Prerrequisitos

Table of Contents

Sección I - Introducción a MATLAB 1
I. MANIPULANDO MATRICES. 10
II. VECTORIZANDO OPERACIONES. 15
III. OTRAS OPERACIONES ÚTILES. 17

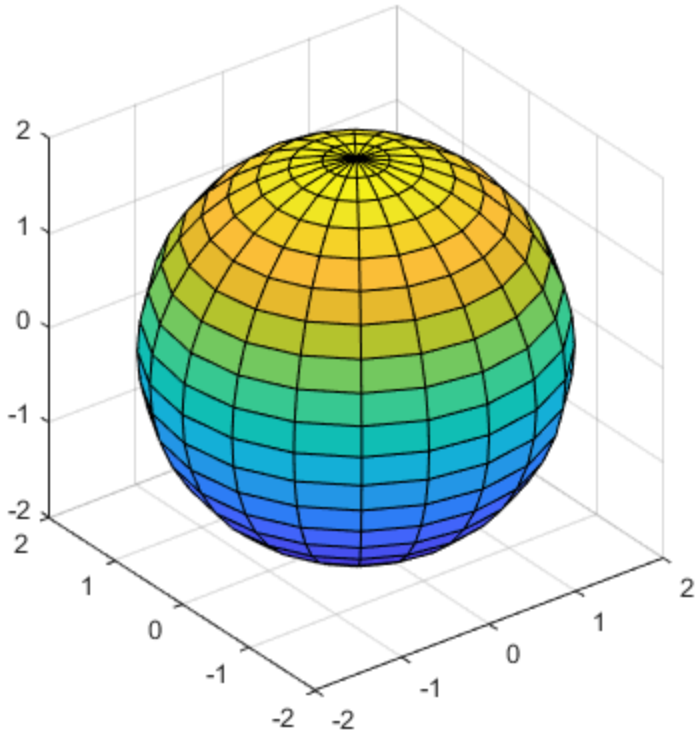
1092228, Ian Grabriel Cañas Fernández

Sección I - Introducción a MATLAB

Programming and scripts

Grafica una esfera de radio r.

```
[x,y,z] = sphere;       % Esfera unitaria.
r = 2;
surf(x*r,y*r,z*r)       % ajusta cada dimensión y grafica.
axis equal              % Usa una escala igual para los ejes.

% Encuentra el área y volumen.
A = 4*pi*r^2;
V = (4/3)*pi*r^3;
```




```
N = 100;
f(1) = 1;
f(2) = 1;

for n = 3:N
    f(n) = f(n-1) + f(n-2);
end
f(1:10)

num = randi(100)
if num < 34
   sz = 'low'
elseif num < 67
   sz = 'medium'
else
   sz = 'high'
end


ans =

     1     1     2     3     5     8    13    21    34    55


num =
```

```
    98


sz =

    'high'
```

Matrices and arrays

```
a = [1 2 3 4]
a = [1 3 5; 2 4 6; 7 8 10]
z = zeros(5,1)


a =

     1     2     3     4


a =

     1     3     5
     2     4     6
     7     8    10


z =

     0
     0
     0
     0
     0


a + 10
sin(a)
a'
p = a*inv(a)

format long
p = a*inv(a)

format short
p = a.*a

a.^3


ans =

    11    13    15
    12    14    16
```

```
    17    18    20


ans =

    0.8415    0.1411   -0.9589
    0.9093   -0.7568   -0.2794
    0.6570    0.9894   -0.5440


ans =

     1     2     7
     3     4     8
     5     6    10


p =

    1.0000         0         0
         0    1.0000         0
         0         0    1.0000


p =

   0.999999999999996                   0                   0
                   0   1.000000000000000                   0
                   0                   0   0.999999999999998


p =

     1     9    25
     4    16    36
    49    64   100


ans =

           1          27         125
           8          64         216
         343         512        1000


A = [a,a]
A = [a; a]


A =

     1     3     5     1     3     5
     2     4     6     2     4     6
     7     8    10     7     8    10
```

```
A =

     1     3     5
     2     4     6
     7     8    10
     1     3     5
     2     4     6
     7     8    10


sqrt(-1)
c = [3+4i, 4+3j; -i, 10j]


ans =

   0.0000 + 1.0000i


c =

   3.0000 + 4.0000i   4.0000 + 3.0000i
  -5.0000 + 0.0000i   0.0000 +10.0000i
```

