Continuous
Probability Density Functions

ECE 341: Random Processes

Lecture #13

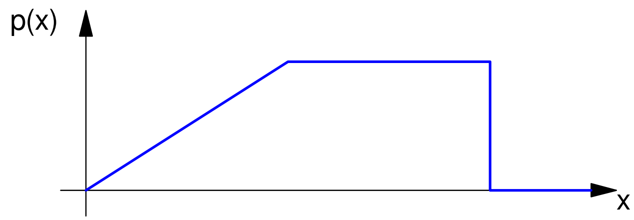
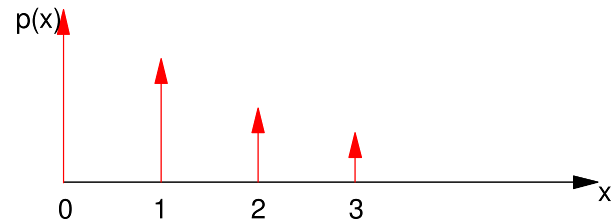
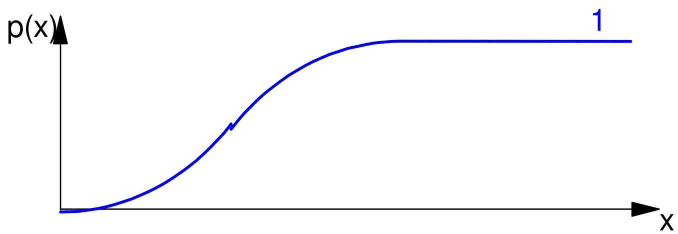
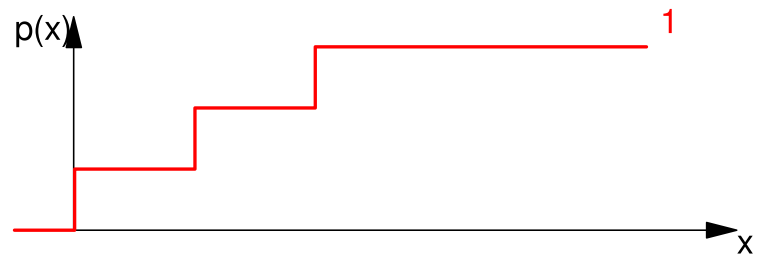
note: All lecture notes, homework sets, and solutions are posted on www.BisonAcademy.com

Continuous Probability Density Functions

- The value of a 1k resistor
- The gain of a 3904 transistor
- The resistance of a thermistor at 25C
- The hottest temperature in May,
- etc.

For each of these cases, the variable can assume any value over a range.

Most of what we did with discrete-time probabilities apply to continuous time - with just a slight change.

	Continuous	Discrete
pdf probability density function pdf = derivative of cdf		
cdf cumulative density function cdf = integral of pdf		
mgf moment generating function	$mgf = \psi(s)$	$mgf = \psi(z)$
m0 Area = 1	$m_0 = \psi(s=0) = 1$	$m_0 = \psi(z=1) = 1$
m1 mean	$m_1 = \psi'(s=0)$	$m_1 = -\psi'(z=1)$
m2	$m_2 = \psi''(s=0)$	$m_2 = \psi''(z=1)$
variance	$\sigma^2 = m_2 - m_1^2$	$\sigma^2 = m_2 - m_1 - m_1^2$

