ENGR 20 - MATLAB

Homework 2

Joseph Park

8/28/2024

Values from previous assignment (a-e):

```
aVec = [3.14 15 9 26]
```

```
aVec = 1×4
    3.1400   15.0000    9.0000   26.0000
```

```
bVec = [10; 4; 19.4; exp(2)]
```

```
bVec = 4×1
   10.0000
    4.0000
   19.4000
    7.3891
```

```
aMat = 2 * ones(8)
```

```
aMat = 8×8
     2     2     2     2     2     2     2     2
     2     2     2     2     2     2     2     2
     2     2     2     2     2     2     2     2
     2     2     2     2     2     2     2     2
     2     2     2     2     2     2     2     2
     2     2     2     2     2     2     2     2
     2     2     2     2     2     2     2     2
     2     2     2     2     2     2     2     2
```

```
bMat = diag([1 2 3 4 5 4 3 2 1])
```

```
bMat = 9×9
     1     0     0     0     0     0     0     0     0
     0     2     0     0     0     0     0     0     0
     0     0     3     0     0     0     0     0     0
     0     0     0     4     0     0     0     0     0
     0     0     0     0     5     0     0     0     0
     0     0     0     0     0     4     0     0     0
     0     0     0     0     0     0     3     0     0
     0     0     0     0     0     0     0     2     0
     0     0     0     0     0     0     0     0     1
```

```
cMat = reshape(1:100, [10,10])
```

```
cMat = 10×10
     1    11    21    31    41    51    61    71    81    91
     2    12    22    32    42    52    62    72    82    92
     3    13    23    33    43    53    63    73    83    93
     4    14    24    34    44    54    64    74    84    94
     5    15    25    35    45    55    65    75    85    95
     6    16    26    36    46    56    66    76    86    96
     7    17    27    37    47    57    67    77    87    97
     8    18    28    38    48    58    68    78    88    98
     9    19    29    39    49    59    69    79    89    99
    10    20    30    40    50    60    70    80    90   100
```

1. Compute the following (a-c):

```
xMat = (aVec * bVec) * aMat^2
```

```
xMat = 8×8
10^4 ×
    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660
    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660
    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660
    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660
    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660
    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660
    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660
    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660    1.4660
```

```
yMat = (bVec * aVec)
```

```
yMat = 4×4
   31.4000  150.0000   90.0000  260.0000
   12.5600   60.0000   36.0000  104.0000
   60.9160  291.0000  174.6000  504.4000
   23.2016  110.8358   66.5015  192.1155
```

```
disp(yMat == aVec * bVec)
```

```
   0   0   0   0
   0   0   0   0
   0   0   0   0
   0   0   0   0
```

The matrix bVec * aVec is not the same as aVec * bVec. The boolean statement above shows that the statement is false. Additionally, matrix multiplication is not commutative; A*B does not equal B*A generally.

The equation for 1c is not computable becaues the sizes of the matrices are not compatible for multiplication.

2. Functions and applications (a-c):

```
cSum = sum(cMat)
```

```
cSum = 1×10
    55   155   255   355   455   555   655   755   855   955
```

```
bMean = mean(bMat,2)
```

```
bMean = 9×1
    0.1111
    0.2222
    0.3333
    0.4444
    0.5556
    0.4444
    0.3333
    0.2222
    0.1111
```

```
cSub = cMat(2:9,2:9)
```

```
cSub = 8×8
    12    22    32    42    52    62    72    82
    13    23    33    43    53    63    73    83
```

```
    14    24    34    44    54    64    74    84
    15    25    35    45    55    65    75    85
    16    26    36    46    56    66    76    86
    17    27    37    47    57    67    77    87
    18    28    38    48    58    68    78    88
    19    29    39    49    59    69    79    89
```

