Optimization Practical Exercise
Sommersemester 2023

Donnermair Maximilian k12005908@students.jku.at
Fromherz Jakob k12011689@students.jku.at
Haslhofer Eva-Maria k12007773@students.jku.at
Scharnreitner Franz k12011695@students.jku.at
Weiss Hannah Maria k12021111@students.jku.at

12 May 2023

1 Exercise 1

1.1 Linesearch Algorithm with Wolfe-Powell Condition

```
function [x_new,f_new,g_new, exit_flag, alpha, eval] = LineSearch (f, x_old, f_old, g_old, p
    , phi_min, alpha_st)
    %Linesearch algorithm with Wolfe-Powell Condition (Algorithm 4.2)

    %Setting of parameters as described in the scriptum
    tau =  0.1;
    tau1 = 0.1;
    tau2 = 0.6;
    xi1 = 1;
    xi2 = 10;
    mu1 = 0.01;
    mu2 = 0.9;
    sigma = 0.9; %sigma greater than mu1

    %if only 6 arguments are given
    if nargin == 6
      alpha_st = 1;
    end

    alpha_l = 0;
    phi_l = f_old;  %phi_l=phi(0)
    x_old=x_old(:); % x_old always considered as column vector
    dphi_l = dot(g_old,p);
    exit_flag = 0;
    flag = true;  % means that alpha_r is infinity
    alpha_r = 10^30; %alpha_r should be very large

    alpha_tilde = 0; %initialization of alpha_tilde
    eval = 0; %number of evaluations needed

    while abs(alpha_r - alpha_l) > 10^(-15)
        %evaluation of function, gradient and exit_flag
        x_temp = x_old + alpha_st * p;
        [f_temp, g_temp, exit_flag] = f(x_temp);
        eval = eval + 1;

        %function could not be evaluated (alpha_hat not in omega_prime)
        if exit_flag ~= 0
            alpha_r = alpha_st;
            alpha_st = alpha_l + tau1 * (alpha_r - alpha_l);

        %function could be evaluated (alpha_hat in omega_prime)
        else
            phi_hat = f_temp;

            %if function smaller that phi_min, function is declared as
            %unbounded
            if phi_hat < phi_min
                exit_flag = 2; % output is not minimum
                fprintf("Error, unbounded function")
                alpha = alpha_st;
                x_new = x_temp;
                f_new = f_temp;
                return
            end
            %setting alpha_r and calculating alpha_st
            if phi_hat > (f_old + mu1 * alpha_st * dphi_l)
                flag = false; %alpha_r is not infinity
                alpha_r = alpha_st;
                length = alpha_r - alpha_l;
                c = (phi_hat - phi_l  - dphi_l*length)/ (length^2);
                alpha_tilde = alpha_l - dphi_l/(2*c);
```

```
                alpha_st = min(max(alpha_l + tau * length ,alpha_tilde),alpha_r - tau *
    length);

            else
                dphi_hat = dot(g_temp,p); %derivative of phi_hat
                %calculating alpha_tilde
                if dphi_hat < sigma * dphi_l
                    if flag
                        if dphi_l/dphi_hat > (1 + xi2)/xi2
                            alpha_tilde = alpha_st + (alpha_st - alpha_l) * max(dphi_hat/(
    dphi_l - dphi_hat),xi1);
                        else
                            alpha_tilde = alpha_st + xi2 * (alpha_st - alpha_l);
                        end
                    else
                        if dphi_l/dphi_hat>1+(alpha_st-alpha_l)/(tau2*(alpha_r-alpha_st))
                            alpha_tilde = alpha_st+max((alpha_st-alpha_l)*dphi_hat/(dphi_l-
    dphi_hat),tau1*(alpha_r-alpha_st));
                        else
                            alpha_tilde=alpha_st+tau2*(alpha_r-alpha_st);
                        end
                    end
                % setting alpha_l, phi_l, derivative of phi_l and alpha_st
                alpha_l = alpha_st;
                phi_l=phi_hat;
                dphi_l = dphi_hat;
                alpha_st=alpha_tilde;

                %returning from algorithm
                else
                    alpha = alpha_st;
                    x_new = x_old+alpha*p;
                    [f_new,g_new,exit_flag] = f(x_new);
                    eval = eval+1;
                    return
                end
            end
        end
    end
    % final return (if there was no return before)
    alpha = alpha_st;
    x_new = x_old+alpha*p;
    [f_new,g_new,exit_flag] = f(x_new);
    eval = eval+1;
    end
```

src/LineSearch.m

1.2 Method of steepest descent

```
function [x,f_val,g,exit_flag, iter, evals] = SteepestDescent (f, x0, phi_min,eps,itmax,
    typ_f,typ_x)
  %Steepest Descent Algorithm for testing LineSearch

  %for varible input arguments
  if nargin < 7
    typ_x(1:length(x0)) = 10^-4;
  end

  if nargin < 6
    typ_f = 10^-4;
  end

  if nargin < 5
    itmax = 1000;
  end
```

```
16
17    if nargin < 4
18      eps = 10^-6;
19    end
20
21    if nargin < 3
22      phi_min = -10^30;
23    end
24
25    %starting value for iteration, function value, gradient, exit_flag
26    xk = x0;
27    [fk,gk,exit_flag] = f(x0);
28    evals = 1;
29
30    %iterations of steepest descent
31    for iter = 1:itmax
32
33      %termination condition fullfilled (relative gradient less than tolerance)
34      if max(abs(gk) .* typ_x / typ_f) <= eps
35        x = xk;
36        f_val = fk;
37        g = gk;
38        return
39      end
40
41      %calling LineSearch
42      [xk,fk,gk,exit_flag,~,eval_temp] = LineSearch(f,xk,fk,gk,-gk,phi_min);
43      typ_f = max(typ_f,abs(fk));
44      typ_x = max(typ_x,abs(xk));
45      evals = evals + eval_temp; %updating number of evaluations
46    end
47
48    %Setting output, if termination condition is not fullfilled and maximal
49    %number of iterations reached
50    x = xk;
51    f_val = fk;
52    g = gk;
53    exit_flag = 1;
54
55  end
```

src/SteepestDescent.m

1.3 Testfunctions a) to d)

```
1  function [f_val, g, exit_flag] = f_a(x)
2      exit_flag=0;
3      x1=x(1);
4      x2=x(2);
5      f_val = 0.5*(x1+x2)^2 + 0.05*(x1-x2)^2;
6      g = [1.1*x1 + 0.9*x2; 0.9*x1 + 1.1*x2];
7  end
```

src/f_a.m

```
1  function [f_val, g, exit_flag] = f_b(x)
2      exit_flag=0;
3      x1=x(1);
4      x2=x(2);
5      f_val = 100*(x2-x1^2)^2 + (1-x1)^2;
6      g = [400*x1*(x1^2-x2) + 2*(x1-1); 200*(x2-x1^2)];
7  end
```

src/f_b.m

```
function [f_val, g, exit_flag] = f_c(x)
    exit_flag=0;
    n = length(x);
    if min(x)<=0
        f_val = 10^30; %dummy values
        g = ones(n,1); %dummy values
        exit_flag = 1;
        return
    end

    x=x(:); %transform to column vector
    %(1:n)', x, log(x) column vectors
    f_val = sum((1:n)'.*x.*log(x)) + 1/sum(x);
    g = zeros(n,1); %column vector
    for i=1:n
        g(i)=i*(log(x(i))+1) - 1/sum(x)^2;
    end
end
```

