Psy 5018H: Math Models Human Behavior
Spring 2006
Prof. Paul Schrater
Homework #1, Due Feb. 1th

Do the problem set below for credit. Please do the tutorial (not for credit) which follows the problem set as well.

Problem Set

Submit homework as a ".m" script file. Commands should be in order and the file should be executable. Answers to verbal questions in text should require no more than one or two sentences, and should be written as matlab comments.

Problem 1:

Dice difference: Dice difference is a game that involves two players. Two dice are thrown, and the absolute value of the difference computed. Player A wins if the value is {3,4,5}. Player B wins if the value is {0,1,2}. Write a matlab script to compute a table with the probability of each possible outcome. Is the game fair? Justify your answer. If not fair, can it be made fair?

Problem 2: You will look at Fechner's law for brightness. Download and execute the script colordiscrim.m You will see two windows filled with gray. Move the slider until the right window's color is just noticeably brighter than the left side and click the button. The colors will update. Repeat as many times as needed until the slider is all the way to its farthest right position. You have recorded the data needed to test Fechner's law for increments in color, as measured by your monitor output (this will not correspond directly to physical light units). Your brightness settings are stored in the variable jnds

after you collect data, run:
The list of just noticeable monitor increments is in jnds. Remove the last entry (e.g.) jnds(end)=[];
Now by hypothesis, each increment in monitor color came at a constant perceptual difference. So we can assume the perceptual increments are all

equal and we can set them to one. Try to test whether Fechner's law holds for your data by plotting. HINT- form the list of perceptual increments. Recover the list of monitor color values from the list of increments in jnds. Plot the appropriate combination of these quantities.

Problem 3: Your have found two kinds of plant in the forest that look identical, one type containing only a deadly poison and the other containing a small amount of poison and an amazing drug. The plants are extracted into a lovely pill called Amazidrug. Some people who take a dose die. It has been found that of those that died, 95% had ingested a Plant type 1dose. Of those that lived, 85% had taken plant type 1. 1 in a hundred thousand died after a single dose.

- Compute the probabilty of a dose containing Plant type1.
- Compute the probabilty of death given the dose is Plant type1.

Tutorial. Do the tutorial to familiarize yourself with matlab programming and some basic (and more advanced) math.

Open a new .m file and type a command like a=5. Execute the command (on a PC) by going to the debug menu and selecting Run.

```
% This is a comment

m=4*5; % this is a matlab command with a comment after it.
```

Where are you in the directory tree? Use
>>pwd

cd to a directory you like
If you want to make a directory for this homework
>> mkdir Homework1
>>cd Homework1
Only files in the current directory and in matlab's search path are visible. To add a directory to the search path, go to File: Set Path and look at the options. Or you can do:
>>path

Please ask questions if you don't understand any part of the assignment, by email or in class.

1) Matlab coding concepts

Colon

In MATLAB, the colon is very important. It constructs lists, but has several uses.

X = a:increment:b is the basic use. This constructs a list of numbers, beginning at a, taking steps of size increment, and ending at the closest increment less than b. For example

X = 0.1:0.5:2.2 yields
0.1 0.6 1.1 1.6 2.1

If you leave out the increment, MATLAB assumes it is 1
So
X = 1:1:10
Can be written more simply as
X = 1:10

We can use integer lists to pull out chucks of a matrix.
A = [1 2 3; 4 5 6; 7 8 9];

If we want the last column

Recall to index into a matrix, use A(rows,columns)

```
A(1:3,3)  = 3
            6
            9
```

When indexing a matrix, Matlab knows how many elements are in each dimension, so that the start and stop points can be left out, leaving only the colon – this means you want all the elements in that direction. So the above expression can also be written

A(:,3)

PRACTICE

Generate a vector x from 0.01 to 1 with increments of 0.01 using colon notation. Now construct y = 5*cos(2*pi*3* x). Use the size() command to determine the size of the vector. Do plot(x,y).

Grab the 26 through 75th element of y and put it in a vector w.
Create y2 = 5*sin(2*pi*x);
Do plot(y,y2);
Grab the 1st through 50th element of y2 and put it in a vector w2.
Do plot(w,w2).