Problems from *An Introduction to MATLAB...* by Troy Siemers

Problem 2.2 (a-g):

```
h5 = hilb(5)
```

```
h5 = 5×5
    1.0000    0.5000    0.3333    0.2500    0.2000
    0.5000    0.3333    0.2500    0.2000    0.1667
    0.3333    0.2500    0.2000    0.1667    0.1429
    0.2500    0.2000    0.1667    0.1429    0.1250
    0.2000    0.1667    0.1429    0.1250    0.1111
```

```
det(h5)
```

```
ans = 3.7493e-12
```

```
transpose(h5)
```

```
ans = 5×5
    1.0000    0.5000    0.3333    0.2500    0.2000
    0.5000    0.3333    0.2500    0.2000    0.1667
    0.3333    0.2500    0.2000    0.1667    0.1429
    0.2500    0.2000    0.1667    0.1429    0.1250
    0.2000    0.1667    0.1429    0.1250    0.1111
```

```
inv(h5)
```

```
ans = 5×5
10^5 ×
    0.0002   -0.0030    0.0105   -0.0140    0.0063
   -0.0030    0.0480   -0.1890    0.2688   -0.1260
    0.0105   -0.1890    0.7938   -1.1760    0.5670
   -0.0140    0.2688   -1.1760    1.7920   -0.8820
    0.0063   -0.1260    0.5670   -0.8820    0.4410
```

```
size(h5)
```

```
ans = 1×2
     5     5
```

```
sum(h5)
```

```
ans = 1×5
    2.2833    1.4500    1.0929    0.8845    0.7456
```

```
sum(h5, 2)
```

```
ans = 5×1
    2.2833
    1.4500
    1.0929
    0.8845
```

```
    0.7456
```

```
max(max(h5))
```

```
ans = 1
```

```
eig(h5)
```

```
ans = 5×1
    0.0000
    0.0003
    0.0114
    0.2085
    1.5671
```

```
h5^2
```

```
ans = 5×5
    1.4636    0.8333    0.5952    0.4663    0.3844
    0.8333    0.4914    0.3571    0.2827    0.2348
    0.5952    0.3571    0.2618    0.2083    0.1736
    0.4663    0.2827    0.2083    0.1663    0.1389
    0.3844    0.2348    0.1736    0.1389    0.1162
```

```
h5.^2
```

```
ans = 5×5
    1.0000    0.2500    0.1111    0.0625    0.0400
    0.2500    0.1111    0.0625    0.0400    0.0278
    0.1111    0.0625    0.0400    0.0278    0.0204
    0.0625    0.0400    0.0278    0.0204    0.0156
    0.0400    0.0278    0.0204    0.0156    0.0123
```

```
h5./h5
```

```
ans = 5×5
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
     1     1     1     1     1
```

The equivalent expression to h5^2 is h5 * h5. It takes the dot product of the same square matrix.

In h5.^2, each element in the matrix is squared. For example, h5(1,1), the first element in the matrix, is 1. If you apply the operation h5.^2, the result is 1^1 or 1.

Finally, the operation ./ takes each element in the first matrix and divides it by the element in the same position in the second matrix. Since the elements are the same in both matrices, the result will always end up as 1.

```
h5(2:3, :)
```

```
ans = 2×5
    0.5000    0.3333    0.2500    0.2000    0.1667
    0.3333    0.2500    0.2000    0.1667    0.1429
```

Problem 2.3:

```
b = [-2; 3];
```

```
for i = [30, 90, 200]
    A = [cosd(i) -sind(i); sind(i) cosd(i)];
    disp("When theta is: " + i + " degrees")
    disp(A*b)
end
```

```
When theta is: 30 degrees
   -3.2321
    1.5981
When theta is: 90 degrees
    -3
    -2
When theta is: 200 degrees
    2.9054
   -2.1350
```

Problem 2.6 (a-c):

```
splineA = [.28 .1 0 0 0 ; .1 .34 .07 0 0 ; 0 .07 2.16 1.01 0 ; 0 0 1.01 2.58 0.28 ;
0 0 0 0.28 1.42 ]
```

```
splineA = 5×5
    0.2800    0.1000         0         0         0
    0.1000    0.3400    0.0700         0         0
         0    0.0700    2.1600    1.0100         0
         0         0    1.0100    2.5800    0.2800
         0         0         0    0.2800    1.4200
```

```
splineB = [-64.65; -54.81; -8.43; -7.92; -2.78]
```

```
splineB = 5×1
  -64.6500
  -54.8100
   -8.4300
   -7.9200
   -2.7800
```

```
rref_spline = rref([splineA splineB])
```

```
rref_spline = 5×6
    1.0000         0         0         0         0 -193.5763
         0    1.0000         0         0         0 -104.4865
         0         0    1.0000         0         0    1.0439
         0         0         0    1.0000         0   -3.3373
         0         0         0         0    1.0000   -1.2997
```

```
rref_spline(:,end)
```

```
ans = 5×1
 -193.5763
 -104.4865
    1.0439
   -3.3373
   -1.2997
```

```
ldivide_spline = splineA \ splineB
```

```
ldivide_spline = 5×1
 -193.5762
 -104.4866
```

```
    1.0439
   -3.3374
   -1.2997
```

```
inverse_spline = inv(splineA) * splineB
```

```
inverse_spline = 5×1
 -193.5762
 -104.4866
    1.0439
   -3.3374
   -1.2997
```