src/f_c.m

```
function [f_val, g, exit_flag] = f_d(x)
    exit_flag=0;
    n = length(x);

    x=x(:);
    v=(1:n)';
    A=(1/(100*n))*eye(n)+hilb(n);
    xAx = x'*A*x;
    f_val = (xAx)^2-(x'*A*v)^2;
    g = 4*xAx*(A*x)-2*(x'*A*v)*A*v;
end
```

src/f_d.m

1.4 Testscript

```
diary test1.txt
diary on
disp("Funktion f_a test")

[x_val,f_val,g_val,~,iter,evals] = SteepestDescent(@f_a,[10,-3]);
displayVals(x_val,f_val,g_val,iter,evals);

disp("-----------------------------------------")

disp("Funktion f_b test")

[x_val,f_val,g_val,~,iter,evals]= SteepestDescent(@f_b,[-1,2],-10^30,10^-6,5000);
    displayVals(x_val,f_val,g_val,iter,evals);

disp("-----------------------------------------")

disp("Funktion f_c test");
for j = [10 100 1000]
    fprintf("For n = %d \n",j);
    x0 = 5 * ones(1,j);
    [x_val,f_val,g_val,~,iter,evals] = SteepestDescent(@f_c,x0,-10^30,10^-9,10000);
    displayVals(x_val,f_val,g_val,iter,evals);

end


disp("-----------------------------------------")

```

```
disp("Funktion f_d test");

for j = [10 100 1000]
    fprintf("For n = %d \n",j);
    x0 = zeros(j,1);
    x0(j) = 10*j;
    [x_val,f_val,g_val,~,iter,evals] = SteepestDescent(@f_d,x0,-10^30,10^-6,10000);
    displayVals(x_val,f_val,g_val,iter,evals);

end

disp("-----------------------------------------------")


diary off
```

src/testex1.m

With a quick printing function.

```
function retval = displayVals (x_val, f_val,g_val,iter,evals)
    if length(x_val) < 11
        fprintf("Minimum computed at: ");
        x_val


 end

    disp("Minimum at function f value: ");
    f_val

    disp("Norm of gradient: ");
    no = norm(g_val);
    no

    fprintf("with %d iterations and %d function evals \n",iter,evals);


end
```

src/displayVals.m

We get the output

```
Funktion f_a test
Minimum computed at:
x_val =

   1.0e-05 *

    0.1724
   -0.1690

Minimum at function f value:

f_val =

   5.8328e-13

norm of gradient:

no =

   4.8506e-07

with 71 iterations and 145 function evals
-----------------------------------------------
Funktion f_b test
Minimum computed at:
```

```
x_val =

    1.0000
    1.0000

Minimum at function f value:

f_val =

   4.2321e-11

norm of gradient:

no =

   6.1267e-06

with 4614 iterations and 20811 function evals
-------------------------------------
Funktion f_c test
For n = 10
Minimum computed at:
x_val =

    0.3949
    0.3811
    0.3767
    0.3745
    0.3731
    0.3722
    0.3716
    0.3712
    0.3708
    0.3705

Minimum at function f value:

f_val =

  -19.9644

norm of gradient:

no =

   7.6904e-08

with 89 iterations and 345 function evals
For n = 100
Minimum at function f value:

f_val =

  -1.8578e+03

norm of gradient:

no =

   2.2556e-06

with 461 iterations and 2055 function evals
For n = 1000
Minimum at function f value:

f_val =

  -1.8412e+05
```

```
norm of gradient:

no =

   4.8703e-04

with 3413 iterations and 18742 function evals
-------------------------------------
Funktion f_d test
For n = 10
Minimum computed at:
x_val =

    0.7074
    1.4125
    2.1185
    2.8354
    3.5448
    4.2462
    4.9424
    5.6360
    6.3286
    7.1201

Minimum at function f value:

f_val =

  -1.6770e+04

norm of gradient:

no =

    0.0436

with 10000 iterations and 50003 function evals
For n = 100
Minimum at function f value:

f_val =

  -1.0983e+10

norm of gradient:

no =

   3.6507e+04

with 10000 iterations and 80002 function evals
-------------------------------------
Funktion f_d test
For n = 1000
Minimum at function f value:

f_val =

  -1.0513e+16

norm of gradient:

no =

   7.1347e+07

with 10000 iterations and 110001 function evals
```

```
162 ----------------------------------------
```

src/test1.txt

1.5 Interpretation

We can see, that the SteepestDescent works for f_a relatively well, while to get a relatively good result for f_b we already need > 4000 iterations. f_c works quite well, although the number of iterations needed grows fast with increase in dimension size. The algorithm fails to work for f_d however for big dimensions.
We want to analyze this behavior with Theorem 4.5 of our script. Therefore, we calculate with the help of the functions f_aH to f_dH from Exercise 2 the condition numbers in our calculated minimum. If we are not too far away from the minimum we get a good approximation of the hessian (because the hessian is continuous, furthermore we can see, that the condition number oh the hessians at the calculated minima in exercise two are roughly the same, especially in order of magnitude) and therefore the condition number is near the calculated value. Then we calculate the upper bound of the convergence rate $Q1(f^k) < 1 - \eta * \frac{4*\kappa}{(1+\kappa)^2}$.For the chosen values $\mu_1 = 0.01$ and $\sigma = 0.9$, we get $\eta = 0.0396$.
We get the output:

```
1  >> testex1_cond
2
3  Funktion f_a test
4  Condition number of H in calculated minimum  10.0000
5  Upper bound for convergence rate 0.9869
6  ----------------------------------------
7  Funktion f_b test
8  Condition number of H in calculated minimum  2507.9466
9  Upper bound for convergence rate 0.9999
10 ----------------------------------------
11 Funktion f_c test
12 For n = 10
13 Condition number of H in calculated minimum  10.5232
14 Upper bound for convergence rate 0.9874
15
16 For n = 100
17 Condition number of H in calculated minimum  100.0715
18 Upper bound for convergence rate 0.9984
19
20 For n = 1000
21 Condition number of H in calculated minimum  1000.3861
22 Upper bound for convergence rate 0.9998
23
24 ----------------------------------------
25 Funktion f_d test
26 For n = 10
27 Condition number of H in calculated minimum  1256.3686
28 Upper bound for convergence rate 0.9999
29
30 For n = 100
31 Condition number of H in calculated minimum  15456.3922
32 Upper bound for convergence rate 1.0000
33
34 For n = 1000
35 Condition number of H in calculated minimum  203338.7870
36 Upper bound for convergence rate 1.0000
```

src/test1_cond.txt

The condition number of the hessian of our calculated minimum of f_a is 10 and the theorem tells us that the convergence rate is smaller than 0.9869. It converges very fast (our theorem only gives us an upper bound). The condition number of the hessian of f_b in the calculated minimum is 2508, the upper bound for the convergence rate is 0.9999. The algorithm converges quite slowly. The condition number of the hessian of f_c for n=10 in the calculated minimum is 10.5233, the upper bound 0.9874. This is better than for f_b. When we increase n the condition matrix and the upper bound increases. The condition number of the hessian of

f_d for n=10 in the calculated minimum is 1256, the upper bound 0.9999, quite near to 1. As we increase the dimension the upper bound in rounded to 1.

