Nonlinear Optimization Final Exam

Shuowen Wei

1.

(a).

Define the set of functions f in the m-file as follows:

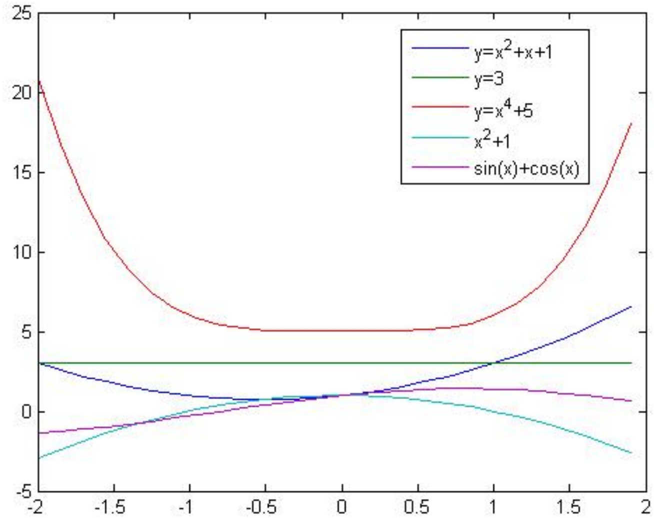
```
function f = Q1_myfun(x)

f(1) = x^2+x+1;
f(2) = 3;
f(3) = x^4 + 5;
f(4) = -x^2+1;
f(5) = sin(x) + cos(x);
```

Then use the *fminimax* in Matlab to solve it as follows:

```
>> x0=1;
[x,fval] = fminimax(@Q1_myfun,x0)

Local minimum possible. Constraints satisfied.

fminimax stopped because the predicted change in the objective
function
is less than the default value of the function tolerance and
constraints
are satisfied to within the default value of the constraint
tolerance.

<stopping criteria details>

x =
    0.0261
fval =
    1.0268    3.0000    5.0000    0.9993    1.0258
```

(b).

After drawing out the graphs of all the functions as follows:



It's easy to find that $f_3(x) = x^4 + 5$ dominates all the other functions, i.e.

$$g(x) = \max\{f_1, f_2, f_3, f_4, f_5\} = f_3(\text{x}) \text{ for all } x$$

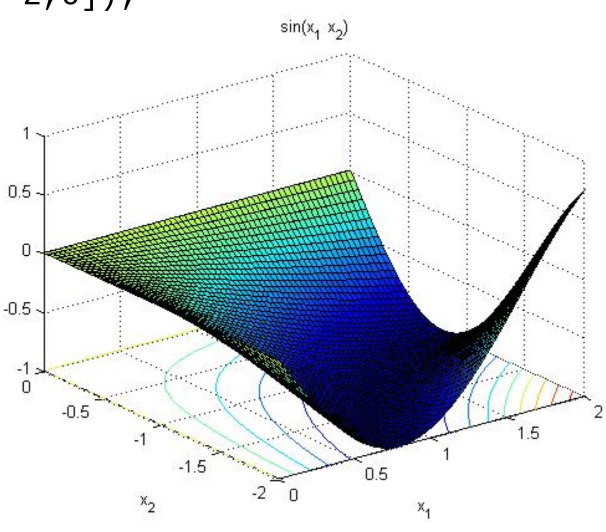
Then, the problem can be reduced to a normal unconstrained optimization problem of single objective function, which is

$$\min: f_3(x) = x^4 + 5.$$

2.

We first of all plot the function within the bounds:

```
f = @(x1,x2) sin(x1.*x2)
ezsurfc(f,[0,2],[-2,0]);
```



Then we define the function f in a m-file as follows:

```
function f = Q2_myfun(x)
f = sin(x(1)*x(2));
```

And solve this problem in Matlab using *fmincon*, the codes are below, here we choose the interior point algorithm:

```
x0 = [1, -1];
lb = [0 -2];
ub = [2 0];
options = optimset('Algorithm','interior-point');
[x, fval] = fmincon(@Q2_myfun,x0,[],[],[],[],lb,ub,[],options)

Local minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the default value of the function tolerance,
and constraints are satisfied to within the default value of the constraint tolerance.

<stopping criteria details>

x =
    1.2533   -1.2533
fval =
   -1.0000
```

Thus the minimum value is -1.

