2ND MIDTERM TEST– 2021.05.12.

2D REGRESSION, INTERPOLATION, OPTIMIZATION

1A PROBLEM

We want to determine the size (radius, height) of a minimum surface, unit volume cone.

Surface area of the cone: $A = r^2 \cdot \pi + \pi \cdot r \cdot a$ where a is the slant height: $a = \sqrt{r^2 + h^2}$

Volume of the cone: $V = \dfrac{r^2 \cdot \pi \cdot h}{3}$

a) Write the function of the surface of the cone depending from the radius and height! Write the Matlab function of the unit volume constraint! (3 p)

b) Solve the constrained optimization problem with different methods! In each case determine the radius, height of the cone and check whether the constraint is fulfilled? Determine also the ratio of height to radius and the surface!
   i. Solve with the Matlab built-in function! (4 p)
   ii. Solve using Lagrange-method! (4 p)
   iii. Solve with the penalty function method (K=1000)! (4 p)

SOLUTION - 1A

```
%% 1A - cone surface
clc; clear all; close all;

% A=R^2*pi+pi*R*a, a = sqrt(r^2+h^2)
% V=pi*r^2*h/3=1;
% a) 3 p
A = @(r,h) r.^2*pi+pi*r.*sqrt(r.^2+h.^2)
V = @(r,h) pi*r.^2.*h/3-1
A = @(v) A(v(1),v(2)); V = @(u) V(u(1),u(2));

% b-i) Matlab built-in function (4 p)
nonlcon = @(u) deal([],V(u))
v0 = [0.5, 0.5]
x = fmincon(A,v0,[],[],[],[],[0,0],[],nonlcon)
r = x(1) % 0.69632
h = x(2) % 1.9695
ratio = h/r % 2.8284
S = A([r,h]) % 6.0929
V([r,h]) %  -4.0023e-10

% Lagrange method (4p)
syms r h lam
Lfv = A([r h])+lam*V([r h])
% lam*((h*pi*r^2)/3 - 1) + pi*r^2 + pi*r*(h^2 + r^2)^(1/2)
F = gradient(Lfv,[r,h,lam])
% pi*(h^2 + r^2)^(1/2) + 2*pi*r + (pi*r^2)/(h^2 + r^2)^(1/2) + (2*pi*h*lam*r)/3
%                              (pi*lam*r^2)/3 + (pi*h*r)/(h^2 + r^2)^(1/2)
%                                                   (h*pi*r^2)/3 - 1
sol=solve(F)
%       h: [3×1 sym]
%     lam: [3×1 sym]
%       r: [3×1 sym]
```

```
double([sol.r sol.h sol.lam])
%      0.69632 +         0i       1.9695 +         0i       -4.062 +         0i
%      0.34816 -    0.60303i     -0.98475 +    1.7056i       2.031 +    3.5178i
%      0.34816 +    0.60303i     -0.98475 -    1.7056i       2.031 -    3.5178i
r = double(sol.r(1)) % 0.69632
h = double(sol.h(1)) % 1.9695
ratio = h/r % 2.8284
S = A([r,h]) % 6.0929
V([r,h]) % 0

%b-iii) penalty function method (4 p)
Bfv=@(u) A(u)+1000*V(u).^2
x =fminsearch(Bfv,v0)
% x =    0.69585       1.9681
ratio = x(2)/x(1) % 2.8284
S = A(x) % 6.0847
V(x) % -0.0020296
```

1B PROBLEM

We want to determine the size (radius, height) of a minimum surface, unit volume pot (cylinder open at the top).