2 Exercise 2

2.1 Testfunctions a) to d) with Hessian

```
function [f_val, g, H] = f_aH(x)
    x1=x(1);
    x2=x(2);
    f_val = 0.5*(x1+x2)^2 + 0.05*(x1-x2)^2;
    g = [1.1*x1+0.9*x2;0.9*x1+1.1*x2];
    if nargout > 2
        H = [1.1,0.9;0.9,1.1];
    end
end
```

src/f_aH.m

```
function [f_val, g, H] = f_bH(x)
    x1=x(1);
    x2=x(2);
    f_val = 100*(x2-x1^2)^2 + (1-x1)^2;
    g = [400*x1*(x1^2-x2) + 2*(x1-1); 200*(x2-x1^2)];
    if nargout > 2
        H = [1200*x1^2 - 400*x2 + 2, -400*x1; -400*x1, 200];
    end
end
```

src/f_bH.m

```
function [f_val, g, H] = f_cH(x)
    n = length(x);
    if min(x)<=0
        f_val = 10^30; %dummy values
        g = ones(n,1); %dummy values
        H = zeros(n);
        return
    end

    x=x(:); %transform to column vector
    %(1:n)', x, log(x) column vectors
    f_val = sum((1:n)'.*x.*log(x)) + 1/sum(x);
    g = zeros(n,1); %column vector
    for i=1:n
        g(i) = i*(reallog(x(i))+1) - 1/sum(x)^2;
    end
    if nargout > 2
        h = 2/sum(x)^3;
        H = h*ones(n,n) + diag((1:n)'./x,0);
    end
end
```

src/f_cH.m

```
function [f_val, g, H] = f_dH(x)
    n = length(x);

    x=x(:);
    v=(1:n)';
    A=(1/(100*n))*eye(n)+hilb(n);
    xAx = x'*A*x;
    f_val = (xAx)^2-(x'*A*v)^2;

```

```
    if nargout > 1
        g = 4*xAx*(A*x)-2*(x'*A*v)*A*v;
        if nargout >2
            H = 4*(2*A*x+(A*x)'+x'*A*x*A)-2*A*v*(A*v)';
        end
    end

end
```

src/f_dH.m

2.2 Tests in Exercise 2

```
dfile1 = 'Test1.txt';
if exist(dfile1, 'file') ; delete(dfile1); end
diary(dfile1)
diary on

options1= optimset('LargeScale','off','GradObj','on','Display','off');
options2= optimset('LargeScale','on','GradObj','on','Hessian','off','Display','off');
options3= optimset('Algorithm','trust-region','LargeScale','on','GradObj','on','Hessian','on
    ','Display','off');
options = [options1,options2,options3];

%(a)
disp("=================================================================")
disp("(A)")
disp("=================================================================")
for i=1:length(options)
    disp("Testing objective function f_a with options: ")
    disp(options(i));
    if i==3
        [x,f,~,out,g,H] = fminunc(@f_aH,[10,-3],options(i));
    else
        [x,f,~,out,g] = fminunc(@f_aH,[10,-3],options(i));
    end
    disp("Calculated minimizer: ");
    x
    disp("Calculated Minimum value: " + f);
    disp("Norm of gradient at calculated Minimum: " + norm(g));
    if i==3
        disp("Condition of Hessian at calculated Minimum: " + cond(H))
    end
    disp(out);
    disp("=================================================================")
end


disp("=================================================================")
disp("(B)")
disp("=================================================================")
for i=1:length(options)
    disp("Testing objective function f_b with options: ")
    disp(options(i));

    if i==3
        [x,f,~,out,g,H] = fminunc(@f_bH,[-1,2],options(i));
    else
        [x,f,~,out,g] = fminunc(@f_bH,[-1,2],options(i));
    end
    disp("Calculated minimizer: ");
    x
    disp("Calculated Minimum value: " + f);
    disp("Norm of gradient at calculated Minimum: " + norm(g));
    if i==3
        disp("Condition of Hessian at calculated Minimum: " + cond(H))
```

```
    end
    disp(out)
    disp("==========================================================")

end


disp("==========================================================")
disp("C")
disp("==========================================================")

for i=1:length(options)
    disp("Testing objective function f_c with options: ")
    disp(options(i));

    disp("==========================================================")
    for j=1:3
        n=10^j;
        disp("Results for n= "+n);
        if i==3
            [x,f,~,out,g,H] = fminunc(@f_cH,5*ones(1,n),options(i));
        else
            [x,f,~,out,g] = fminunc(@f_cH,5*ones(1,n),options(i));
        end
        if j==1
            disp("Calculated minimizer: ");
            x
        end
        disp("Calculated Minimum value: " + f);
        disp("Norm of gradient at calculated Minimum: " + norm(g));
        if i==3
            disp("Condition of Hessian at calculated Minimum: " + cond(H))
        end
        disp(out)
        disp("==========================================================")
    end

end

disp("==========================================================")
disp("(D)")
disp("==========================================================")
for i=1:length(options)
    disp("Testing objective function f_d with options: ")
    disp(options(i));

    disp("==========================================================")
    for j=1:3
        n=10^j;
        disp("Results for n= "+n);
        if i==3
            [x,f,~,out,g,H] = fminunc(@f_dH,[zeros(1,n-1),10*n],options(i));
        else
            [x,f,~,out,g] = fminunc(@f_dH,[zeros(1,n-1),10*n],options(i));
        end
        if j==1
            disp("Calculated minimizer: ");
            x
        end
        disp("Calculated Minimum value: " + f);
        disp("Norm of gradient at calculated Minimum: " + norm(g));
        if i==3
            disp("Condition of Hessian at calculated Minimum: " + cond(H))
        end
        disp(out)
        disp("==========================================================")
    end