Array indexing

```
A = [1 2 3 4; 5 6 7 8; 9 10 11 12; 13 14 15 16]
A(4,2)
A(8)
A(4,5) = 17
A(1:3,2)
A(3,:)
B = 0:10:100


A =

     1     2     3     4
     5     6     7     8
     9    10    11    12
    13    14    15    16


ans =

    14


ans =

    14
```

```
A =

     1     2     3     4     0
     5     6     7     8     0
     9    10    11    12     0
    13    14    15    16    17


ans =

     2
     6
    10


ans =

     9    10    11    12     0


B =

     0    10    20    30    40    50    60    70    80    90   100
```

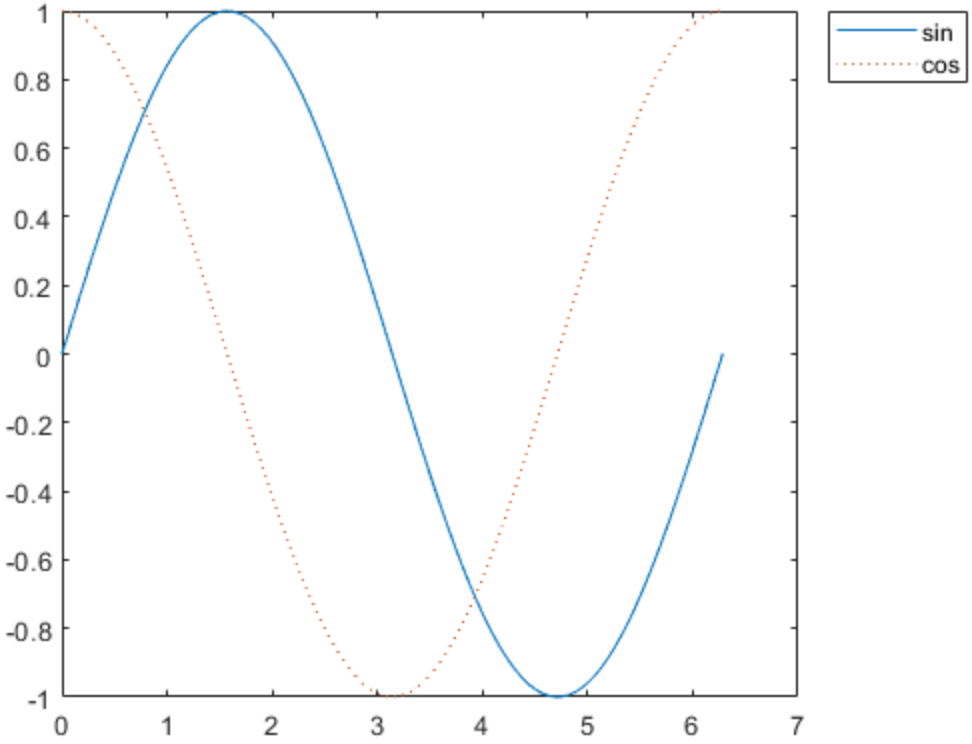
Calling functions

```
A = [1 3 5];
max(A)

B = [3 6 9];
union(A,B)

maxA = max(A)

[minA,maxA] = bounds(A)

disp("hello world")


ans =

     5


ans =

     1     3     5     6     9


maxA =
```