3.

Three types of optimization software:

- Open Source Software
- Proprietary Software
- Freeware

AMPL is an example of Freeware, which is short for A Mathematical Programming Language. its creators, Robert Fourer, David Gay, and Brian Kernighan of Bell Laboratories, offer a version of the software to students for free, but its source code can't be read or modified directly and it also has more restrictions than the professional version.

Opensolver is an example of Open Source Software, it's an open source optimization program and designed to work with Microsoft Excel. It can be downloaded and used for free and its souce code can be read, redistributed, and modified by users.

Maple is an example of Proprietary software, it is owned by company called the Cybernet Group, its source code is kept secret and there are major restrictions on its use.

When user just wants to solve some easy problems and don't want to pay for it, opensource and freeware are advantageous. And when user want to solve some non-linear programming, but OpenSolver cannot handle it, then user had to switch to Maple and AMPL.

4.

Let's consider the following quadratic function:

$$f(x) = \frac{1}{2}x^T Ax - x^T b$$

where A here is a symmetric and positive definite matrix, so $\nabla f = Ax - b$

Define $\phi(\alpha) = f(x + \alpha d)$, where $\alpha > 0$ and $d = -\nabla f = b - Ax$ is the descent direction, thus $\phi(0) = f(x)$ and $\phi'(0) = d^T f'(x) = -d^T d$.

For the first Wolfe Condition:

$$\phi(\alpha) \leq \phi(0) + c_1 \alpha d^T \phi'(0)$$

where $0 < c_1 < c_2 < 1$

$$\phi(\alpha) = f(x + \alpha d) = \frac{1}{2}(x + \alpha d)^T A(x + \alpha d) - (x + \alpha d)^T b$$

$$= \frac{1}{2}(x^T Ax + \alpha(x^T Ad + d^T Ax) + \alpha^2 d^T Ad) - x^T b - \alpha d^T b$$

$$= f(x) + \frac{1}{2}\alpha^2 d^T Ad - \alpha d^T (b - Ax)$$

$$= f(x) + \frac{1}{2}\alpha^2 d^T Ad - \alpha d^T d$$

Thus, by $\phi(\alpha) \leq \phi(0) + c_1 \alpha d^T \phi'(0)$, we get

$$f(x) + \frac{1}{2}\alpha^2 d^T Ad - \alpha d^T d \leq f(x) - c_1 \alpha d^T d$$

$$\alpha \leq 2(1 - c_1)\frac{d^T d}{d^T Ad}$$

For the second Wolfe condition:

$$\phi'(\alpha) \geq c_2 \phi(0)$$

where $0 < c_1 < c_2 < 1$

Thus

$$d^T \nabla f(x + \alpha d) \geq -c_2 d^T d$$

$$d^T (A(x + \alpha d) - b) \geq -c_2 d^T d$$

$$d^T (A\alpha d + Ax - b) \geq -c_2 d^T d$$

$$d^T A\alpha d - d^T d \geq -c_2 d^T d$$

$$\alpha \geq (1 - c_2)\frac{d^T d}{d^T Ad}$$

Thus we have the step size (at the k^{th} step):

$$(1 - c_2)\frac{d_k^T d_k}{d_k^T Ad_k} \leq \alpha_k \leq 2(1 - c_1)\frac{d_k^T d_k}{d_k^T Ad_k}$$

For the second part, the steepest descent method has the optimal line length, which can be solved from:

$$\phi'(\alpha_k) = \frac{d}{d\alpha_k} f(x + \alpha_k d_k) = 0$$

$$d_{\mathrm{k}}^T \nabla f(x + \alpha_k d_k) = 0$$

$$d_{\mathrm{k}}^T d_{\mathrm{k+1}}^T = 0$$

thus the current descent direction and its successive descent direction are perpendicular to each other, which gives a “zig-zag” parttern in iterations.