```

```
end

diary off
```

src/Prog1Ex2.m

2.3 Test output

As can be observed, for every testfunction and every given starting value `fminunc` was called with the three required options. Running `Prog1Ex2.m` produces the output below (console output logged using `Matlab`-diary functionality)

```
================================================
(A)
================================================
Testing objective function f_a with options:
                    Display: 'off'
                MaxFunEvals: []
                    MaxIter: []
                     TolFun: []
                       TolX: []
                FunValCheck: []
                  OutputFcn: []
                   PlotFcns: []
              ActiveConstrTol: []
                  Algorithm: []
    AlwaysHonorConstraints: []
             DerivativeCheck: []
                 Diagnostics: []
               DiffMaxChange: []
               DiffMinChange: []
              FinDiffRelStep: []
                FinDiffType: []
          GoalsExactAchieve: []
                GradConstr: []
                   GradObj: 'on'
                    HessFcn: []
                    Hessian: []
                   HessMult: []
                HessPattern: []
                 HessUpdate: []
             InitBarrierParam: []
     InitTrustRegionRadius: []
                   Jacobian: []
                  JacobMult: []
                JacobPattern: []
                  LargeScale: 'off'
                   MaxNodes: []
                  MaxPCGIter: []
                MaxProjCGIter: []
                  MaxSQPIter: []
                     MaxTime: []
                MeritFunction: []
                   MinAbsMax: []
          NoStopIfFlatInfeas: []
              ObjectiveLimit: []
        PhaseOneTotalScaling: []
             Preconditioner: []
           PrecondBandWidth: []
             RelLineSrchBnd: []
    RelLineSrchBndDuration: []
               ScaleProblem: []
       SubproblemAlgorithm: []
                    TolCon: []
                 TolConSQP: []
                 TolGradCon: []
```

```
                    TolPCG: []
                  TolProjCG: []
               TolProjCGAbs: []
                   TypicalX: []
                UseParallel: []

Calculated minimizer:

x =

   1.0e-06 *

   -0.2002    0.6078

Calculated Minimum value: 1.1571e-13
Norm of gradient at calculated Minimum: 5.8763e-07
       iterations: 8
        funcCount: 9
         stepsize: 1.8956e-04
     lssteplength: 1
    firstorderopt: 4.8838e-07
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
    gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    5.251446e-08, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                          Options relative
    norm(gradient) =  5.25e-08             OptimalityTolerance =  1e-06 (default)'

==========================================================
Testing objective function f_a with options:
                    Display: 'off'
                MaxFunEvals: []
                    MaxIter: []
                     TolFun: []
                       TolX: []
                FunValCheck: []
                  OutputFcn: []
                   PlotFcns: []
             ActiveConstrTol: []
                  Algorithm: []
    AlwaysHonorConstraints: []
            DerivativeCheck: []
                Diagnostics: []
              DiffMaxChange: []
              DiffMinChange: []
             FinDiffRelStep: []
                FinDiffType: []
          GoalsExactAchieve: []
                GradConstr: []
                    GradObj: 'on'
                    HessFcn: []
                    Hessian: 'off'
                   HessMult: []
                HessPattern: []
                 HessUpdate: []
           InitBarrierParam: []
     InitTrustRegionRadius: []
                   Jacobian: []
                  JacobMult: []
                JacobPattern: []
                  LargeScale: 'on'
                   MaxNodes: []
                  MaxPCGIter: []
               MaxProjCGIter: []
                  MaxSQPIter: []
                     MaxTime: []
               MeritFunction: []
```

```
                MinAbsMax: []
         NoStopIfFlatInfeas: []
             ObjectiveLimit: []
       PhaseOneTotalScaling: []
             Preconditioner: []
          PrecondBandWidth: []
            RelLineSrchBnd: []
    RelLineSrchBndDuration: []
              ScaleProblem: []
       SubproblemAlgorithm: []
                   TolCon: []
                TolConSQP: []
               TolGradCon: []
                   TolPCG: []
                TolProjCG: []
              TolProjCGAbs: []
                 TypicalX: []
              UseParallel: []

Calculated minimizer:

x =

   1.0e-06 *

   -0.2002    0.6078

Calculated Minimum value: 1.1571e-13
Norm of gradient at calculated Minimum: 5.8763e-07
       iterations: 8
        funcCount: 9
         stepsize: 1.8956e-04
     lssteplength: 1
    firstorderopt: 4.8838e-07
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
    gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    5.251446e-08, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                    Options relative
    norm(gradient) =  5.25e-08             OptimalityTolerance =  1e-06 (default)'

==========================================================
Testing objective function f_a with options:
                   Display: 'off'
              MaxFunEvals: []
                  MaxIter: []
                   TolFun: []
                     TolX: []
              FunValCheck: []
                OutputFcn: []
                 PlotFcns: []
           ActiveConstrTol: []
                Algorithm: 'trust-region'
    AlwaysHonorConstraints: []
          DerivativeCheck: []
              Diagnostics: []
            DiffMaxChange: []
            DiffMinChange: []
          FinDiffRelStep: []
              FinDiffType: []
        GoalsExactAchieve: []
               GradConstr: []
                  GradObj: 'on'
                  HessFcn: []
                  Hessian: 'on'
                 HessMult: []
              HessPattern: []
```

```
                 HessUpdate: []
            InitBarrierParam: []
      InitTrustRegionRadius: []
                  Jacobian: []
                 JacobMult: []
               JacobPattern: []
                 LargeScale: 'on'
                  MaxNodes: []
                 MaxPCGIter: []
               MaxProjCGIter: []
                  MaxSQPIter: []
                    MaxTime: []
               MeritFunction: []
                  MinAbsMax: []
          NoStopIfFlatInfeas: []
              ObjectiveLimit: []
       PhaseOneTotalScaling: []
              Preconditioner: []
           PrecondBandWidth: []
              RelLineSrchBnd: []
    RelLineSrchBndDuration: []
               ScaleProblem: []
       SubproblemAlgorithm: []
                    TolCon: []
                 TolConSQP: []
                TolGradCon: []
                    TolPCG: []
                 TolProjCG: []
               TolProjCGAbs: []
                  TypicalX: []
                UseParallel: []

Calculated minimizer:

x =

   1.0e-03 *

    0.7869   -0.2361

Calculated Minimum value: 2.0406e-07
Norm of gradient at calculated Minimum: 0.00079236
Condition of Hessian at calculated Minimum: 10
         iterations: 12
          funcCount: 13
           stepsize: 0.0025
       cgiterations: 12
      firstorderopt: 6.5317e-04
          algorithm: 'trust-region'
            message: 'Local minimum possible.  fminunc  stopped because the final change
    in function value relative to  its  initial value is less than the default value of the
     function tolerance.  Stopping  criteria details:  Optimization  stopped because
    the relative objective function value is changingby less than options.
    FunctionTolerance = 1.000000e-06.  Optimization  Metric
            Options relative change objective =  7.82e-07  FunctionTolerance =
     1e-06 (default)'
    constrviolation: []

================================================================
================================================================
(B)
================================================================
Testing objective function f_b with options:
                  Display: 'off'
             MaxFunEvals: []
                 MaxIter: []
                  TolFun: []
                    TolX: []
```

```
                FunValCheck: []
                  OutputFcn: []
                   PlotFcns: []
              ActiveConstrTol: []
                  Algorithm: []
    AlwaysHonorConstraints: []
             DerivativeCheck: []
                 Diagnostics: []
               DiffMaxChange: []
               DiffMinChange: []
             FinDiffRelStep: []
                FinDiffType: []
          GoalsExactAchieve: []
                GradConstr: []
                   GradObj: 'on'
                   HessFcn: []
                   Hessian: []
                  HessMult: []
                HessPattern: []
                HessUpdate: []
           InitBarrierParam: []
     InitTrustRegionRadius: []
                  Jacobian: []
                 JacobMult: []
               JacobPattern: []
                 LargeScale: 'off'
                   MaxNodes: []
                 MaxPCGIter: []
               MaxProjCGIter: []
                 MaxSQPIter: []
                   MaxTime: []
               MeritFunction: []
                  MinAbsMax: []
         NoStopIfFlatInfeas: []
             ObjectiveLimit: []
       PhaseOneTotalScaling: []
              Preconditioner: []
           PrecondBandWidth: []
             RelLineSrchBnd: []
    RelLineSrchBndDuration: []
               ScaleProblem: []
       SubproblemAlgorithm: []
                    TolCon: []
                 TolConSQP: []
                TolGradCon: []
                    TolPCG: []
                  TolProjCG: []
               TolProjCGAbs: []
                  TypicalX: []
                UseParallel: []

Calculated minimizer:

x =

    1.0000    1.0000

Calculated Minimum value: 6.6115e-11
Norm of gradient at calculated Minimum: 0.00029178
       iterations: 39
        funcCount: 50
         stepsize: 4.0089e-04
     lssteplength: 1
    firstorderopt: 2.6299e-04
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization  completed because the size of the
     gradient is less  t h a n the  default  value of the optimality tolerance.  Stopping
    criteria details:  Optimization  completed: The first-order optimality measure,
```

```
    6.624363e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                        Options relative
    norm(gradient) =   6.62e-07              OptimalityTolerance =   1e-06 (default)'

================================================================
Testing objective function f_b with options:
                    Display: 'off'
                MaxFunEvals: []
                    MaxIter: []
                     TolFun: []
                       TolX: []
                FunValCheck: []
                  OutputFcn: []
                   PlotFcns: []
              ActiveConstrTol: []
                  Algorithm: []
    AlwaysHonorConstraints: []
             DerivativeCheck: []
                 Diagnostics: []
               DiffMaxChange: []
               DiffMinChange: []
             FinDiffRelStep: []
                FinDiffType: []
          GoalsExactAchieve: []
                GradConstr: []
                   GradObj: 'on'
                   HessFcn: []
                   Hessian: 'off'
                  HessMult: []
                HessPattern: []
                HessUpdate: []
           InitBarrierParam: []
      InitTrustRegionRadius: []
                  Jacobian: []
                 JacobMult: []
               JacobPattern: []
                 LargeScale: 'on'
                  MaxNodes: []
                 MaxPCGIter: []
               MaxProjCGIter: []
                  MaxSQPIter: []
                    MaxTime: []
               MeritFunction: []
                  MinAbsMax: []
         NoStopIfFlatInfeas: []
             ObjectiveLimit: []
       PhaseOneTotalScaling: []
             Preconditioner: []
          PrecondBandWidth: []
             RelLineSrchBnd: []
    RelLineSrchBndDuration: []
              ScaleProblem: []
        SubproblemAlgorithm: []
                    TolCon: []
                 TolConSQP: []
                TolGradCon: []
                    TolPCG: []
                  TolProjCG: []
               TolProjCGAbs: []
                  TypicalX: []
                UseParallel: []

Calculated minimizer:

x =

    1.0000    1.0000