Surface of the pot: $A = r^2 \cdot \pi + 2 \cdot r \cdot \pi \cdot h$

Volume of the pot: $V = r^2 \cdot \pi \cdot h$

a) Write the function of the surface of the pot depending from the radius and height! Write the Matlab function of the unit volume constraint! (3 p)
b) Solve the constrained optimization problem with different methods! In each case determine the radius, height of the pot and check whether the constraint is fulfilled? Determine also the ratio of height to radius and the surface!
    i. Solve with the Matlab built-in function! (4 p)
    ii. Solve using Lagrange-method! (4 p)
    iii. Solve with the penalty function method (K=10000)! (4 p)

SOLUTION - 1B

```
%% 1B - pot surface
clc; clear all; close all;

% A=r^2*pi+2*r*pi*h
% V=r^2*pi*h=1;
% a) 3 p
A = @(r,h) r.^2*pi+2*pi*r.*h
V = @(r,h) r.^2*pi.*h-1
A = @(v) A(v(1),v(2)); V = @(u) V(u(1),u(2));

% b-i) Matlab built-in function (4 p)
nonlcon = @(u) deal([],V(u))
v0 = [0.5, 0.5]
x = fmincon(A,v0,[],[],[],[],[0,0],[],nonlcon)
r = x(1) % 0.68278
h = x(2) %  0.68278
ratio = h/r % 1
S = A([r,h]) % 4.3938
V([r,h]) % -2.9621e-13
```

```
% Lagrange method (4p)
syms r h lam
Lfv = A([r h])+lam*V([r h])
% lam*(h*pi*r^2 - 1) + pi*r^2 + 2*pi*h*r
F = gradient(Lfv,[r,h,lam])
% 2*pi*h + 2*pi*r + 2*pi*h*lam*r
%           lam*pi*r^2 + 2*pi*r
%                 h*pi*r^2 - 1
sol=solve(F)
%       h: [3×1 sym]
%     lam: [3×1 sym]
%       r: [3×1 sym]
double([sol.r sol.h sol.lam])
%       0.68278 +          0i      0.68278 +          0i      -2.9292 +          0i
%      -0.34139 +    0.59131i     -0.34139 +    0.59131i       1.4646 +     2.5367i
%      -0.34139 -    0.59131i     -0.34139 -    0.59131i       1.4646 -     2.5367i
r = double(sol.r(1)) % 0.68278
h = double(sol.h(1)) % 0.68278
ratio = h/r % 1
S = A([r,h]) % 4.3938
V([r,h]) %  2.2204e-16

%b-iii) penalty function method (4 p)
Bfv=@(u) A(u)+10000*V(u).^2
x =fminsearch(Bfv,v0)
% x =      0.68274      0.68278
ratio = x(2)/x(1) % 1.0001
S = A(x) % 4.3933
V(x) %   -0.00014667
```

1C PROBLEM

Surface interpolation and constrained optimization problem (15p)

Download points.txt here.

a) Load the X, Y, Z coordinates of the points from points.txt file! The region to be examined: 1000<x<1500, 500<y<1000. Plot the X,Y coordinates of the points and the region in 2D. (2p)

b) Use cubic spline interpolation to display the surface on a contour map within the specified range in the previous figure. The contour distance should be 1 meter. Label contour lines! (5p)

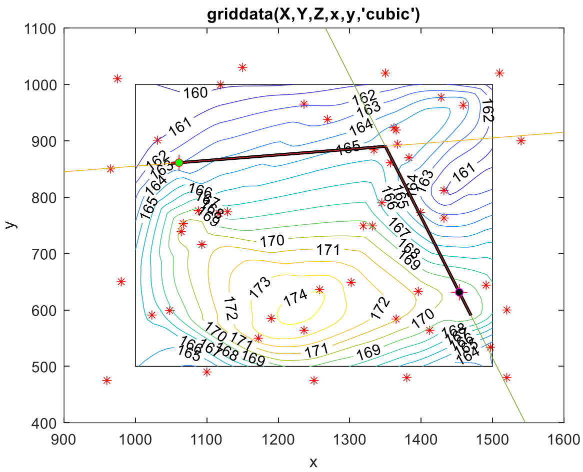
c) Xp=[1050 1350 1470]; Yp=[860 890 590] are the vertices of a power-line. Plot it in the previous figure. (1p)