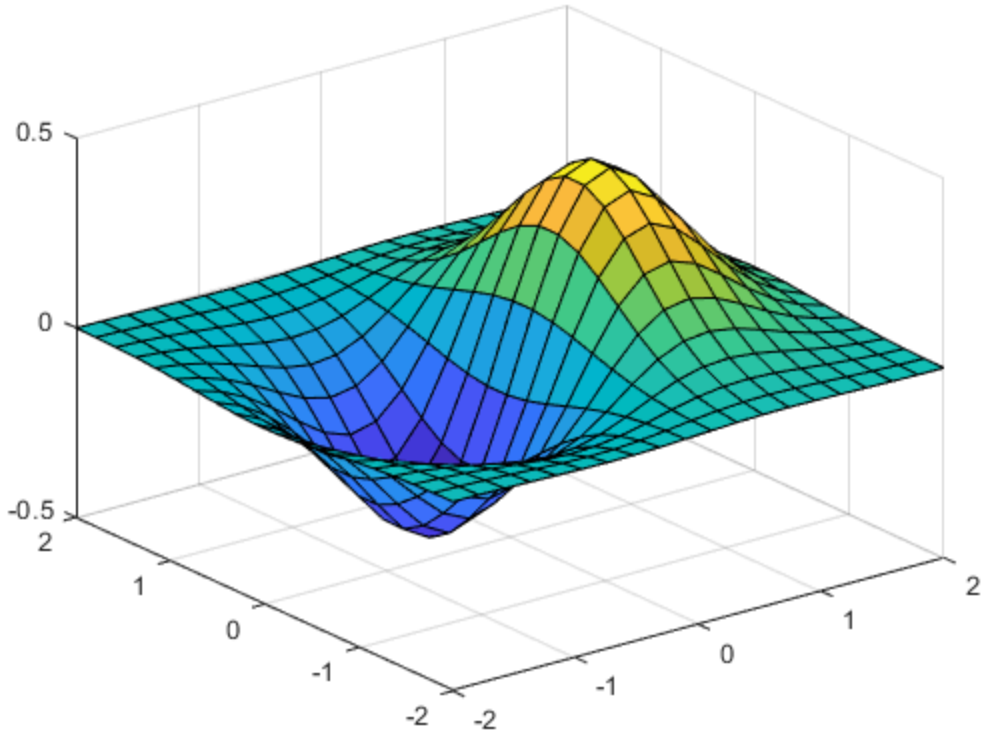
```
     5


minA =

     1


maxA =

     5

hello world
```

2-D and 3-D plots

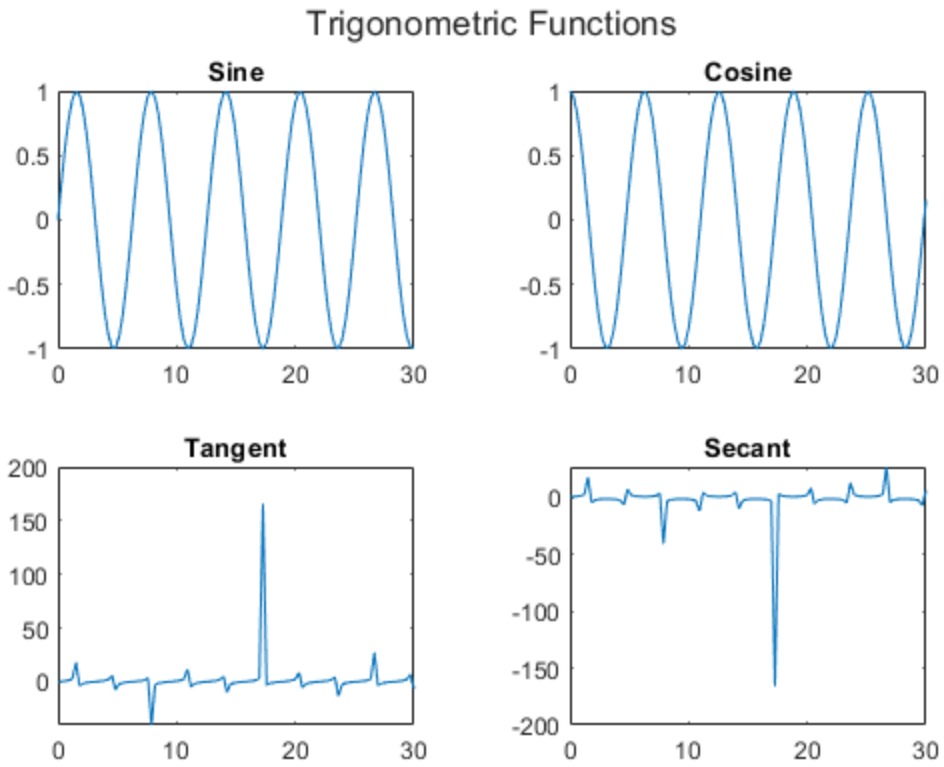
```
x = linspace(0,2*pi);
y = sin(x);
plot(x,y)

xlabel("x")
ylabel("sin(x)")
title("Plot of the Sine Function")

plot(x,y,"r--")

x = linspace(0,2*pi);
y = sin(x);
plot(x,y)

hold on

y2 = cos(x);
plot(x,y2,":")
legend("sin","cos")

hold off
```




```
x = linspace(-2,2,20);
y = x';
z = x .* exp(-x.^2 - y.^2);

surf(x,y,z)
```




```
t = tiledlayout(2,2);
title(t,"Trigonometric Functions")
x = linspace(0,30);

nexttile
plot(x,sin(x))
title("Sine")

nexttile
plot(x,cos(x))
title("Cosine")

nexttile
plot(x,tan(x))
title("Tangent")

nexttile
plot(x,sec(x))
title("Secant")
```