```

```
Calculated Minimum value: 6.6115e-11
Norm of gradient at calculated Minimum: 0.00029178
       iterations: 39
        funcCount: 50
         stepsize: 4.0089e-04
     lssteplength: 1
    firstorderopt: 2.6299e-04
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
    gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    6.624363e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                      Options relative
    norm(gradient) =   6.62e-07             OptimalityTolerance =   1e-06 (default)'

==========================================================
Testing objective function f_b with options:
                   Display: 'off'
                MaxFunEvals: []
                    MaxIter: []
                     TolFun: []
                       TolX: []
                FunValCheck: []
                  OutputFcn: []
                   PlotFcns: []
              ActiveConstrTol: []
                  Algorithm: 'trust-region'
    AlwaysHonorConstraints: []
            DerivativeCheck: []
                Diagnostics: []
              DiffMaxChange: []
              DiffMinChange: []
             FinDiffRelStep: []
                FinDiffType: []
          GoalsExactAchieve: []
                GradConstr: []
                    GradObj: 'on'
                    HessFcn: []
                    Hessian: 'on'
                   HessMult: []
                HessPattern: []
                 HessUpdate: []
           InitBarrierParam: []
     InitTrustRegionRadius: []
                   Jacobian: []
                  JacobMult: []
                JacobPattern: []
                  LargeScale: 'on'
                   MaxNodes: []
                  MaxPCGIter: []
                MaxProjCGIter: []
                  MaxSQPIter: []
                     MaxTime: []
                MeritFunction: []
                   MinAbsMax: []
          NoStopIfFlatInfeas: []
              ObjectiveLimit: []
       PhaseOneTotalScaling: []
              Preconditioner: []
            PrecondBandWidth: []
              RelLineSrchBnd: []
    RelLineSrchBndDuration: []
                ScaleProblem: []
       SubproblemAlgorithm: []
                     TolCon: []
                  TolConSQP: []
                 TolGradCon: []
                     TolPCG: []
```

```
                TolProjCG: []
             TolProjCGAbs: []
                TypicalX: []
              UseParallel: []

Calculated minimizer:

x =

    1.0000    1.0000

Calculated Minimum value: 1.931e-17
Norm of gradient at calculated Minimum: 3.3208e-08
Condition of Hessian at calculated Minimum: 2508.0095
         iterations: 30
          funcCount: 31
           stepsize: 2.3363e-05
       cgiterations: 25
      firstorderopt: 2.7783e-08
          algorithm: 'trust-region'
            message: 'Local minimum found.  Optimization completed because the size of
    the gradient is less than the default value of the optimality tolerance.  Stopping
     criteria details:  Optimization completed: The first-order optimality measure,
    2.778283e-08,  is less than options.OptimalityTolerance = 1.000000e-06, and no
    negative/zero curvature is detected in the trust-region model.  Optimization
    Metric                                  Options relative first-order optimality
    =  2.78e-08  OptimalityTolerance =  1e-06 (default)'
    constrviolation: []

================================================================
================================================================
C
================================================================
Testing objective function f_c with options:
                   Display: 'off'
              MaxFunEvals: []
                  MaxIter: []
                   TolFun: []
                     TolX: []
              FunValCheck: []
                OutputFcn: []
                 PlotFcns: []
           ActiveConstrTol: []
                 Algorithm: []
    AlwaysHonorConstraints: []
           DerivativeCheck: []
               Diagnostics: []
             DiffMaxChange: []
             DiffMinChange: []
           FinDiffRelStep: []
               FinDiffType: []
         GoalsExactAchieve: []
               GradConstr: []
                  GradObj: 'on'
                  HessFcn: []
                  Hessian: []
                 HessMult: []
              HessPattern: []
               HessUpdate: []
         InitBarrierParam: []
     InitTrustRegionRadius: []
                 Jacobian: []
                JacobMult: []
             JacobPattern: []
               LargeScale: 'off'
                 MaxNodes: []
               MaxPCGIter: []
            MaxProjCGIter: []
```

```
                MaxSQPIter: []
                   MaxTime: []
             MeritFunction: []
                MinAbsMax: []
          NoStopIfFlatInfeas: []
             ObjectiveLimit: []
       PhaseOneTotalScaling: []
             Preconditioner: []
           PrecondBandWidth: []
             RelLineSrchBnd: []
    RelLineSrchBndDuration: []
            ScaleProblem: []
       SubproblemAlgorithm: []
                   TolCon: []
                TolConSQP: []
                TolGradCon: []
                   TolPCG: []
                 TolProjCG: []
              TolProjCGAbs: []
                 TypicalX: []
              UseParallel: []

================================================
Results for n= 10
Calculated minimizer:

x =

    0.3949    0.3811    0.3767    0.3745    0.3731    0.3722    0.3716    0.3712    0.3708
      0.3705

Calculated Minimum value: -19.9644
Norm of gradient at calculated Minimum: 1.0378e-05
       iterations: 29
        funcCount: 55
         stepsize: 3.0515e-06
     lssteplength: 1
    firstorderopt: 8.0412e-06
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
     gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    2.967900e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                      Options relative
    norm(gradient) =   2.97e-07             OptimalityTolerance =   1e-06 (default)'

================================================
Results for n= 100
Calculated Minimum value: -1857.764
Norm of gradient at calculated Minimum: 0.00082223
       iterations: 79
        funcCount: 160
         stepsize: 2.0476e-05
     lssteplength: 0.1895
    firstorderopt: 1.7558e-04
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
     gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    6.702941e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                      Options relative
    norm(gradient) =   6.70e-07             OptimalityTolerance =   1e-06 (default)'

================================================
Results for n= 1000
Calculated Minimum value: -184123.6576
Norm of gradient at calculated Minimum: 0.010364
       iterations: 203
```

```
        funcCount: 408
         stepsize: 2.7483e-05
     lssteplength: 0.2436
    firstorderopt: 0.0019
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
     gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    7.088470e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                   Options relative
    norm(gradient) =   7.09e-07             OptimalityTolerance =   1e-06 (default)'

================================================
Testing objective function f_c with options:
                   Display: 'off'
              MaxFunEvals: []
                  MaxIter: []
                   TolFun: []
                     TolX: []
              FunValCheck: []
                OutputFcn: []
                 PlotFcns: []
           ActiveConstrTol: []
                 Algorithm: []
    AlwaysHonorConstraints: []
           DerivativeCheck: []
               Diagnostics: []
             DiffMaxChange: []
             DiffMinChange: []
           FinDiffRelStep: []
              FinDiffType: []
         GoalsExactAchieve: []
               GradConstr: []
                  GradObj: 'on'
                  HessFcn: []
                  Hessian: 'off'
                 HessMult: []
              HessPattern: []
                HessUpdate: []
          InitBarrierParam: []
     InitTrustRegionRadius: []
                 Jacobian: []
                JacobMult: []
              JacobPattern: []
                LargeScale: 'on'
                 MaxNodes: []
                MaxPCGIter: []
              MaxProjCGIter: []
                MaxSQPIter: []
                   MaxTime: []
              MeritFunction: []
                  MinAbsMax: []
        NoStopIfFlatInfeas: []
           ObjectiveLimit: []
      PhaseOneTotalScaling: []
             Preconditioner: []
          PrecondBandWidth: []
            RelLineSrchBnd: []
    RelLineSrchBndDuration: []
              ScaleProblem: []
       SubproblemAlgorithm: []
                   TolCon: []
                TolConSQP: []
                TolGradCon: []
                    TolPCG: []
                TolProjCG: []
             TolProjCGAbs: []
                 TypicalX: []
```

```
                UseParallel: []

================================================================
Results for n= 10
Calculated minimizer:

x =

    0.3949    0.3811    0.3767    0.3745    0.3731    0.3722    0.3716    0.3712    0.3708
      0.3705

Calculated Minimum value: -19.9644
Norm of gradient at calculated Minimum: 1.0378e-05
       iterations: 29
        funcCount: 55
         stepsize: 3.0515e-06