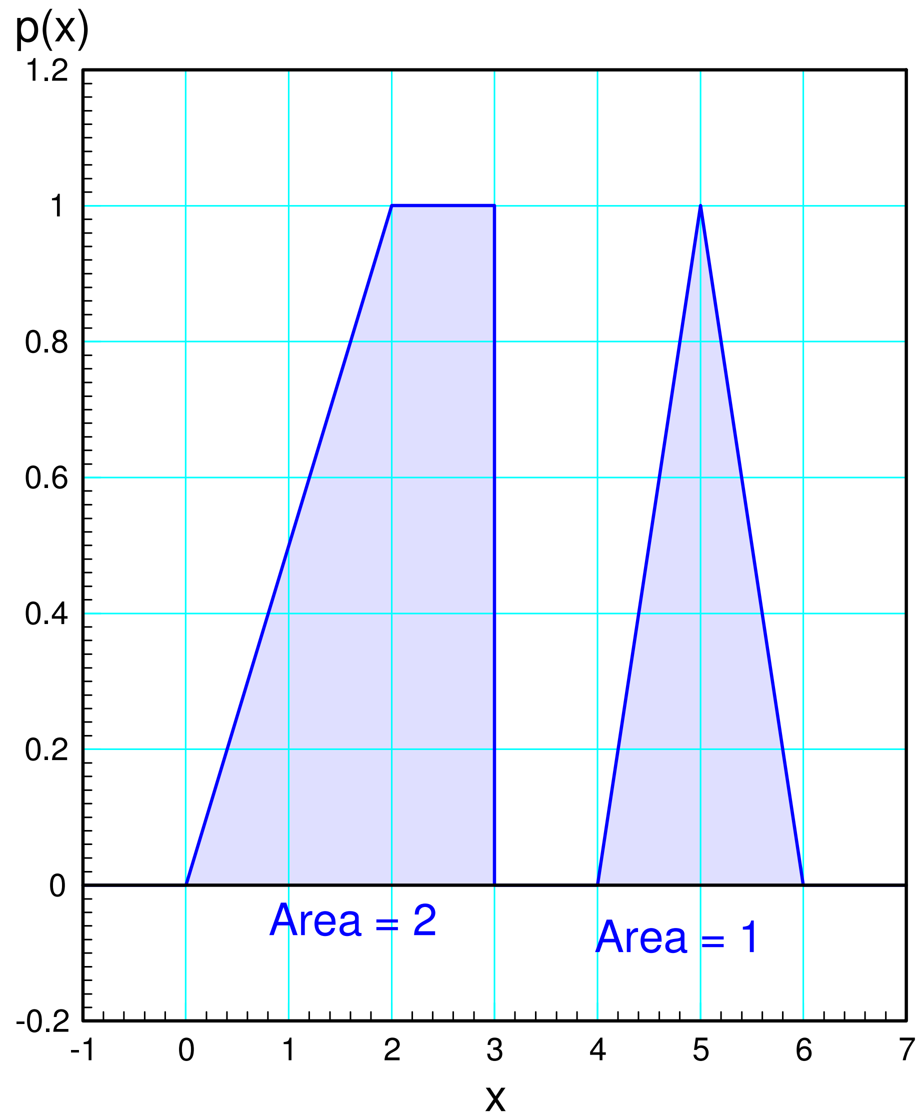
Example of a Continuous pdf

Find a scalar to make this a valid pdf

- Total area must be 1.0000
- All probabilities add to 1.000

Current area is 3.000

- Scale by 1/3



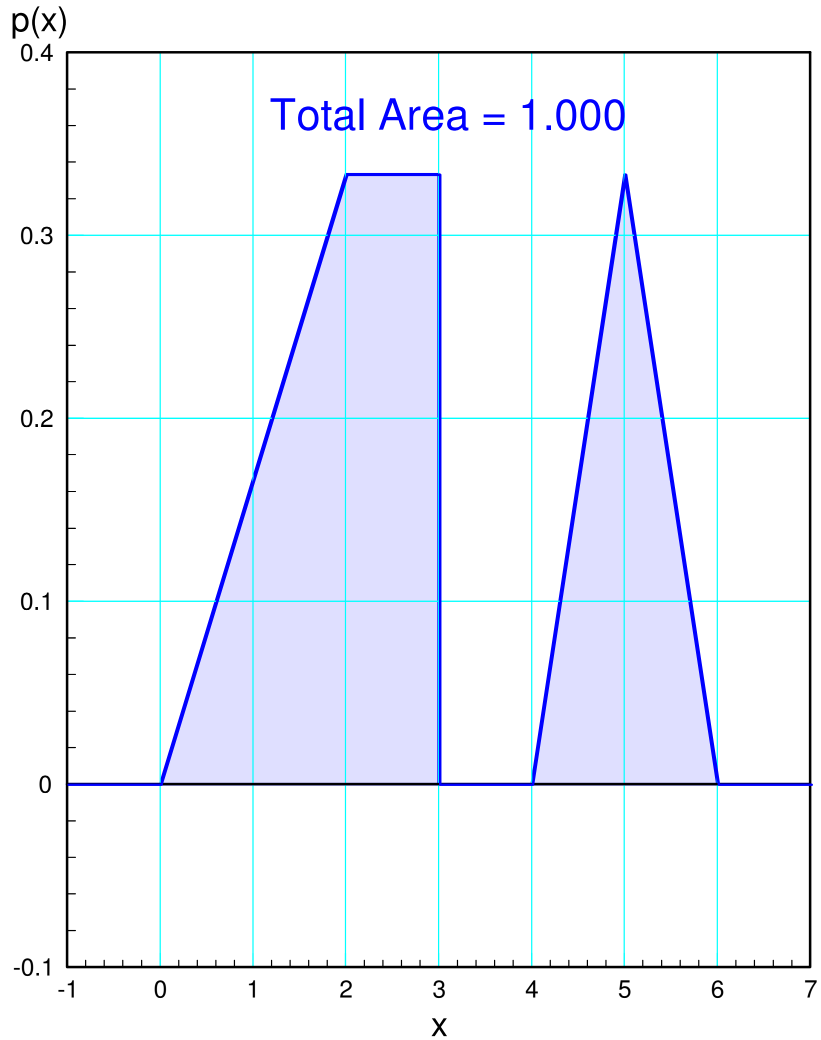
Implementation in Matlab

To implement this pdf, it's actually easier to use the cdf (the integral of the pdf)

$$c(x) = \int_{-\infty}^{x} p(y)\, dy$$

Expressing p(x) mathematically:

$$p(x) = \begin{cases} 0 & x < 0 \\ x/6 & 0 < x < 2 \\ 1/3 & 2 < x < 3 \\ 0 & 3 < x < 4 \\ (x-4)/3 & 4 < x < 5 \\ (6-x)/3 & 5 < x < 6 \\ 0 & 6 < x \end{cases}$$



The cdf is then the integral of the pdf

- You start at 0
- You end at 1
- Each time you integrate, you get an integration constant

$$p(x) = \begin{cases} 0 & x < 0 \\ x/6 & 0 < x < 2 \\ 1/3 & 2 < x < 3 \\ 0 & 3 < x < 4 \\ (x-4)/3 & 4 < x < 5 \\ (6-x)/3 & 5 < x < 6 \\ 0 & 6 < x \end{cases} \qquad cdf(x) = \begin{cases} 0 & x < 0 \\ \frac{x^2}{12} + c & 0 < x < 2 \\ \frac{x}{3} + c & 2 < x < 3 \\ c & 3 < x < 4 \\ \frac{x^2}{6} - \frac{4}{3}x + c & 4 < x < 5 \\ 2x - \frac{x^2}{6} + c & 5 < x < 6 \\ 1 & 6 < x \end{cases}$$