Plotting vectors and loops

Download to a directory visible to matlab (check your current directory, using pwd) the function file arrow2.m

We will make some random 2D points:
>> points = randn(2,20)
inspect points- it is a list of 20 vectors. You can think of the first row as containing the x coordinates, and the second row the y coordinates. We can visualize them as points using plot-
plot can take a list of x coordinates and y coordinates, and put a symbol at each point . For example
plot(points(1,1:20),points(2,1:20),'p'), where the 'p' stands for pentagram (stars). Next let's visualize them as vectors. To do this we will use arrow2. Do:

>>help arrow2
You will see that the function takes start points and end points as input. Let [0,0] be the startpoint. Draw an arrow to the first point
>>arrow2([0,0],points(1:2,1))
How can we draw all of them? Using a FOR LOOP.
For Loops

```
>>for j=1:20, % read as: For each instance of j between 1 and 20
      arrow2([0,0],points(1:2,j)) % draw arrow to j^th point
end;  % End of loop.
```

Now draw them end to end:

```
      Points = [[0;0] points];
for j=1:20, % read as: For each instance of j between 1 and 20
      arrow2(Points(1:2,j),Points(1:2,j+1)) % draw arrow to j^th point
end;
```

Now do the same thing with the sinusoids we made before

```
Points = [[0;0] [y; y2]];
for j=1:100, % read as: For each instance of j between 1 and 20
      arrow2(Points(1:2,j),Points(1:2,j+1)) % draw arrow to j^th point
end;
```

In the above code we have drawn vectors between points. What are the vectors? For any point j,

```
      (For instance
      j = 3;
      V = Points(1:2,j+1) - Points(1:2,j);
```

This would be the coordinates of the third vector plotted.

Sorting and using index lists to grab parts of vectors

Type

>>help sort

Imagine you have a list of vectors and you want to sort them based on one of the dimensions.
Do:
[yn,I]=sort(y); Use the index vector I to rearrange y2 to match the sort on y. y2n = y2(I);

Now do
```
plot(y,y2);
hold on;  % command keeps previously plotted things in the picture
plot(yn,y2n,'rx');
hold off
```

Although y and y2 are sinusoids, they are also vectors.
Compute the dot product of y and y2 in four different ways.

1) Using the dot() command.
2) Using a for loop.
3) Using the sum() and .* (the pointwise multiplication operator).
4) Using matrix multiplication * and transpose ' operators.

2) Linear Algebra concepts

```
% Generate a random x vector.
rx = ceil(10*rand(1,100));
% Generate a random y vector.
ry = ceil(10*rand(1,100));
% do a scatter plot
scatter(rx,ry); axis([0 20 0 20]) % Enter the following matrix
Q = [ 1.2500    0.7500; 0.7500    1.2500]
% Q will transform our x and y vectors.  Let's transform all the
% points.  We could do it as a for loop:
for j=1:100
      temp = Q*[rx(j); ry(j)]
      rxtransform(j) = temp(1);
      rytransform(j) = temp(2);
end
scatter(rxtransform,rytransform);
% However, let's do it as a matrix multiply.
% First create a new matrix X by stacking rx and ry to form a 2 by 100 matrix.
% Then compute Y = Q*X.  Separate it back into rxtransform and rytransform by
% setting rxtransform and rytransform equal to the 1st and 2nd rows of Y
% respectively
% How can you undo a transformation?
%  Look at equation Y=Q*X .  If it were a simple equation will one dimensional
% variables, you could just do X = Y/Q = Y*(1/Q), or Q^(-1)
% There is a similar idea in Linear Algebra, except 1/Q has to be modified:
%  we define a matrix R, such that R*Q*X = X.   With this matrix,
%  R*Y = R*(Q*X) = X.   How to find R?  R = inv(Q); in matlab code.
```

3) Probability concepts

```
% Let's sample from a discrete probability distribution
% Plot the distribution
p = binopdf(1:6,7,0.5);
plot(1:6,p)
% The theoretical expectation is given by: theory_mean = sum(p.*(1:6));
% generate 100 samples from the distribution
r = binornd(7,0.5,100,1);
% compute the empirical expected value of r two ways
% 1) use the mean() function
% 2) generate a uniform weight vector
weight = ones(100,1)/100;
% we can also compute the mean as
emp_mean2 = sum(weight.*r);
% You can also compute the empirical average as:
%  weight'*r (Why does this work?)
% Let's bin the samples
N = hist(r,1:6);
p_empirical = N/sum(N);
plot(1:6,p_empirical)
% Finally we can compute the empirical mean as
emp_mean3 = sum(p_empirical.*(1:6))
```