I. MANIPULANDO MATRICES.

```
clear
clc
```

1) Realice las transformaciones lineales $(a) q = Qx$ y $(b) x = Q^{-1} q$, donde Comente sus observaciones.

```
format rational
Q = [5/6 1/6 0; 5/6 0 1/6; 0 5/6 1/6]
format default
x = [56; 14; 0.005]

q = Q*x
x = inv(Q)*q
```

```
Q =

       5/6            1/6            0
       5/6            0              1/6
       0              5/6            1/6


x =

   56.0000
```

```
   14.0000
    0.0050


q =

   49.0000
   46.6675
   11.6675


x =

   56.0000
   14.0000
    0.0050
```

Claramente apreciamos el cumplimiento de los conceptos de álgebra lineal en cuando al producto y la matriz inversa.

2) Importe los datos de un archivo .csv, .xls o .xlsx como una matrix a su plataforma de preferencia y despliegue las primeras 15 filas y 4 columnas.

```
A = readmatrix('Matriz.xlsx');
A(1:15,1:4)
```

```
ans =

    1.0000    1.0000    1.0000    1.0000
    3.0000    2.0000    1.0000         0
    2.0000    3.0000    4.0000    5.0000
    4.0000    4.0000    4.0000    4.0000
    3.0000    5.0000    7.0000    9.0000
    5.0000    6.0000    7.0000    8.0000
    5.2000    7.0000    8.8000   10.6000
    5.8286    8.0000   10.1714   12.3429
    6.4571    9.0000   11.5429   14.0857
    7.0857   10.0000   12.9143   15.8286
    7.7143   11.0000   14.2857   17.5714
    8.3429   12.0000   15.6571   19.3143
    8.9714   13.0000   17.0286   21.0571
    9.6000   14.0000   18.4000   22.8000
   10.2286   15.0000   19.7714   24.5429
```

3) Cree un nuevo vector columna $x = [x^1 x^2 \dots x^{20}]$ y copie allí el contenido de los últimos 20 elementos de la tercera columna de la matriz creada en el paso anterior.

```
X = A(end-19:end,3)
```

```
X =

   41.7143
```

```
   43.0857
   44.4571
   45.8286
   47.2000
   48.5714
   49.9429
   51.3143
   52.6857
   54.0571
   55.4286
   56.8000
   58.1714
   59.5429
   60.9143
   62.2857
   63.6571
   65.0286
   66.4000
   67.7714
```