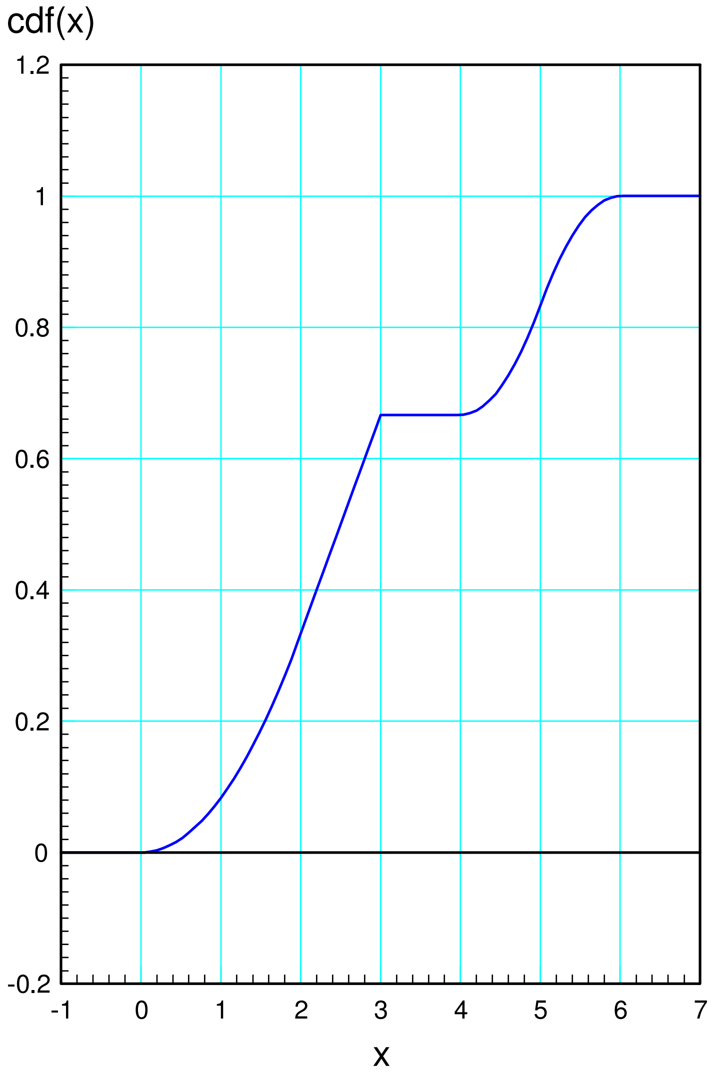
Integration Constants

- The cdf should be continuous
- cdf(x-) = cdf(x+)

$$cdf(x) = \begin{cases} 0 & x < 0 & \\ \frac{x^2}{12} + 0 & 0 < x < 2 & cdf(0) = 0 \\ \frac{x}{3} - \frac{1}{3} & 2 < x < 3 & cdf(2) = \frac{1}{3} \\ \frac{2}{3} & 3 < x < 4 & cfd(3) = \frac{2}{3} \\ \frac{x^2}{6} - \frac{4}{3}x + \frac{10}{3} & 4 < x < 5 & cdf(4) = \frac{2}{3} \\ 2x - \frac{x^2}{6} - 5 & 5 < x < 6 & cdf(6) = 1 \\ 1 & 6 < x & \end{cases}$$

In Matlab you can express this using if statements

```
function [ y ] = cdf( x )
if ( x < 0)
    y = 0;
elseif (x < 2)
    y = x*x/12;
elseif (x < 3)
    y = (x-2)/3 + 1/3;
elseif (x < 4)
     y = 2/3;
elseif (x < 5)
     y = (x-4)^2 /6 + 2/3;
elseif (x < 6)
    y = 1 - (6 - x)^2 / 6;
else y = 1;
end
```

