Assignment 4

Jennifer Tossell Mathilde Leuridan

March 2019

1 4A Least squares approximations

a)
We want to find values of a,b,c,d,e that satisfy the level set (x,y): f(x,y) =1 for the function f(x,y)= $ax^2 + bxy + cy^2 + dx + ey$ for the x and y observation points given in the question .
Then we want to plot the level set with the a,b,c,d,e values found previously.
To do this we write the m-file:

```
% We are trying to find value of a,b,c,d,e in the level
    set for the function that satisfy
% the observations made.

% These are the ten observations given in the question
x= [ 1.03,0.95,0.87,0.77,0.67,0.56,0.43,0.3,0.16,0.01];
y= [0.39,0.32,0.27,0.22,0.18,0.15,0.13,0.12,0.13,0.15];

% We create a matrix to represent the set of 10 equations
     in 5 unknowns
% (which we get from f(x,y))

A=zeros(10,5);
for i=1:10;
    A(i,1)=x(i)^2;
    A(i,2)=x(i)*y(i);
    A(i,3)=y(i)^2;
```

```
    A(i,4)=x(i);
    A(i,5)=y(i);
end
A


w=ones(10,1); % This represents the value of the level set, because we want f(x,y)=1

% Now we use the lemma given in the question to find the least squares
% minimizer v_star for the values of a,b,c,d,e
B= A';
C=inv(B*A)*B;
v_star=C*w

% We plot the approximate orbit of the planetoid

fimplicit(@(z,u) [z^2, z*u, u^2, z, u]*v_star-1)
hold on

% We superimpose the 10 given observation points on the plot
scatter(x,y,'*m')

xlabel('x axis')
ylabel('y axis')
title('The approximation of the orbit of the planetoid and the observed values')
h = zeros(5, 1);
h(1) = plot(NaN,NaN,'*b');
h(2) = plot(NaN,NaN,'*b');
h(3) = plot(NaN,NaN,'*b');
h(4) = plot(NaN,NaN,'*b');
h(5) = plot(NaN,NaN,'*b');
legend(h, 'a=-2.2427','b=-0.7477','c=-3.3039','d=1.3922','e=7.0554');

hold on
xlim([-1.2 1.6]);
ylim([-0.2 2.2]);
```

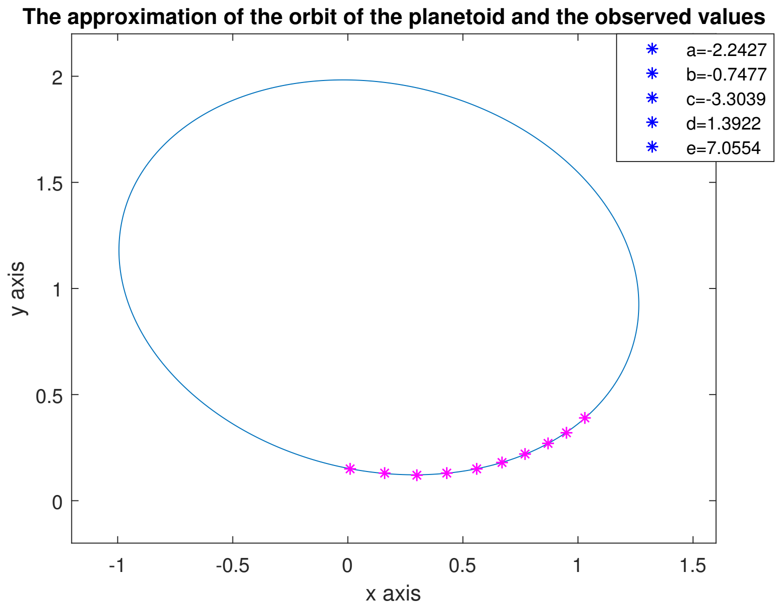
which gives us the matrix A:

```
A =

    1.0609    0.4017    0.1521    1.0300    0.3900
    0.9025    0.3040    0.1024    0.9500    0.3200
    0.7569    0.2349    0.0729    0.8700    0.2700
    0.5929    0.1694    0.0484    0.7700    0.2200
    0.4489    0.1206    0.0324    0.6700    0.1800
    0.3136    0.0840    0.0225    0.5600    0.1500
    0.1849    0.0559    0.0169    0.4300    0.1300
    0.0900    0.0360    0.0144    0.3000    0.1200
    0.0256    0.0208    0.0169    0.1600    0.1300
    0.0001    0.0015    0.0225    0.0100    0.1500
```

and numbers a,b,c,d,e:

```
v_star =

   -2.2427    a
   -0.7477    b
   -3.3039    c
    1.3922    d
    7.0554    e
```

and then using the matlab file, we can plot the ellipse:



where the pink points are the observed points given in the question and the blue curve the approximation of the level curve with the a, b, c, d, e we computed earlier in the file.