4) Cree un nuevo vector fila $y = [y^1 y^2 \dots y^5]^T$, donde cada $y^i \sim \mathcal{N}(1.7, 5)$ es un número aleatorio generado según la distribución Gaussiana especificada.

```
Y = transpose(normrnd(1.7, 5, 5, 1)) % normrnd(mu,sigma,sz) genera un array
 de números rándom de distribución normal, donde el vector sz especifica el
 tamaño de la matriz Y.
```

```
Y =

    0.5519   -5.8308   -0.5231    0.9203    3.0803
```

*5) Genere la matriz A = xy.

```
A = X*Y
```

```
A =

   23.0208 -243.2276  -21.8224   38.3894  128.4942
   23.7776 -251.2241  -22.5398   39.6516  132.7187
   24.5345 -259.2206  -23.2573   40.9137  136.9432
   25.2913 -267.2172  -23.9747   42.1758  141.1676
   26.0482 -275.2137  -24.6922   43.4379  145.3921
   26.8050 -283.2102  -25.4096   44.7000  149.6166
   27.5619 -291.2067  -26.1271   45.9622  153.8410
   28.3187 -299.2033  -26.8445   47.2243  158.0655
   29.0755 -307.1998  -27.5620   48.4864  162.2900
   29.8324 -315.1963  -28.2794   49.7485  166.5144
   30.5892 -323.1928  -28.9969   51.0106  170.7389
   31.3461 -331.1894  -29.7143   52.2727  174.9634
   32.1029 -339.1859  -30.4317   53.5349  179.1879
```

```
   32.8598 -347.1824  -31.1492   54.7970  183.4123
   33.6166 -355.1789  -31.8666   56.0591  187.6368
   34.3735 -363.1755  -32.5841   57.3212  191.8613
   35.1303 -371.1720  -33.3015   58.5833  196.0857
   35.8872 -379.1685  -34.0190   59.8455  200.3102
   36.6440 -387.1650  -34.7364   61.1076  204.5347
   37.4009 -395.1615  -35.4539   62.3697  208.7591
```

6) Aplique un comando de reshape para convertir la matriz A en un s´olo vector columna a

```
a = reshape(A, [], 1)
```

```
a =

   23.0208
   23.7776
   24.5345
   25.2913
   26.0482
   26.8050
   27.5619
   28.3187
   29.0755
   29.8324
   30.5892
   31.3461
   32.1029
   32.8598
   33.6166
   34.3735
   35.1303
   35.8872
   36.6440
   37.4009
 -243.2276
 -251.2241
 -259.2206
 -267.2172
 -275.2137
 -283.2102
 -291.2067
 -299.2033
 -307.1998
 -315.1963
 -323.1928
 -331.1894
 -339.1859
 -347.1824
 -355.1789
 -363.1755
 -371.1720
 -379.1685
```

```
-387.1650
-395.1615
 -21.8224
 -22.5398
 -23.2573
 -23.9747
 -24.6922
 -25.4096
 -26.1271
 -26.8445
 -27.5620
 -28.2794
 -28.9969
 -29.7143
 -30.4317
 -31.1492
 -31.8666
 -32.5841
 -33.3015
 -34.0190
 -34.7364
 -35.4539
  38.3894
  39.6516
  40.9137
  42.1758
  43.4379
  44.7000
  45.9622
  47.2243
  48.4864
  49.7485
  51.0106
  52.2727
  53.5349
  54.7970
  56.0591
  57.3212
  58.5833
  59.8455
  61.1076
  62.3697
 128.4942
 132.7187
 136.9432
 141.1676
 145.3921
 149.6166
 153.8410
 158.0655
 162.2900
 166.5144
 170.7389
 174.9634
```

```
  179.1879
  183.4123
  187.6368
  191.8613
  196.0857
  200.3102
  204.5347
  208.7591
```

II. VECTORIZANDO OPERACIONES.

1) Compute la suma de los elementos de cada columna de la matriz A.

```
S = sum(A)
```

```
S =

   1.0e+03 *

    0.6042   -6.3839   -0.5728    1.0076    3.3725
```

2) Compute el promedio de los elementos de cada columna de la matriz A.

```
P = median(A)
```

```
P =

   30.2108 -319.1946  -28.6381   50.3796  168.6267
```