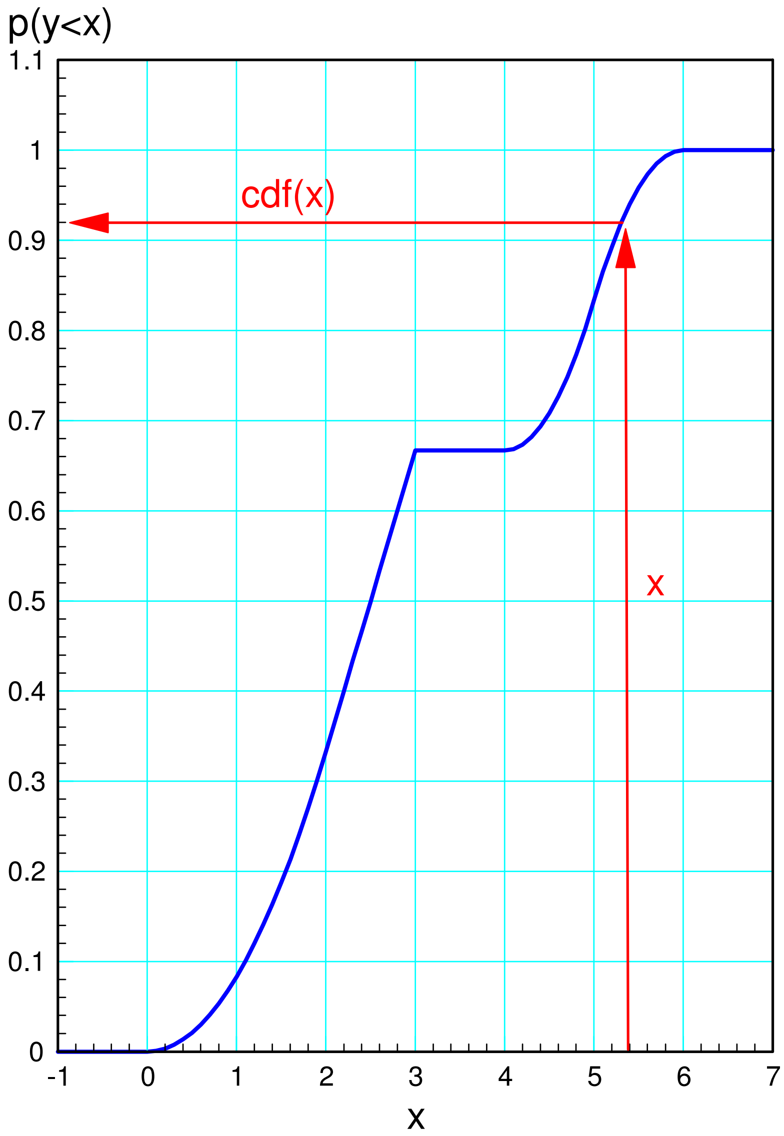


With this function,

- Given x,
- Determine cdf(x)

Example:

```
x = [-1:0.1:10]';
y = 0*x;

for i=1:length(x)
   y(i) = cdf(x(i));
   end

plot(x,y)
```



Generating x given p:

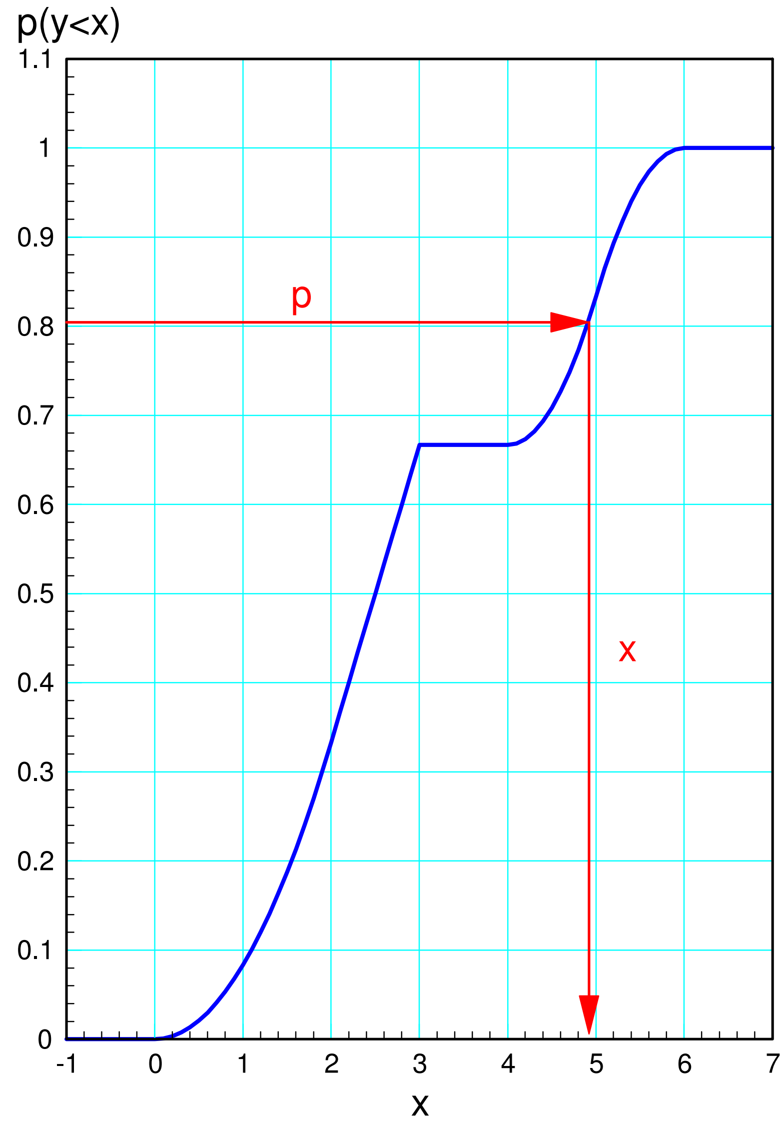
Method #1: Sweep right until cdf(x) > p

- Simple but converges slowly
- 3 decimal places accuracy in 5000 iterations

```
function [x] = pdf(p)

x = 0;
while(cdf(x) < p)
   x = x + 0.001;
   end

end
```