b)
To prove the lemma, we first write

$$Av = w + E$$

and try to minimise the error $\|\mathbf{E}\|^2$.
In order to do this we find the partial derivatives of $\|\mathbf{Av}-\mathbf{w}\|^2$ ie $\|\mathbf{E}\|^2$ in the variables v_k for k=1,2,...,m and set them equal to 0.
to get

$$2\sum_{i=1}^{n} A_{ik} \sum_{j=1}^{m} (A_{ij} v_j^* - w_i) = 0, \qquad \forall k = 1, ..., m$$

that is

$$\sum_{i=1}^{n} A_{ik} \sum_{j=1}^{m} (A_{ij} v_j^*) = \sum_{i=1}^{n} A_{ik} w_i, \qquad \forall k = 1, ..., m$$

which we can rewrite in matrix form as:

$$A^T A v^* = A^T w$$

since $\sum_{j=1}^{m} (A_{ij} v_j^*) = Av^*$

thus

$$A^T(Av^* - w) = 0$$

But $Av^* = w + E$ so $E = Av^* - w$ and

$$A^T E = 0$$

Then applying A^T to the identity $Av^* = w + E$, we get

$$A^T A v^* = A^T w + A^T E$$

but since $A^T E = 0$,

$$A^T A v^* = A^T w$$

and thus finally, applying the inverse of $(A^T A)$ on both sides, we get:

$$v^* = (A^T A)^{-1} A^T w$$

which concludes the proof of the lemma.

2 4B Magic squares

a) We show that V_N is a subspace of the vector spaces of all the matrices.
Let A and B $\in V_N$ and let $\lambda \in \mathbb{R}$
Then, if we multiply A by λ, we just multiply all the (a_{ij}) entries of A by λ, thus we multiply the sums along rows, columns and diagonals of the entries by λ. Thus all the sums along diagonals, rows and columns are still the same and the matrix is still in V_N, so is still a magic square.
On the other hand, if we do A+B, we just sum all the entries of the two matrices one by one to get $(A+B)_{ij} = A_{ij} + B_{ij} = a_{ij} + b_{ij}$. Thus if the sum along rows, columns and diagonals of A was r and the sums of B were s, then the new sum along rows, columns and diagonals of A+B is r+s. Thus A+B is a magic square and A+B $\in V_N$. Thus V_N is a subspace, which makes it a vector space too.

Now, as for V_2, it is the vector space with all the square 2 by 2 matrices which have only one coefficient. Since if we have the matrix

$$M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

and the sum of each row, column and diagonal is r, then we have the equations:

$$\begin{cases} a+b=r \\ a+d=r \\ c+b=r \\ c+d=r \end{cases}$$

which in turns implies:

$$a=b=c=d=r/2$$

Thus the matrix M finally becomes:

$$M = \begin{bmatrix} r/2 & r/2 \\ r/2 & r/2 \end{bmatrix}$$

b)
The magic matrix A with row sum r is equivalent to Mv=0 with the given v and M because if you expand Mv = 0 by multiplying the matrices M and v, you get

the system of equations:

$$\begin{cases} -r + a_{11} + a_{12} + a_{13} = 0 \\ -r + a_{21} + a_{22} + a_{23} = 0 \\ -r + a_{31} + a_{32} + a_{33} = 0 \\ -r + a_{11} + a_{21} + a_{31} = 0 \\ -r + a_{12} + a_{32} + a_{32} = 0 \\ -r + a_{13} + a_{23} + a_{33} = 0 \\ -r + a_{11} + a_{22} + a_{33} = 0 \\ -r + a_{13} + a_{22} + a_{31} = 0 \end{cases}$$

which in turn becomes:

$$\begin{cases} a_{11} + a_{12} + a_{13} = r \\ a_{21} + a_{22} + a_{23} = r \\ a_{31} + a_{32} + a_{33} = r \\ a_{11} + a_{21} + a_{31} = r \\ a_{12} + a_{32} + a_{32} = r \\ a_{13} + a_{23} + a_{33} = r \\ a_{11} + a_{22} + a_{33} = r \\ a_{13} + a_{22} + a_{31} = r \end{cases}$$

which is equivalent to the square matrix A being magic.

c)