     lssteplength: 1
    firstorderopt: 8.0412e-06
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
    gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    2.967900e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                      Options relative
    norm(gradient) =   2.97e-07             OptimalityTolerance =   1e-06 (default)'

================================================================
Results for n= 100
Calculated Minimum value: -1857.764
Norm of gradient at calculated Minimum: 0.00082223
       iterations: 79
        funcCount: 160
         stepsize: 2.0476e-05
     lssteplength: 0.1895
    firstorderopt: 1.7558e-04
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
    gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    6.702941e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                      Options relative
    norm(gradient) =   6.70e-07             OptimalityTolerance =   1e-06 (default)'

================================================================
Results for n= 1000
Calculated Minimum value: -184123.6576
Norm of gradient at calculated Minimum: 0.010364
       iterations: 203
        funcCount: 408
         stepsize: 2.7483e-05
     lssteplength: 0.2436
    firstorderopt: 0.0019
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
    gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    7.088470e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                      Options relative
    norm(gradient) =   7.09e-07             OptimalityTolerance =   1e-06 (default)'

================================================================
Testing objective function f_c with options:
                  Display: 'off'
             MaxFunEvals: []
                 MaxIter: []
                  TolFun: []
                    TolX: []
             FunValCheck: []
```

```
                 OutputFcn: []
                   PlotFcns: []
             ActiveConstrTol: []
                 Algorithm: 'trust-region'
    AlwaysHonorConstraints: []
            DerivativeCheck: []
                Diagnostics: []
              DiffMaxChange: []
              DiffMinChange: []
             FinDiffRelStep: []
                FinDiffType: []
          GoalsExactAchieve: []
                GradConstr: []
                    GradObj: 'on'
                    HessFcn: []
                    Hessian: 'on'
                   HessMult: []
                HessPattern: []
                 HessUpdate: []
           InitBarrierParam: []
      InitTrustRegionRadius: []
                   Jacobian: []
                  JacobMult: []
                JacobPattern: []
                  LargeScale: 'on'
                   MaxNodes: []
                  MaxPCGIter: []
                MaxProjCGIter: []
                  MaxSQPIter: []
                     MaxTime: []
                MeritFunction: []
                     MinAbsMax: []
          NoStopIfFlatInfeas: []
              ObjectiveLimit: []
        PhaseOneTotalScaling: []
              Preconditioner: []
           PrecondBandWidth: []
             RelLineSrchBnd: []
     RelLineSrchBndDuration: []
                ScaleProblem: []
         SubproblemAlgorithm: []
                      TolCon: []
                   TolConSQP: []
                   TolGradCon: []
                      TolPCG: []
                   TolProjCG: []
                TolProjCGAbs: []
                    TypicalX: []
                 UseParallel: []

================================================================
Results for n= 10
Calculated minimizer:

x =

    0.3949    0.3812    0.3767    0.3745    0.3731    0.3723    0.3716    0.3712    0.3708
      0.3705

Calculated Minimum value: -19.9644
Norm of gradient at calculated Minimum: 0.00012999
Condition of Hessian at calculated Minimum: 10.523
          iterations: 12
           funcCount: 13
            stepsize: 0.0015
         cgiterations: 8
        firstorderopt: 9.4470e-05
            algorithm: 'trust-region'
```

```
            message: 'Local minimum possible.  fminunc  stopped because the final change
    in function value relative to  its  initial value is less than the default value of the
     function tolerance.  Stopping  criteria details:  Optimization  stopped because
    the relative objective function value is changing by less than options.
    FunctionTolerance = 1.000000e-06.  Optimization  Metric
            Options relative change objective =  2.20e-07  FunctionTolerance =
     1e-06 (default)'
    constrviolation: []

==========================================================
Results for n= 100
Calculated Minimum value: -1857.764
Norm of gradient at calculated Minimum: 1.8685e-07
Condition of Hessian at calculated Minimum: 100.0717
         iterations: 11
          funcCount: 12
           stepsize: 2.2483e-05
       cgiterations: 9
      firstorderopt: 1.8658e-07
          algorithm: 'trust-region'
            message: 'Local minimum found.  Optimization  completed because the size of
    the gradient is less than the default value of the optimality tolerance.  Stopping
     criteria details:  Optimization  completed: The first-order optimality measure,
    1.865793e-07,  is  less than options.OptimalityTolerance = 1.000000e-06, and no
    negative/zero curvature  is detected in the trust-region model.  Optimization
    Metric                                  Options relative  first-order optimality
    =  1.87e-07  OptimalityTolerance =  1e-06 (default)'
    constrviolation: []

==========================================================
Results for n= 1000
Calculated Minimum value: -184123.6576
Norm of gradient at calculated Minimum: 0.0010339
Condition of Hessian at calculated Minimum: 999.9952
         iterations: 16
          funcCount: 17
           stepsize: 0.0087
       cgiterations: 12
      firstorderopt: 1.0831e-04
          algorithm: 'trust-region'
            message: 'Local minimum possible.  fminunc  stopped because the final change
    in function value relative to  its  initial value is less than the default value of the
     function tolerance.  Stopping  criteria details:  Optimization  stopped because
    the relative objective function value is changing by less than options.
    FunctionTolerance = 1.000000e-06.  Optimization  Metric
            Options relative change objective =  2.83e-08  FunctionTolerance =
     1e-06 (default)'
    constrviolation: []

==========================================================
==========================================================
(D)
==========================================================
Testing objective function f_d with options:
                  Display: 'off'
             MaxFunEvals: []
                 MaxIter: []
                  TolFun: []
                    TolX: []
             FunValCheck: []
               OutputFcn: []
                PlotFcns: []
          ActiveConstrTol: []
                Algorithm: []
    AlwaysHonorConstraints: []
          DerivativeCheck: []
              Diagnostics: []
            DiffMaxChange: []
```

```
              DiffMinChange: []
             FinDiffRelStep: []
               FinDiffType: []
         GoalsExactAchieve: []
                GradConstr: []
                   GradObj: 'on'
                   HessFcn: []
                   Hessian: []
                  HessMult: []
                HessPattern: []
                HessUpdate: []
          InitBarrierParam: []
     InitTrustRegionRadius: []
                  Jacobian: []
                 JacobMult: []
               JacobPattern: []
                 LargeScale: 'off'
                   MaxNodes: []
                  MaxPCGIter: []
               MaxProjCGIter: []
                  MaxSQPIter: []
                   MaxTime: []
              MeritFunction: []
                  MinAbsMax: []
        NoStopIfFlatInfeas: []
             ObjectiveLimit: []
      PhaseOneTotalScaling: []
             Preconditioner: []
          PrecondBandWidth: []
             RelLineSrchBnd: []
    RelLineSrchBndDuration: []
              ScaleProblem: []
     SubproblemAlgorithm: []
                   TolCon: []
                TolConSQP: []
               TolGradCon: []
                   TolPCG: []
                TolProjCG: []
              TolProjCGAbs: []
                  TypicalX: []
               UseParallel: []

==========================================================
Results for n= 10
Calculated minimizer:

x =

    0.7073    1.4114    2.1295    2.8233    3.5305    4.2432    4.9540    5.6603    6.3619
      7.0693

 Calculated Minimum value: -16769.7156
Norm of gradient at calculated Minimum: 0.029042
       iterations: 54
        funcCount: 57
         stepsize: 0.0481
     lssteplength: 1
    firstorderopt: 0.0135
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
     gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    9.368012e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                    Options relative
    norm(gradient) =  9.37e-07            OptimalityTolerance =  1e-06 (default)'

==========================================================
Results for n= 100
```

```
Calculated Minimum value: -10987678753.1027
Norm of gradient at calculated Minimum: 13.4558
       iterations: 78
        funcCount: 83
         stepsize: 0.2896
     lssteplength: 1
    firstorderopt: 3.7624