From the command window

```
>> pdf(0.8)

ans =     4.8950
```

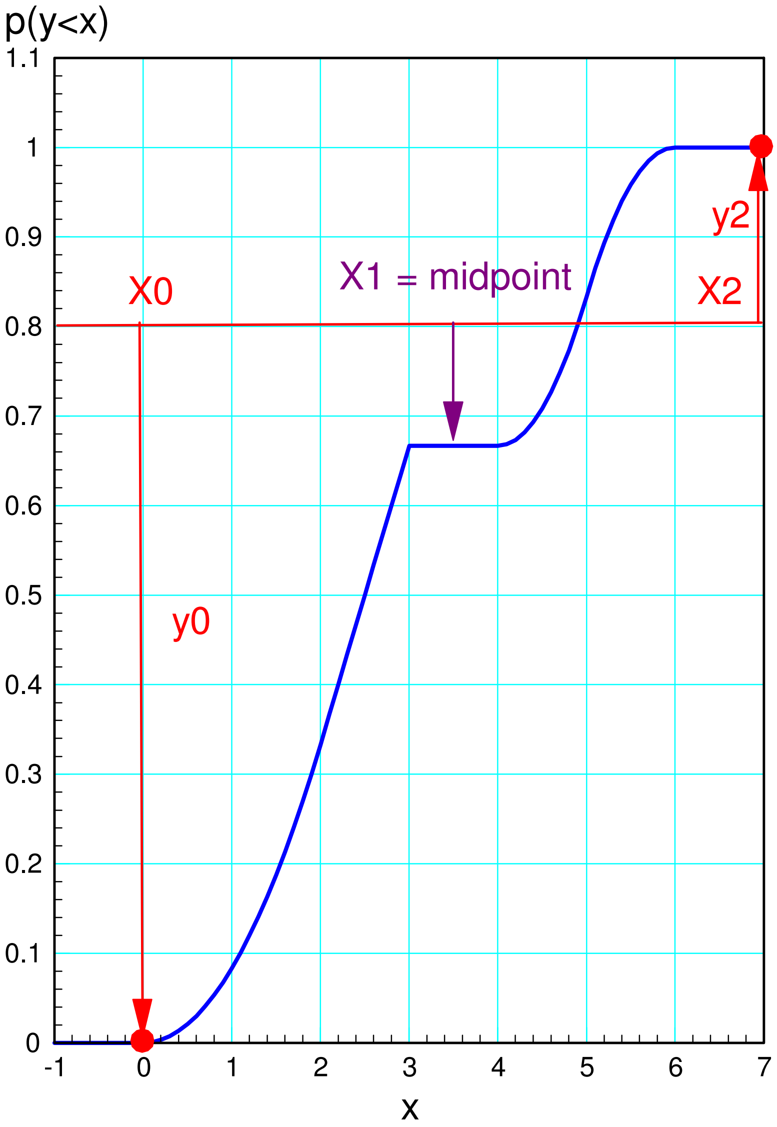


Method #2: Interval Halving

- Start with two guesses: (left & right)
- Check the midpoint
- If low, replace the left endpoint
- If high, replace the right endpoint
- Repeat

Code

```
function [x] = pdf2(p)

x0 = 0;
x2 = 10;
for n=1:20
   x1 = (x2 + x0)/2;
   y1 = cdf(x1) - p;
   if(y1>0) x2 = x1;
   else x0 = x1;
   end

x = (x0 + x2)/2;
end
```