5.

a).

Let the vector variable $x = (x_1, x_2, x_3, x_4)^T$ respectively denoted the number of hay bales shipped from Asheville to Nashville, Asheville to Dallas, Statesville to Dallas, Statesville to Nashville, thus the object function is :

$$\min c^T x$$

where $c^T = (5, 10, 4, 15)$

The constraint functions are:

$$x_1 + x_2 \leq 700$$

$$x_3 + x_4 \leq 800$$

$$x_1 + x_4 \geq 600$$

$$x_2 + x_3 \geq 400$$

$$x_1, x_2, x_3, x_4 \geq 0$$

b).

Using *linprog.m* in Matlab to solve this question and codes are as follows:

```
clc;clear
f = [5; 10; 4; 15];
A = [1 1 0 0
    0 0 1 1
    -1 0 0 -1
    0 -1 -1 0];
b = [700; 800; -600; -400];
lb=zeros(4,1);
[x, fval, exitflag, output, lambda] = linprog(f,A,b,[],[],lb)


Optimization terminated.
x =
  600.0000
    0.0000
  400.0000
    0.0000
fval =
  4.6000e+003
exitflag =
     1
output =
         iterations: 5
```

```
           algorithm: 'large-scale: interior point'
        cgiterations: 0
             message: 'Optimization terminated.'
     constrviolation: 0
       firstorderopt: 6.2528e-012
lambda =
    ineqlin: [4x1 double]
      eqlin: [0x1 double]
      upper: [4x1 double]
      lower: [4x1 double]
```

So, we just ship 600 hay bales from Asheville to Nashville and 400 hay bales from Statesville to Dallas, and this way makes the cost is the minimum.

6.

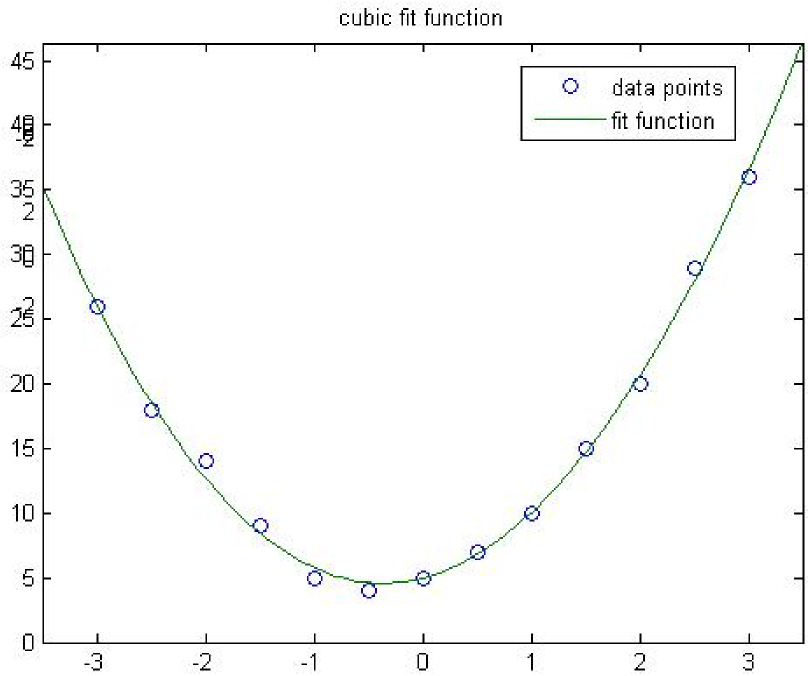
To fit the data points, it's natural to use *lsqcurvefit.m* in Matlab. Because we want to find a fit function to fit these data points such that the approximate values $f(x\_observation)$ has the least-squares error from the observe data points $y\_observation$. The codes and results are as follows:

```
clc;clear
x = [-3 -2.5 -2 -1.5 -1 -0.5 0 0.5 1 1.5 2 2.5 3];
y = [26 18 14 9 5 4 5 7 10 15 20 29 36];
myfun = @(a,x) a(1)*x.^3 + a(2)*x.^2 + a(3)*x + a(4);
a0 = [0, 1, 2, 3];
[a_para res_norm] = lsqcurvefit(myfun, a0, x, y)

Local minimum found.

Optimization completed because the size of the gradient is less
than
the default value of the function tolerance.

<stopping criteria details>

a_para =
   -0.0490    2.9291    2.1741    4.9790
res_norm =
    5.2410
```