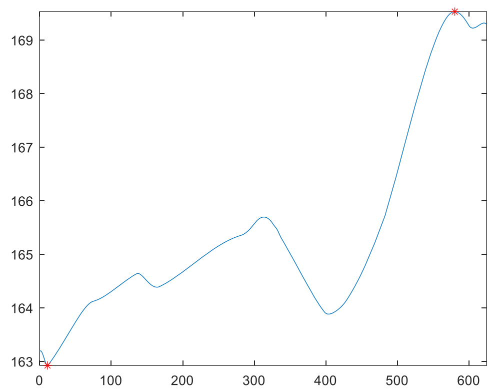
d) Along the power-line, find the lowest (3p) and the highest point of the surface (3p) by any method. Plot them in the previous figure. (1p)
(Hint: You can fit straight lines to the wire sections and then use constrained optimization solution along the lines or fit a parametric linear spline for the entire wire and you can search for extremes along the fitted spline.)

1C – SOLUTION

```
%% 1C feladat - 2D. interpolácó és feltételes szélsőérték keresése (15p)
% a) load data, area - 2p
clc; clear all; close all;
data = load('pontok.txt')
X = data(:,1); Y = data(:,2); Z = data(:,3);
figure(1); plot(X,Y,'r*'); hold on;
rectangle('Position',[1000,500,500,500])
```

```
% b) spline interp., contours - 5 points
f = @(x,y) griddata(X,Y,Z,x,y,'cubic')
F = @(u) f(u(1),u(2))
h = ezcontour(f,[1000,1500,500,1000]);
set(h, 'ShowText','on','LevelList',160:1:180);
axis([900 1600 400 1100])

% c) power line - 1p
Xp=[1050 1350 1470]; Yp=[860 890 590];
plot(Xp,Yp,'k','LineWidth',2)

% d) max, min - powerline - 7 pont
% min/max using line constraints
% minimum in the first line
c1 = polyfit(Xp(1:2),Yp(1:2),1) %  0.1  755
e1 = @(x) 0.1*x+755; fplot(e1)
% y = 0.1*x+755  -> 0.1*x-y=-755
Aeq = [0.1 -1]; beq = -755; lb = [1050,860]; ub = [1350,890];
x0 = [1060,860];
[xmin fmin] = fmincon(F,x0,[],[],Aeq,beq,lb,ub)
% xmin =       1060.9       861.09
% fmin =       162.92
plot(xmin(1),xmin(2),'r+','MarkerSize',10)

% maximum in the second line
c2 = polyfit(Xp(2:3),Yp(2:3),1) %  -2.5        4265
e2 = @(x) -2.5*x+4265; fplot(e2)
% y = -2.5*x + 4265   ->  2.5*x+y = 4265
Aeq = [2.5 1]; beq = 4265; lb = [1350,590]; ub = [1470,890];
Fmax = @(u) -1*F(u); x0 = [1450, 630];
xmax = fmincon(Fmax,x0,[],[],Aeq,beq,lb,ub)
% xmax =       1453.5       631.3
fmax = F(xmax) % fmax  =   169.53
plot(xmax(1),xmax(2),'r+','MarkerSize',10)

% d) Other solution - fitting a parametric linear spline (parameter = distance) – 7 p
dt = [0 sqrt(diff(Xp).^2+diff(Yp).^2)], t = cumsum(dt)
xv = @(u) interp1(t,Xp,u)
yv = @(u) interp1(t,Yp,u)
fplot(xv,yv,[0 max(t)])
zv = @(u) f(xv(u),yv(u)) % heights from the surface
figure(2); fplot(zv,[0 max(t)])
% minimum
tmin = fminsearch(zv,10) % 10.995
xmin = xv(tmin) % 1060.9
ymin = yv(tmin) %  861.09
zmin = zv(tmin) % 162.92
hold on; plot(tmin,zmin,'r*')
figure(1);
plot(xmin,ymin,'ro','MarkerFaceColor','g','MarkerSize',5)
% maximum
zvm = @(u) -1*zv(u);
tmax = fminsearch(zvm,550) % 580.12
xmax = xv(tmax) % 1453.5
ymax = yv(tmax) % 631.3
zmax = zv(tmax) % 169.53
figure(2); hold on; plot(tmax,zmax,'r*')
figure(1); plot(xmax,ymax,'mo','MarkerFaceColor','k','MarkerSize',5)
```