Interval Halving Example

- Converges faster
- Five decimal places of accuracy after 18 iterations

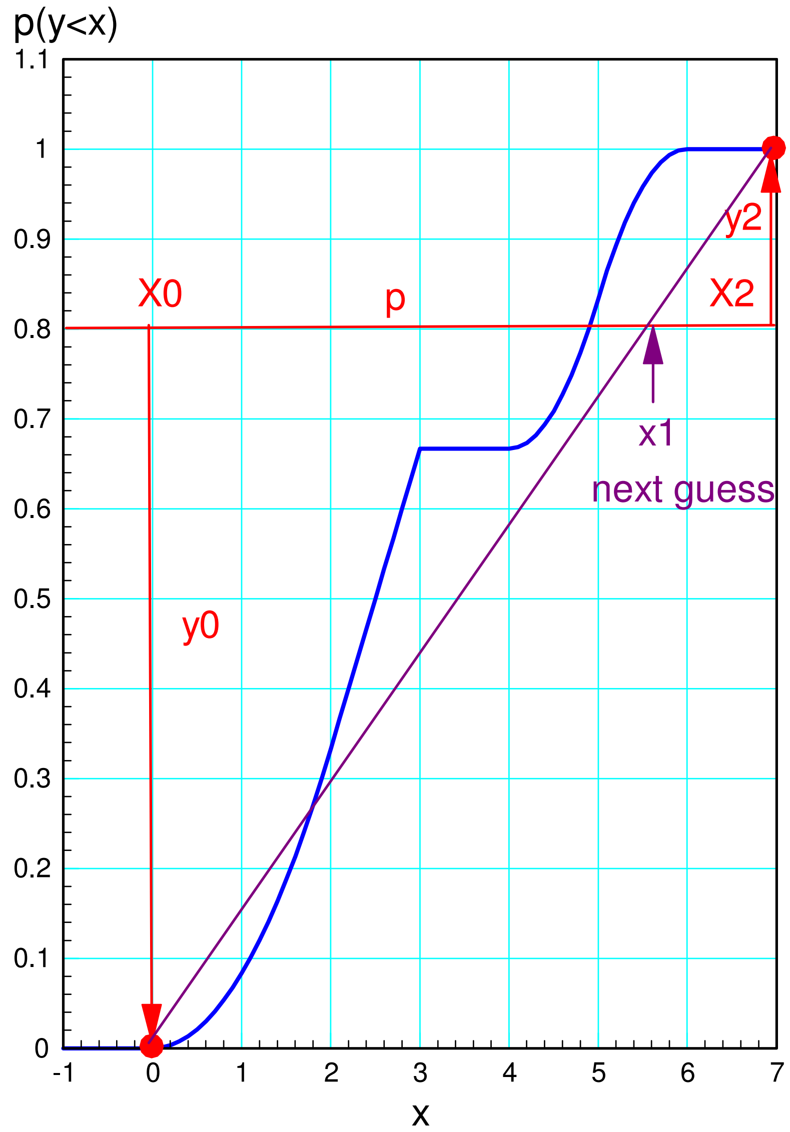
```
x = cdf2(0.8)
```

n	left	middle	right
1	0.0000	5.0000	10.0000
2	0.0000	2.5000	5.0000
3	2.5000	3.7500	5.0000
4	3.7500	4.3750	5.0000
5	4.3750	4.6875	5.0000
6	4.6875	4.8438	5.0000
7	4.8438	4.9219	5.0000
8	4.8438	4.8828	4.9219
9	4.8828	4.9023	4.9219
10	4.8828	4.8926	4.9023
11	4.8926	4.8975	4.9023
12	4.8926	4.8950	4.8975
13	4.8926	4.8938	4.8950
14	4.8938	4.8944	4.8950
15	4.8944	4.8947	4.8950
16	4.8944	4.8946	4.8947
17	4.8944	4.8945	4.8946
18	4.8944	4.8944	4.8945
19	4.8944	4.8944	4.8944
20	4.8944	4.8944	4.8944

Method #3: California Method

- Interpolate to find the next guess

```
function [x] = pdf3(p)

x0 = 0;
y0 = cdf(x0) - p;
x2 = 10;
y2 = cdf(x2) - p;
for n=1:20
   x1 = x0 - (x2-x0)/(y2-y0)*y0;
   y1 = cdf(x1) - p;
   disp([n, x0, x1, x2])
   if(y1 > 0)
       x2 = x1;
       y2 = y1;
   else
       x0 = x1;
       y0 = y1;
   end
end
x = x1;
end
```



California Method Example:

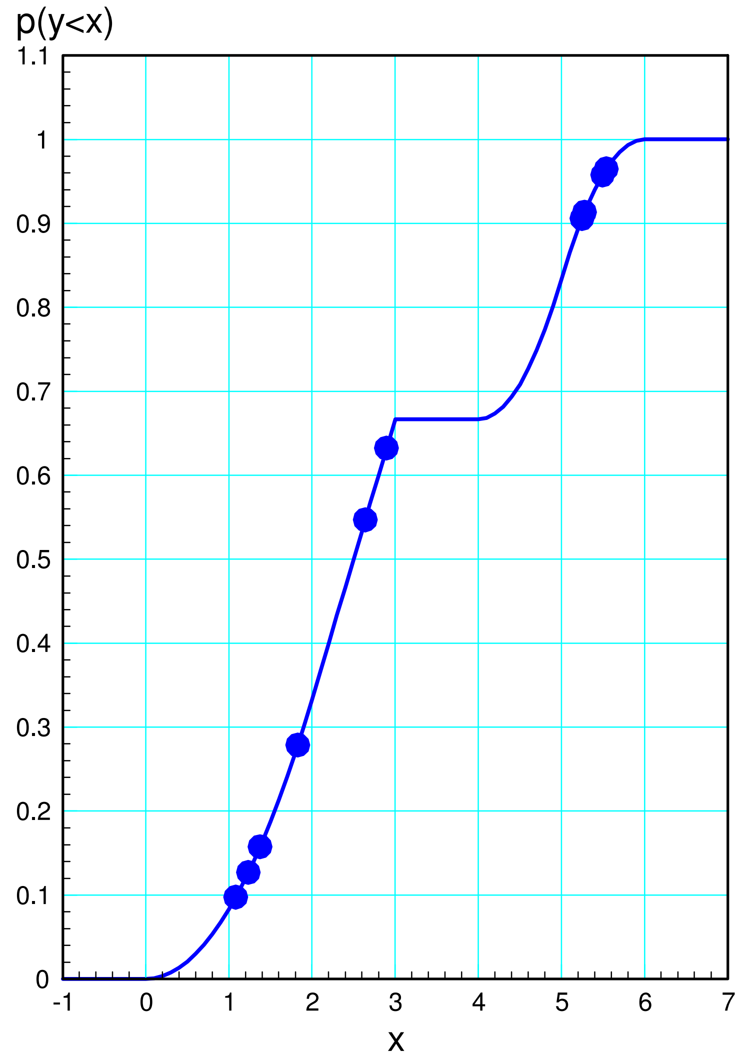
- note: converges much faster
- good to five decimal places in 7 iterations

```
>> pdf3(0.8)

    n     x0          x1          x2
    1     0.0000      8.0000     10.0000
    2     0.0000      6.4000      8.0000
    3     0.0000      5.1200      6.4000
    4     0.0000      4.7030      5.1200
    5     4.7030      4.8773      5.1200
    6     4.8773      4.8935      5.1200
    7     4.8935      4.8944      5.1200
    8     4.8944      4.8944      5.1200
    9     4.8944      4.8944      5.1200
   10     4.8944      4.8944      5.1200

ans =     4.8944
```