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization  completed because the size of the
     gradient is less  t h a n the  default value of the optimality tolerance.  Stopping
    criteria details:  Optimization  completed: The first-order optimality measure,
    6.246662e-07, is less  than  options.OptimalityTolerance = 1.000000e-06.
        Optimization  Metric                                        O p t i o n s relative
    norm(gradient) =  6.25e-07          OptimalityTolerance =  1e-06 (default)'

==========================================================
Results for n= 1000
Calculated Minimum value: -1.051366255854342e+16
Norm of gradient at calculated Minimum: 25147.6496
       iterations: 152
        funcCount: 161
         stepsize: 3.3739
     lssteplength: 1
    firstorderopt: 3.6475e+03
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization  completed because the size of the
     gradient is less  t h a n the  default value of the optimality tolerance.  Stopping
    criteria details:  Optimization  completed: The first-order optimality measure,
    5.936574e-07, is less  than  options.OptimalityTolerance = 1.000000e-06.
        Optimization  Metric                                        O p t i o n s relative
    norm(gradient) =  5.94e-07          OptimalityTolerance =  1e-06 (default)'

==========================================================
Testing objective function f_d with options:
                  Display: 'off'
             MaxFunEvals: []
                 MaxIter: []
                  TolFun: []
                    TolX: []
              FunValCheck: []
                OutputFcn: []
                 PlotFcns: []
           ActiveConstrTol: []
                Algorithm: []
    AlwaysHonorConstraints: []
           DerivativeCheck: []
               Diagnostics: []
             DiffMaxChange: []
             DiffMinChange: []
           FinDiffRelStep: []
               FinDiffType: []
         GoalsExactAchieve: []
               GradConstr: []
                  GradObj: 'on'
                  HessFcn: []
                  Hessian: 'off'
                 HessMult: []
              HessPattern: []
               HessUpdate: []
          InitBarrierParam: []
     InitTrustRegionRadius: []
                 Jacobian: []
                JacobMult: []
             JacobPattern: []
               LargeScale: 'on'
                 MaxNodes: []
               MaxPCGIter: []
           MaxProjCGIter: []
```

```
                MaxSQPIter: []
                   MaxTime: []
             MeritFunction: []
                MinAbsMax: []
          NoStopIfFlatInfeas: []
             ObjectiveLimit: []
       PhaseOneTotalScaling: []
             Preconditioner: []
          PrecondBandWidth: []
             RelLineSrchBnd: []
    RelLineSrchBndDuration: []
             ScaleProblem: []
       SubproblemAlgorithm: []
                   TolCon: []
                TolConSQP: []
                TolGradCon: []
                   TolPCG: []
                TolProjCG: []
             TolProjCGAbs: []
                TypicalX: []
             UseParallel: []

================================================================
Results for n= 10
Calculated minimizer:

x =

    0.7073    1.4114    2.1295    2.8233    3.5305    4.2432    4.9540    5.6603    6.3619
      7.0693

Calculated Minimum value: -16769.7156
Norm of gradient at calculated Minimum: 0.029042
       iterations: 54
        funcCount: 57
         stepsize: 0.0481
     lssteplength: 1
    firstorderopt: 0.0135
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
    gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    9.368012e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                      Options relative
    norm(gradient) =   9.37e-07            OptimalityTolerance =   1e-06 (default)'

================================================================
Results for n= 100
Calculated Minimum value: -10987678753.1027
Norm of gradient at calculated Minimum: 13.4558
       iterations: 78
        funcCount: 83
         stepsize: 0.2896
     lssteplength: 1
    firstorderopt: 3.7624
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
    gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    6.246662e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                      Options relative
    norm(gradient) =   6.25e-07            OptimalityTolerance =   1e-06 (default)'

================================================================
Results for n= 1000
Calculated Minimum value: -1.051366255854342e+16
Norm of gradient at calculated Minimum: 25147.6496
       iterations: 152
```

```
        funcCount: 161
         stepsize: 3.3739
     lssteplength: 1
    firstorderopt: 3.6475e+03
        algorithm: 'quasi-newton'
          message: 'Local minimum found.  Optimization completed because the size of the
     gradient is less than the default value of the optimality tolerance.  Stopping
    criteria details:  Optimization completed: The first-order optimality measure,
    5.936574e-07, is less than options.OptimalityTolerance = 1.000000e-06.
        Optimization Metric                                    Options relative
    norm(gradient) =   5.94e-07             OptimalityTolerance =   1e-06 (default)'

================================================
Testing objective function f_d with options:
                   Display: 'off'
              MaxFunEvals: []
                  MaxIter: []
                   TolFun: []
                     TolX: []
              FunValCheck: []
                OutputFcn: []
                 PlotFcns: []
           ActiveConstrTol: []
                Algorithm: 'trust-region'
    AlwaysHonorConstraints: []
           DerivativeCheck: []
               Diagnostics: []
             DiffMaxChange: []
             DiffMinChange: []
           FinDiffRelStep: []
              FinDiffType: []
         GoalsExactAchieve: []
               GradConstr: []
                  GradObj: 'on'
                  HessFcn: []
                  Hessian: 'on'
                 HessMult: []
              HessPattern: []
               HessUpdate: []
          InitBarrierParam: []
     InitTrustRegionRadius: []
                 Jacobian: []
                JacobMult: []
              JacobPattern: []
                LargeScale: 'on'
                 MaxNodes: []
               MaxPCGIter: []
           MaxProjCGIter: []
               MaxSQPIter: []
                  MaxTime: []
           MeritFunction: []
                MinAbsMax: []
        NoStopIfFlatInfeas: []
           ObjectiveLimit: []
     PhaseOneTotalScaling: []
            Preconditioner: []
         PrecondBandWidth: []
           RelLineSrchBnd: []
    RelLineSrchBndDuration: []
            ScaleProblem: []
      SubproblemAlgorithm: []
                   TolCon: []
                TolConSQP: []
               TolGradCon: []
                   TolPCG: []
                TolProjCG: []
             TolProjCGAbs: []
                 TypicalX: []
```

```
                UseParallel: []

================================================================
Results for n= 10
Calculated minimizer:

x =

    0.7976    0.6364    2.8053    3.8166    4.0723    3.9870    3.7920    3.6033    3.4749
     11.9926

Calculated Minimum value: -16758.8021
Norm of gradient at calculated Minimum: 11.9204
Condition of Hessian at calculated Minimum: 1254.7567
         iterations: 401
          funcCount: 402
           stepsize: 0.0255
       cgiterations: 1936
      firstorderopt: 6.9473
          algorithm: 'trust-region'
            message: 'Solver stopped prematurely.  fminunc  stopped because it exceeded
    the iteration limit,  options  .MaxIterations = 400 (the default value).'
    constrviolation: []

================================================================
Results for n= 100
Calculated Minimum value: -10968504409.5875
Norm of gradient at calculated Minimum: 86246.1883
Condition of Hessian at calculated Minimum: 15458.5502
         iterations: 401
          funcCount: 402
           stepsize: 0.1551
       cgiterations: 8032
      firstorderopt: 4.6548e+04
          algorithm: 'trust-region'
            message: 'Solver stopped prematurely.  fminunc  stopped because it exceeded
    the iteration limit,  options  .MaxIterations = 400 (the default value).'
    constrviolation: []

================================================================
Results for n= 1000
Calculated Minimum value: -1.051215141638858e+16
Norm of gradient at calculated Minimum: 3110412089.4636
Condition of Hessian at calculated Minimum: 203328.5383
         iterations: 253
          funcCount: 254
           stepsize: 4.9497
       cgiterations: 818
      firstorderopt: 4.9217e+08
          algorithm: 'trust-region'
            message: 'Local minimum possible.  fminunc  stopped because the final change
    in function value relative to  its  initial value is less than the default value of the
     function tolerance.  Stopping  criteria details:  Optimization  stopped because
    the relative objective function value is changing by less than options.
    FunctionTolerance = 1.000000e-06.  Optimization  Metric
            Options relative change objective =  9.91e-07  FunctionTolerance =
     1e-06 (default)'
    constrviolation: []

================================================================
```