To find to row reduced form of M, we write in matlab:

```
%We find the row reduced form of the given matrix M.

M= [-1,1,1,1,0,0,0,0,0,0;
    -1,0,0,0,1,1,1,0,0,0;
    -1,0,0,0,0,0,0,1,1,1;
    -1,1,0,0,1,0,0,1,0,0;
    -1,0,1,0,0,1,0,0,1,0;
    -1,0,0,1,0,0,1,0,0,1;
    -1,1,0,0,0,1,0,0,0,1;
    -1,0,0,1,0,1,0,1,0,0]

rref(M)
```

which gives the output:

```
ans =

    1.0000         0         0         0         0         0         0   -1.0000   -1.0000   -1.0000
         0    1.0000         0         0         0         0         0   -0.6667   -0.6667    0.3333
         0         0    1.0000         0         0         0         0   -0.6667    0.3333   -0.6667
         0         0         0    1.0000         0         0         0    0.3333   -0.6667   -0.6667
         0         0         0         0    1.0000         0         0    0.6667   -0.3333   -1.3333
         0         0         0         0         0    1.0000         0   -0.3333   -0.3333   -0.3333
         0         0         0         0         0         0    1.0000   -1.3333   -0.3333    0.6667
         0         0         0         0         0         0         0         0         0         0
```

Thus we see that there are 3 columns without leading ones. Thus the dimension of the kernel of this matrix M is 3. However we are looking for the dimension of the solutions v of Mv=0 , thus we are looking for the dimension of the kernel of M , which is 3. Thus indeed, V_3 is three dimensional.
Now we want to see whether the given matrices M_1, M_2 and M_3 are a basis of V_3 or not.
To show these matrices for a basis, we just need to show they are linearly independent.
We thus write the m-file:

```matlab
%We take the given matrices M1,M2,M3 and put them as line vectors inside a
%matrix B

A1=[1,1,1,1,1,1,1,1,1];
A2 =[0,1,-1,-1,0,1,1,-1,0];
A3 =[1,-1,0,-1,0,1,0,1,-1];
B=zeros(3,9);
B(1,:)=A1
B(2,:)=A2
B(3,:)=A3

%and then we want to see whether these matrices are linearly independent,
% that is we want the combinations for which, if b is a vector with random
% variables b1,b2,b3,b4,b5,b6,b7,b8,b9, Bb=0
```

```
v=zeros(3,1)

%We get the result b of Bb=v
B\v
```

which outputs the vector b:

```
ans =

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

Thus we deduce that the 3 matrices are linearly independent in V_3. However since V_3 is a 3 dimensional space and because the matrices are linearly independent, as established, it means that they span the vector space V_3. And thus they form a basis.

d)

To see whether an arbitrary square matrix of size n is magic or not, we write the m-file:

```
% This file tests if an arbitrary square matrix is magic or not.
% It will return 1 if A is magic and 0 if A is not.
% Suppose A is an nxn matrix

function magicsquares(A)

```

```
r=sum(A(1,:)); % This calculates the number we are suppposed to get when we sum each column,
                % row and main diagonal

M=zeros( 2*length(A)+2,length(A)^2+1);
% This initialises the M matrix to a zero matrix of the right size


M(:,1)=-ones(1,2*length(A)+2);
% This makes every entry in the first column -1.


for i=1:length(A)

    M(i,2+(i-1)*length(A):2+i*length(A)-1)= ones(1,length(A));
    % The first n rows of M : which  sum each row

    for k= 1:length(A);

        M(i+length(A),i+(k-1)*length(A)+1)= 1;
        % The next n rows of M: which sum each column


        M(2*length(A)+1,2+(k-1)*(length(A)+1))=1;
        % Second to last row of M : which sums the main diagonal of A

        M(2*length(A)+2,length(A)+1+(k-1)*(length(A)-1))=1;
        % Last row of M: which sums the antidiagonal of A

    end

end
v=zeros(length(A)^2+1,1);
% We create the vector v as defined in question b)
v(1)=r;

B = reshape(A',[],1);
% This puts all the entries of A in a column vector

v(2:length(v),1)=B;
% Here we test whether Mv=0, in which case A is magic.
if M*v== zeros(1,length(A)^2+1)
```

```
    1 % This represents the true value
else
    0 % This represents the false value


end
end
```

and then to investigate whether the different powers of a magic square A are also magic or not, we write the m-file:

```
% This tests whether the given powers of A are magic or
    not.

function magictester(A)

    for i= 2:5
        magicsquares(A^i);
    end

    magicsquares(inv(A));

end
```

generally, for sizes 3,4,5 of A,
A^2, A^4 and A^{-1} are not magic but A^3 and A^5 are, as we can see by:

Size 3

```
>> magictester(magic(3))

ans =

     0


ans =

     1


ans =

     0


ans =

     1


ans =

     0

>> |
```

Size 4

```
>> magictester(magic(4))

ans =

     0


ans =

     1


ans =

     0


ans =

     1

Warning: Matrix is close to s
> In magictester (line 9)

ans =

     0
```

Size 5

```
>> magictester(magic(5))

ans =

     0


ans =

     1


ans =

     0


ans =

     1


ans =

     0
```

thus we can make the guess for the general behaviour:

The odd powers of a magic square are magic, whereas the even powers of a magic square are not magic anymore.
We also claim that the inverses of a magic square are not magic anymore.