So now with this function, we can generate random values of x:

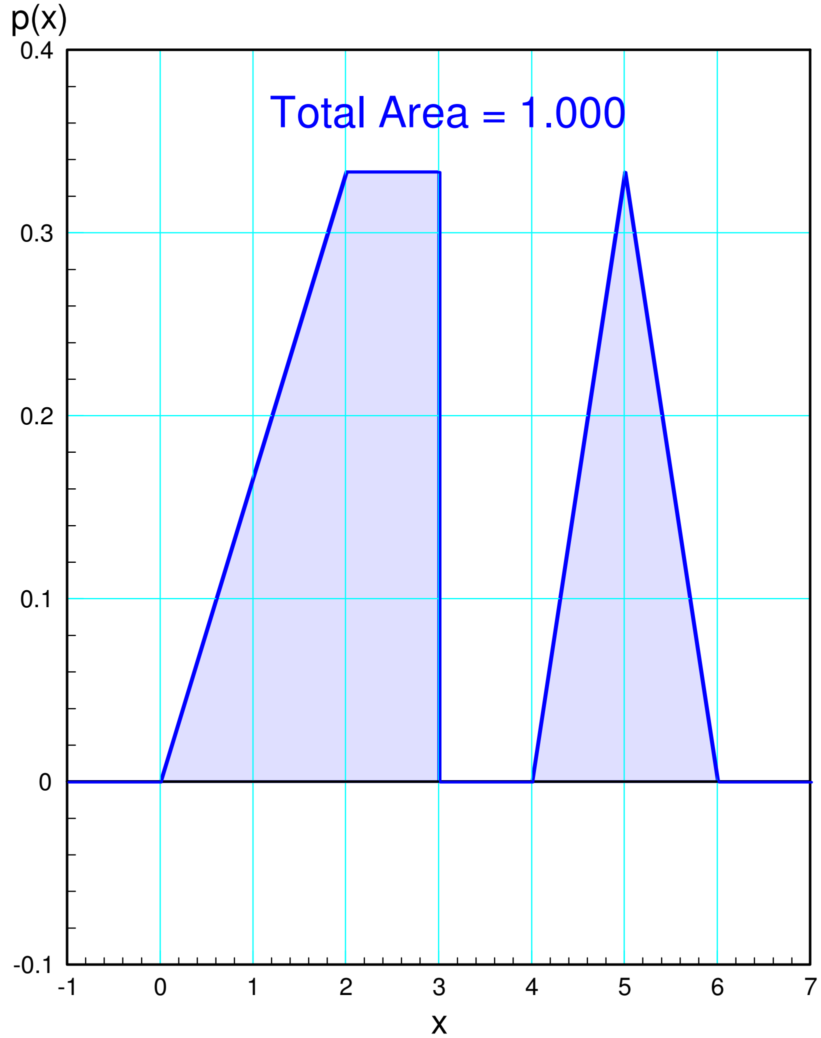
```
for i=1:10
   p = rand;
   x = pdf(p);
   disp([p,x])
   end
```

p	x
0.9058	5.2482
0.1270	1.2344
0.9134	5.2791
0.6324	2.8971
0.0975	1.0819
0.2785	1.8281
0.5469	2.6406
0.9575	5.4951
0.9649	5.5410
0.1576	1.3753



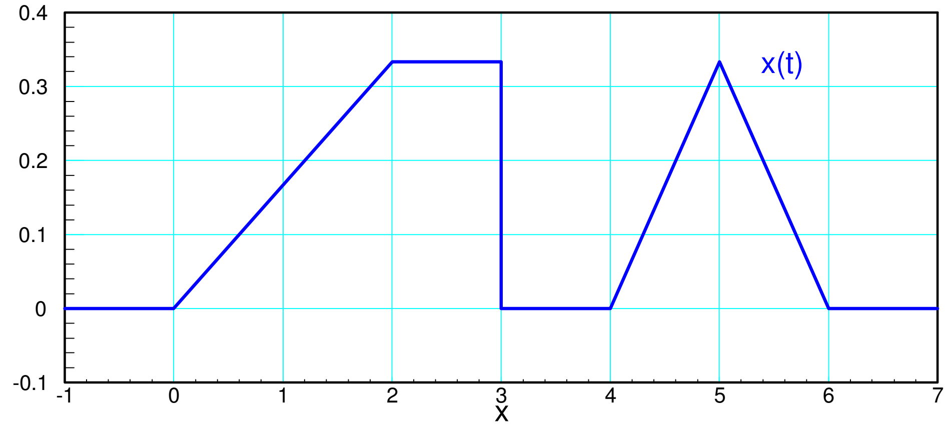
Moment Generating Function

- Find the LaPlace transform of the pdf
- The result is the moment generating function