3) Compute la varianza de los elementos de cada columna de la matriz A.

```
va = var(A)
```

```
va =

   1.0e+03 *

    0.0200    2.2381    0.0180    0.0558    0.6246
```

4) Compute el vector media a a partir de los vectores fila de la matriz A.

```
a = median(transpose(A))
```

```
a =

  Columns 1 through 7
```

```
   23.0208   23.7776   24.5345   25.2913   26.0482   26.8050   27.5619

  Columns 8 through 14

   28.3187   29.0755   29.8324   30.5892   31.3461   32.1029   32.8598

  Columns 15 through 20

   33.6166   34.3735   35.1303   35.8872   36.6440   37.4009
```

5) Compute la distancia euclideana entre a y cada vector fila de la matriz A.

```
d = pdist2(transpose(A),a,"euclidean")
```

```
d =

   1.0e+03 *

         0
    1.5788
    0.2659
    0.0911
    0.6254
```

6) Vectorice e implemente las siguientes combinaciones lineales. Asuma los valores de su preferencia para los coeficientes a^i, a^{ji} y los descriptores x^i.

```
za = a(1)
for i = 2:5
    za = za + X(i)*a(i);
end

za
```

```
za =

   23.0208


za =

   4.5268e+03
```

```
for j=1:3
    zb(j) = A(j);
    for i = 2:5
        zb(j) = zb(j) + X(i)*A(j,i);
    end
end
zb
```

```
zb =

   1.0e+03 *

   -3.6025   -3.7210   -3.8394
```

III. OTRAS OPERACIONES ÚTILES.

1) Aplique una función de slicing para extraer una matriz C de tamaño 4x4 a partir de las primeras 4 filas y 4 columnas de la matriz A.

```
C = A(1:4,1:4)
```

```
C =

   23.0208 -243.2276  -21.8224   38.3894
   23.7776 -251.2241  -22.5398   39.6516
   24.5345 -259.2206  -23.2573   40.9137
   25.2913 -267.2172  -23.9747   42.1758
```

2) Compute la matriz de eigenvalues λ y la matriz de eigenvectors V de la matriz C. Regenere a $C = V\lambda V^{-1}$

```
[V, lambda] = eig(C) % V = eigenvectors; lambda = eigenvalues

C = V*lambda*inv(V)
```

```
V =

    0.9837    0.4762    0.3759    0.9031
    0.1007    0.4919    0.1083    0.1579
    0.1039    0.5075    0.5225   -0.2145
    0.1071    0.5232    0.7576    0.3368
```

```
lambda =

         0         0         0         0
         0 -209.2848         0         0
         0         0    0.0000         0
         0         0         0   -0.0000
```

```
C =

   23.0208 -243.2276  -21.8224   38.3894
   23.7776 -251.2241  -22.5398   39.6516
   24.5345 -259.2206  -23.2573   40.9137
   25.2913 -267.2172  -23.9747   42.1758
```

