Introduction to MATLAB

Practice Problems

May 18, 2008

1 Basics

1.1. Find 3 different ways to create the vector $x = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$. `>> x = (1:3)';`

```
>> x = [1 2 3]';
>> x = [1;2;3];
>> x = [1
        2
        3];
```

1.2. Let $x = [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8]^\top$ and $y = [90\ 100]$.

(a) Find $A = xy$ and $A^\top$.

```
>> x = (1:8)';
>> y = [90 100];
>> A = x*y;
A =
    90   100
   180   200
   270   300
   360   400
   450   500
   540   600
   630   700
   720   800
>>A' =
ans =
    90   180   270   360   450   540   630   720
   100   200   300   400   500   600   700   800
```

(b) What are the dimensions of A and $A^\top$?

```
>> size(A)
     8     2
>> size(A')
     2     8
```

2 Programming

2.1. Write a function which takes inputs $x,\ a,\ b,\ c$, where x is a vector and $a,\ b,\ c$ are scalars, and returns the values of

$$f(x) = ax^2 + bx + c$$

and

$$f'(x) = 2ax + b$$

```
function [f,fp] = ffp(x,a,b,c)
f = a*x.^2+b*x+c;
fp = 2*a*x+b;
```

2.2. Write a script that calls the previous function for 3 different sets of parameters $a,\ b,\ c$, over the range $-1 \leq x \leq 1$.

```
x = linspace(-1,1,101);
[f1,fp1] = ffp(x,1,0,0);
[f2,fp2] = ffp(x,1,4,4);
[f3,fp3] = ffp(x,0,3,1);
```

2.3. Create a function m-file to calculate $f = b\sin(x)$ for $x \in [-2\pi,\ 2\pi]$ with the stepsize$= 0.01$ where the input is b and the output is f. Run the program for $b = 2,\ 3,\ 5$.

```
function f = fsin(b)
x = -2*pi:0.01:2*pi;
f = b*sin(x);
```

```
>> f1 = fsin(2);
>> f2 = fsin(3);
>> f3 = fsin(5);
```

2.4. Write a function that creates the Fibonacci Sequence up to N numbers, where N is the input. Try different approaches with the '*for*' loop and '*while*' loop. Test your function for $N = 5,\ 10$. The Fibonacci sequence is defines by

$$F(n) = \begin{cases} 0 & \text{if } n = 0; \\ 1 & \text{if } n = 1; \\ F(n-1) + F(n-2) & \text{if } n > 1. \end{cases}$$

```
function fout = fibonacci(N)
if N < 1
    error('N needs to be a positive integer!');
end
fout=zeros(1,N);
fout(1) = 0;
if N > 1
```

```
    fout(2) = 1;
    if N > 2
        for i = 3:N
            fout(i) = fout(i-1)+fout(i-2);
        end
    end
end

>> fibonacci(5)
ans =
     0     1     1     2     3

>> fibonacci(10)
ans =
     0     1     1     2     3     5     8    13    21    34
```

3 Numerical Functions

3.1. Create a 300 element column vector where the even entries are 2 and the odd entries are 1.

```
>> x = ones(300,1);
>> x(2:2:300) = 2;
```

3.2. Create a 7×7 identity matrix and then change the $4^{th.}$ column to $[1\ 2\ 3\ 4\ 5\ 6\ 7]^\top$.

```
>> M = eye(7);
>> M(:,4) = (1:7);
>> M
M =
1 0 0 1 0 0 0
0 1 0 2 0 0 0
0 0 1 3 0 0 0
0 0 0 4 0 0 0
0 0 0 5 1 0 0
0 0 0 6 0 1 0
0 0 0 7 0 0 1
```

3.3. Let $A = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$, $b = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$

(a) Find A^{-1}.

```
>> A = [1 0 1;1 1 0; 0 1 1];
>> Ainv = inv(A)
Ainv =
0.5000 0.5000 -0.5000
```

```
-0.5000 0.5000 0.5000
0.5000 -0.5000 0.5000
```

(b) Solve the linear system $Ax = b$ for vector x.

```
>> b = ones(3,1);
>> x = A\b
x =
    0.5000
    0.5000
    0.5000
```

3.4. Write a script and all the necessary functions to solve the IVP for $0 \leq t \leq 2\pi$.

$$\begin{cases} y''(t) = -y(t) \\ y(0) = 0 \\ y'(0) = 1 \end{cases}$$

```
function dy = fivp(t,y)
M = [ 0 1
     -1 0];
dy = M*y;


[t,y] = ode45(@fivp,[0 2*pi],[0;1])
```

3.5. Use *fminsearch* to find a minimizer of x^2. Verify your result with the apparent solution $x = 0$.

```
function y = f(x)
y = x^2;


>> x = fminsearch(@f,10)
x =
     0
```

3.6. Use *fminsearch* to find a minimizer of $f(x) = \sin(\pi x_1)\sin(\pi x_2)$, $x = [x_1, x_2]^\top$. Try different initial guesses $\begin{cases} x_1 = -1/4 \\ x_2 = 1/4 \end{cases}$ and $\begin{cases} x_1 = 1/4 \\ x_2 = -1/4 \end{cases}$.

```
function fout = fsinsin(x)
fout = sin(pi*x(1))*sin(pi*x(2));

>> x = fminsearch(@fsinsin,[-1/4,1/4])
x =
   -0.5000    0.5000

>> x = fminsearch(@fsinsin,[1/4,-1/4])
```

```
x =
    0.5000   -0.5000
```

4 Inputs and Outputs

4.1. Plot the results (y and y') of 3.4 in one figure. Include a title, axis labels, and a legend.

```
>> plot(t,y(:,1),t,y(:,2));
>> title('Solutions to IVP');
>> xlabel('t');
>> ylabel('y and dy/dt');
>> legend('y','dy/dt');
```

4.2. Save all variables currently in memory into a mat-file, then clear them all.

```
>> save mydata;
>> clear;
```

4.3. Reload the variables you just saved.

```
>> load mydata;
```