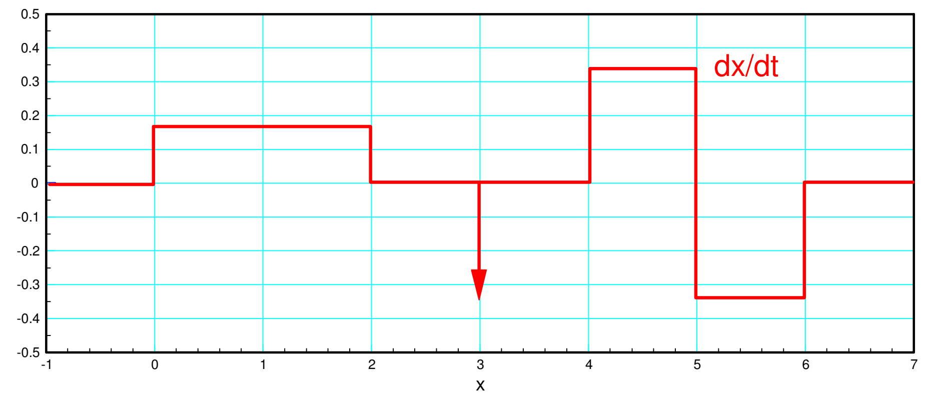
Take the derivative

- goal: end up with something we recognize (a delta function typically)
- differentiation is multiplying by 's'
- $L(\delta) = -0.333\ e^{-3s}$



Integrate once to get back to the pdf

$$X(s) = \left(\frac{-0.333}{s}\right) e^{-3s}$$

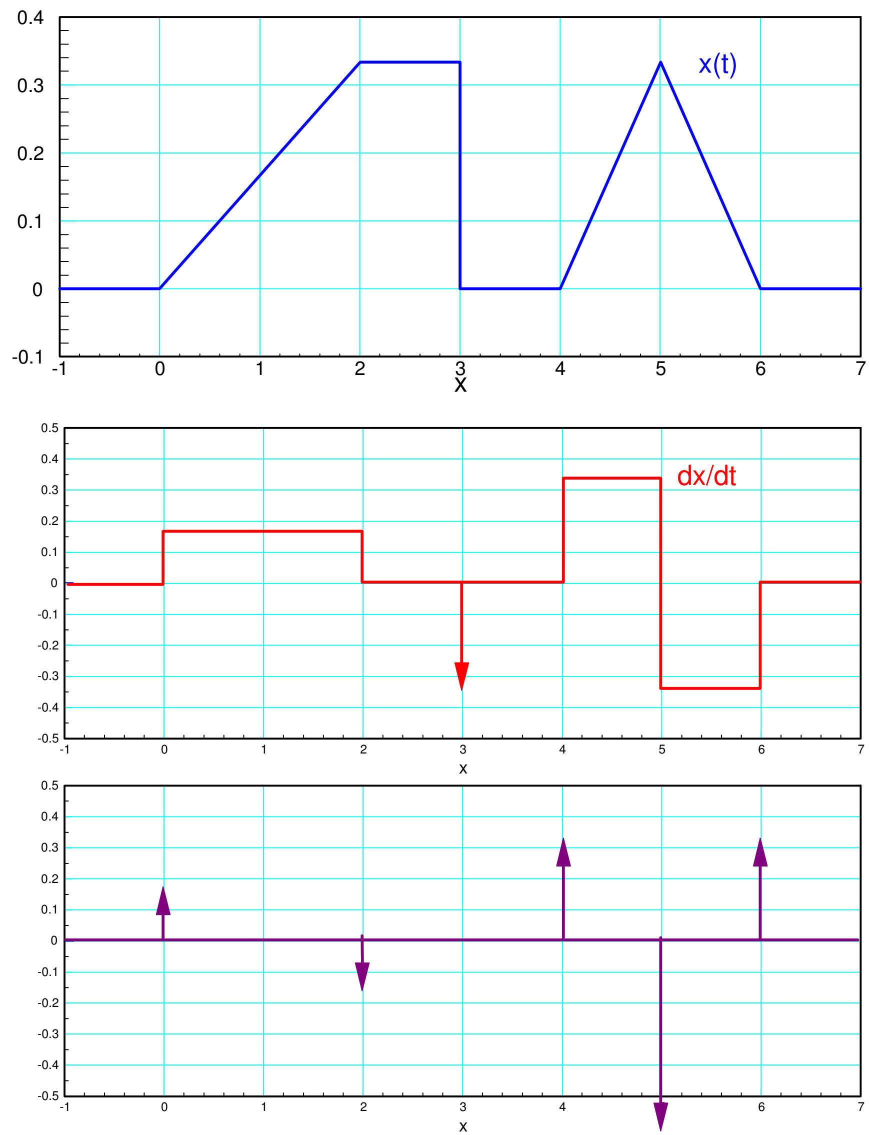


Take another derivative

- $L(p'') = 0.166 - 0.166e^{-2s}$
  $+0.333e^{-4s} - 0.666e^{-5s}$
  $+0.333e^{-6s}$

Integrate twice to get back to the pdf

$$X(s) = \left(\frac{0.166}{s^2}\right) - \left(\frac{0.166}{s^2}\right)e^{-2s}$$
$$+\left(\frac{0.333}{s^2}\right)e^{-4s} - \left(\frac{0.666}{s^2}\right)e^{-5s}$$
$$+\left(\frac{0.333}{s^2}\right)e^{0-6s}$$



The total answer is then

$$\psi(s) = L(p) + \frac{L(p')}{s} + \frac{L(p'')}{s^2}$$

$$\psi(s) = \left(\frac{-0.333e^{-3s}}{s}\right) + \left(\frac{0.166-0.166e^{-2s}+0.333e^{-4s}-0.666e^{-5s}+0.333e^{-6s}}{s^2}\right)$$

This is the moment generating function for the previous pdf

Summary

Moment Generating Functions are really just the LaPlace transform of the pdf

They are useful for

- Generating moments (duh), and
- Avoiding convolution of pdf's

The latter will show up in the following lectures.