3) Lleve a cabo la operación de singular value decomposition sobre las matrices A y C. Regenere las matrices A y C utilizando las matrices resultantes de la descomposición.

```
[UA,SA,VA] = svd(A)
A = UA*SA*VA
```

```
UA =

  Columns 1 through 7

   -0.1686    0.7458    0.5879    0.2087    0.0522    0.0156   -0.0572
   -0.1742   -0.0208    0.0701   -0.2170    0.3624    0.0547    0.3457
   -0.1797    0.1061   -0.1797    0.0243    0.0374   -0.0435   -0.1091
   -0.1853    0.2405   -0.3054   -0.1360    0.1121   -0.0441   -0.1217
   -0.1908   -0.0236   -0.1065    0.1631    0.1042   -0.3616   -0.1232
   -0.1964   -0.1345   -0.0127    0.2846   -0.0542    0.8797   -0.0602
   -0.2019    0.0197    0.0072    0.0699   -0.3767   -0.0725    0.8514
   -0.2074   -0.2568    0.1847    0.0216    0.4123   -0.0376    0.0629
   -0.2130    0.0507    0.0994   -0.3706   -0.0607    0.0721   -0.0529
   -0.2185   -0.2135    0.2983   -0.0715   -0.0686   -0.0643   -0.0946
   -0.2241    0.3238   -0.4105   -0.0225    0.0384    0.0733    0.0379
   -0.2296    0.0463   -0.1110    0.2092    0.0390   -0.0513   -0.0070
   -0.2352   -0.0496   -0.0840    0.1983   -0.4858   -0.1319   -0.1942
   -0.2407    0.0921   -0.0855   -0.3637   -0.0114    0.0872   -0.0295
   -0.2463   -0.1735    0.0796    0.1352    0.3387   -0.0549    0.0502
   -0.2518   -0.0494    0.1221   -0.4921   -0.2941    0.0474   -0.1521
   -0.2573   -0.1484    0.0815   -0.1035   -0.1030   -0.0445   -0.0966
   -0.2629   -0.0230   -0.2022    0.3376   -0.0700   -0.1108   -0.0527
   -0.2684   -0.2599    0.2665    0.1807   -0.0755   -0.1388   -0.1053
   -0.2740    0.0213   -0.2105   -0.0355    0.2375    0.0220    0.0462

  Columns 8 through 14

    0.0206   -0.0604   -0.0478   -0.0262   -0.0520    0.0215    0.0139
   -0.3590    0.0397    0.1151   -0.1601   -0.0780    0.4468   -0.0423
   -0.1529   -0.3263   -0.1405   -0.2936   -0.0985   -0.1359   -0.4311
    0.3632    0.1585    0.3289   -0.4354   -0.1677   -0.2075    0.0294
   -0.2303    0.2337   -0.1608    0.1522   -0.2048   -0.2683    0.2574
   -0.0367    0.0729   -0.0162   -0.0317   -0.0984   -0.1107    0.0620
    0.1112    0.0012   -0.0179   -0.0509   -0.0545   -0.2006    0.0021
    0.7440   -0.0689   -0.1535    0.1078   -0.0097    0.0974   -0.0365
   -0.0174    0.7887   -0.1295    0.0262    0.0650   -0.0057   -0.1769
   -0.0801   -0.0818    0.8230    0.1257    0.0019   -0.1056   -0.0262
    0.0496   -0.0685    0.1151    0.7606   -0.0400    0.0727   -0.1346
   -0.0246    0.0361    0.0320   -0.0923    0.9217   -0.0207    0.0101
    0.1319    0.0428   -0.0024   -0.0966   -0.1069    0.7168    0.0259
   -0.0248   -0.2285   -0.0923   -0.0565    0.0451    0.0248    0.7766
   -0.2044   -0.0221   -0.0916    0.0366   -0.0467    0.0679   -0.0122
    0.0233   -0.2776   -0.1998    0.0397    0.0689   -0.1284   -0.2304
   -0.0542   -0.1170   -0.1379    0.0228   -0.0170   -0.1049   -0.0936
   -0.0039    0.0777    0.0475   -0.1379   -0.1306   -0.1033    0.0338
   -0.0859    0.0177   -0.1389    0.0967   -0.0649   -0.1517    0.0577
   -0.1211   -0.1099   -0.0286   -0.1085   -0.0359    0.0852   -0.1392
```

```
  Columns 15 through 20

   -0.0042    0.0251    0.0372    0.0268   -0.0537    0.0619
   -0.3153    0.2768    0.1049    0.0231    0.1229   -0.2896
   -0.1391   -0.4621   -0.2871   -0.1254   -0.0607   -0.3518
    0.1857    0.2294    0.1613   -0.2534    0.2400   -0.0704