The squared 2-norm of the residual is 5.241, which is a very good result, now we want to plot the data points and the fit function:

```
xx = -3.5:0.05:3.5;
plot(x, y, 'o', xx, myfun(a_para, xx), '-')
```

```
legend('data points','fit function');
title('cubic fit function');
```



7.

(a).

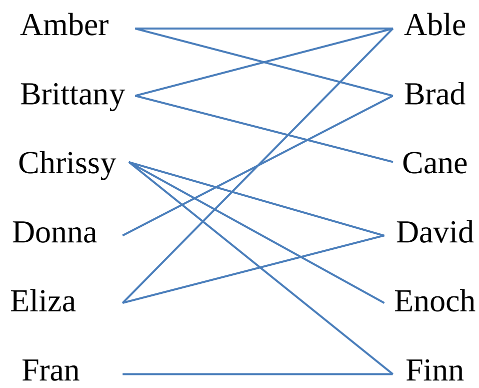
The presence of multiple minimum or maximum will distract those algorithms used to solve local optimization problems, those methods do not suffice for solving global optimization problems, it will get trapped into the minimum or maximum area.

(b).

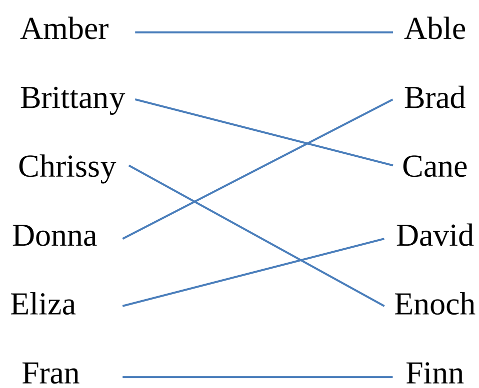
Because Genetic algorithms abstract from the biological process and also introduce a change in semantics by being goal-driven: that those candidates fitting the goal best will have a better chance to survive and reproduce, while the others will be wiped out. Thus it will not just stop at local optimizers and it had a larger chance to find the global ones.

8.

The bipartite graph is:



Yes, we can match everybody as follows:



9.

```
clc;clear
x = [0 1 2 3 4];
q = [0.1 0.25 0.45 0.15 0.05];

p1 =[0 0.3 0.6 0.1 0];
p2 =[0 1 0 0 0];
p3 =[0.2 0.5 0.15 0.09 0.06];
p4 =[0.2 0.2 0.2 0.2 0.2];

KL1 = kldiv(x,q,p1)
KL2 = kldiv(x,q,p2)
KL3 = kldiv(x,q,p3)
KL4 = kldiv(x,q,p4)

KL1 =
```

```
   Inf
KL2 =
   Inf
KL3 =
    0.4606
KL4 =
    0.3447
```

It is clear that strategy 4 has the smallest variance in the car sales from the top dealerships, which means this is the best choice for the CEO.

Dr. Plemmons,

Thanks for everything you’ve done for us, we really appreciate your life long service for education, we leant a lot from you, and it’s our great honor to be your last class. And I personally really appreciate your recommendation letter for my application to CS Program.

Sincerely yours,

Shuowen