1D PROBLEM

Surface interpolation and constrained optimization problem (15p)

Download points.txt here.

a) Load the X, Y, Z coordinates of the points from points.txt file! The region to be examined: 1000<x<1500, 500<y<1000. Plot the X,Y coordinates of the points and the region in 2D. (2p)

b) Formula of the hypothesis surface:
$f(x,y)=p(1)+p(2)*x+p(3)*y+p(4)*x^2+p(5)*x*y+p(6)*y^2+p(7)*x^3+p(8)*y^3+p(9)*y*x^2+p(10)*x*y^2$
Use regression to determine the coefficients of the hypothesis function! Display the surface on a contour map within the specified range in the previous figure. The contour distance should be 1 meter. Label contour lines! (7p)

c) Xp=[1040 1350], Yp=[860 890] are the vertices of a power-line. Plot it in the previous figure. (1p)

d) Along the power-line, find the lowest (2p) and the highest point of the surface (2p) by any method. Plot them in the previous figure. (1p)
(Hint: You can fit straight line to the wire section and then use constrained optimization solution along the line)

1D – MEGOLDÁS

```
%% D - feladat
clc; clear all; close all; format shortG;
% a) load data, area - 2p
clc; clear all; close all;
data = load('pontok.txt')
X = data(:,1); Y = data(:,2); Z = data(:,3);
figure(1); plot(X,Y,'r*'); hold on;
rectangle('Position',[1000,500,500,500])

% b) regression - 7 p
% f = p(1)+p(2)*x+p(3)*y+p(4)*x^2+p(5)*x*y+p(6)*y^2+p(7)*x^3+p(8)*y^3+p(9)*x^2*y+p(10)*x*y^2
A = [ones(size(X)), X, Y, X.^2, X.*Y, Y.^2, X.^3, Y.^3, X.^2.*Y, X.*Y.^2]; b = Z;
p = A\b
%-130.43, 0.24552, 0.67132, -5.3499e-05, -0.00024265, -0.0006826, -2.5128e-08, 2.6448e-07,
9.5371e-08, 1.5349e-08
f =@(x,y)
p(1)+p(2)*x+p(3)*y+p(4)*x.^2+p(5)*x.*y+p(6)*y.^2+p(7)*x.^3+p(8)*y.^3+p(9)*x.^2.*y+p(10)*x.*y.^2
F = @(u) f(u(1),u(2))
h = ezcontour(f,[1000,1500,500,1000]);
set(h, 'ShowText','on','LevelList',160:1:180);
axis([900 1600 400 1100])

% c) power line - 1p
Xp=[1040 1350], Yp=[860 890]
plot(Xp,Yp,'k','LineWidth',2)

% d) max, min - powerline - 5 pont
% min/max using line constraint
% minimum
c = polyfit(Xp,Yp,1) %  0.096774       759.35
e = @(x) c(1)*x+c(2); fplot(e)
% y = c(1)*x+c(2)  -> c(1)*x-y=-c(2)
Aeq = [c(1) -1]; beq = -c(2); lb = [1040,860]; ub = [1350,890];
x0 = [1050,860];
[xmin fmin] = fmincon(F,x0,[],[],Aeq,beq,lb,ub)
% xmin =        1040          860
```

```
% fmin =       162.99
plot(xmin(1),xmin(2),'mo','MarkerSize',5)

% maximum
Fmax = @(u) -1*F(u); x0 = [1200, 850];
xmax = fmincon(Fmax,x0,[],[],Aeq,beq,lb,ub)
% xmax =       1239.1       879.27
fmax = F(xmax) % fmax   =   165.27
plot(xmax(1),xmax(2),'mo','MarkerSize',5)
```