   -0.2797    0.2512   -0.0977   -0.3499   -0.3965   -0.0304
   -0.0768    0.0918   -0.0200   -0.1489   -0.0986   -0.0382
    0.0607   -0.0496   -0.0404   -0.1066   -0.0580    0.0277
   -0.1933   -0.0544   -0.0917    0.0248   -0.1480   -0.0742
    0.0313   -0.2990   -0.1167    0.1122   -0.0334   -0.0032
   -0.0520   -0.1564   -0.1251   -0.0091   -0.1812    0.0322
    0.0214   -0.0191    0.0384   -0.0051    0.1589   -0.1274
   -0.0573    0.0820    0.0129   -0.0920   -0.0101   -0.0670
    0.0500   -0.0087   -0.0463   -0.1958   -0.0844    0.0063
    0.0169   -0.2975   -0.1096    0.0993    0.0140   -0.0651
    0.8253    0.0167   -0.0546   -0.0291   -0.1121   -0.0875
    0.0698    0.5580   -0.1947    0.0925   -0.0941    0.0156
   -0.0394   -0.1895    0.8728   -0.0253   -0.1250   -0.0268
   -0.0679    0.1229    0.0071    0.8188   -0.0365   -0.0884
   -0.0935   -0.0166   -0.0926   -0.1113    0.7896    0.0167
   -0.1020   -0.0849   -0.0521    0.0031    0.0093    0.8557


SA =

   1.0e+03 *

    1.6577         0         0         0         0
         0    0.0000         0         0         0
         0         0    0.0000         0         0
         0         0         0    0.0000         0
         0         0         0         0    0.0000
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0


VA =

   -0.0823    0.0204    0.1964   -0.5376   -0.8156
```

```
    0.8701   -0.4141    0.2637   -0.0439   -0.0057
    0.0781   -0.3004   -0.8082   -0.4879    0.1115
   -0.1373    0.0609    0.4490   -0.6736    0.5675
   -0.4596   -0.8568    0.1925    0.1316   -0.0154


A =

   23.0208   -5.6934  -54.9181  150.2849  228.0053
   23.7776   -5.8806  -56.7236  155.2258  235.5014
   24.5345   -6.0678  -58.5292  160.1666  242.9975
   25.2913   -6.2550  -60.3347  165.1075  250.4935
   26.0482   -6.4422  -62.1402  170.0484  257.9896
   26.8050   -6.6293  -63.9457  174.9893  265.4857
   27.5619   -6.8165  -65.7513  179.9301  272.9817
   28.3187   -7.0037  -67.5568  184.8710  280.4778
   29.0755   -7.1909  -69.3623  189.8119  287.9739
   29.8324   -7.3781  -71.1678  194.7527  295.4699
   30.5892   -7.5653  -72.9734  199.6936  302.9660
   31.3461   -7.7524  -74.7789  204.6345  310.4621
   32.1029   -7.9396  -76.5844  209.5754  317.9581
   32.8598   -8.1268  -78.3899  214.5162  325.4542
   33.6166   -8.3140  -80.1955  219.4571  332.9503
   34.3735   -8.5012  -82.0010  224.3980  340.4463
   35.1303   -8.6883  -83.8065  229.3389  347.9424
   35.8872   -8.8755  -85.6121  234.2797  355.4385
   36.6440   -9.0627  -87.4176  239.2206  362.9345
   37.4009   -9.2499  -89.2231  244.1615  370.4306


[UC,SC,VC] = svd(C)
C = UC*SC*VC


UC =

   -0.4762    0.3772    0.7072   -0.3617
   -0.4919   -0.7870    0.2359    0.2881
   -0.5075   -0.0864   -0.6059   -0.6065
   -0.5232    0.4805   -0.2777    0.6468


SC =

  521.3532         0         0         0
         0    0.0000         0         0
         0         0    0.0000         0
         0         0         0    0.0000


VC =

   -0.0927   -0.3430   -0.8432    0.4034
    0.9797   -0.0173   -0.0051    0.1998
```

```
    0.0879    0.7720   -0.5041   -0.3771
   -0.1546    0.5348    0.1867    0.8094


C =

   23.0208   85.1598  209.3477 -100.1548
   23.7776   87.9596  216.2303 -103.4475
   24.5345   90.7594  223.1130 -106.7403
   25.2913   93.5592  229.9957 -110.0330
```

Published with MATLAB® R2022a