.1867
    1.1892    0.7258
   -0.0376   -0.5883
    0.3273    2.1832
```


Random Numbers: Normal Distribution

- To change the mean and standard deviation of the random numbers:
  - (new_standard_deviation)*randn + new_mean

```
>> 5*randn+2
ans =
    1.3180
>>
>> 5*randn+2
ans =
    2.5697
```

```
>> 5*randn+2
ans =
    7.3338
>>
>> 5*randn+2
ans =
    2.2964
```

```
>> 5*randn+2
ans =
    1.5218
>>
>> 5*randn+2
ans =
   -2.1617
```

Random Numbers: Normal Distribution

- Alternatively, use the "random" function

R = random(NAME,A) returns an array of random numbers chosen from the one-parameter probability distribution specified by NAME with parameter values A.

```
>> x=random('normal',10,0.5,1)
x =
    9.3532
```

Single random number drawn from a Normal distribution with mean 10 and standard deviation of 0.5

Exercise

Simulate rolling a six-sided die 100 times. What is the average of the numbers you rolled? How close is this to the theoretical average?

Exercise

Simulate rolling a six-sided die 100 times. What is the average of the numbers you rolled? How close is this to the theoretical average?

```
>> rolls=randi(6,10,10);mean(mean(rolls))
ans =
    3.5500
```

(rolls is a matrix)

or

```
>> rolls=randi(6,[100 1]); mean(rolls)
ans =
    3.2200
```

(rolls is a vector)

Theoretical is (1+2+3+4+5+6)/6 = 3.5