src/Test1.txt

2.4 Interpretation

We will proceed to extract the relevant information of the above output for every function tested. For the subsequent sections we will use the following notation

1. O1 : LargeScale = off, GradObj = on
2. O2 : LargeScale = on, GradObj = on, Hessian = off
3. O3 : LargeScale = on, GradObj = on, Hessian = on

Also notice that the default "quasi-newton" algorithm used in `fminunc` does not make use of a provided Hessian, therefore the algorithm has been changed to "trust-region" in O3 when an analytic Hessian is supplied. Additionally, where an analytic Hessian is supplied (i.e. O3) the condition of the Hessian at the approximated minimizer has been calculated.

2.4.1 `f_a`

Option	`fminunc` output description	Minimum value	Norm of gradient	iterations	cond(H)
O1	Local Min found	$1.1571 \cdot 10^{-13}$	$5.8763 \cdot 10^{-7}$	8	/
O2	Local Min found	$1.157 \cdot 10^{-13}$	$5.8763 \cdot 10^{-7}$	8	/
O3	Local Min possible	$2.0406 \cdot 10^{-7}$	0.00079236	12	10

All options yield a very small minimum value after few iterations, however comparing the order of magnitude of the different ouputs (i.e. calculated minimal values and norms of gradients at calculated minimum) one notices, that O1 and O2 seem to be about twice as accurate (i.e. square of the error in O3) as O3 while also needing fewer iterations. An educated guess would attribute this rather strange disparity to the use of different algorithms in O1,O2 ("quasi-newton") and O3("trust-region").

2.4.2 `f_b`

Option	`fminunc` output description	Minimum value	Norm of gradient	iterations	cond(H)
O1	Local Min found	$6.6115e-11$	0.00029178	39	/
O2	Local Min found	$6.6115e-11$	0.00029178	39	/
O3	Local Min possible	$1.931e-17$	$3.3208e-08$	30	2508.0095

Again all options yield a minimum value very close to zero. This time, however, O3 performs better, in that it calculates a minimizer whose value is smaller by a factor of 10^{-6} than that produced by O1, O2 where additionally the norm of the gradient is significantly smaller(by a factor of about 10^{-4}) than the norm of the gradient of the outputs of O1 and O3. This is not very surprising, given the fact that in O3 an analytic Hessian is supplied, allowing for faster convergence. Interestingly enough, `fminunc` outputs "local Min possible" with O3, probably due to positive semi-definiteness of the Hessian in the approximated minimizer.

2.4.3 `f_c`

1. $n = 10$

Option	`fminunc` output description	Minimum value	Norm of gradient	iterations	cond(H)
O1	Local Min found	-19.9644	$1.0378e-05$	29	/
O2	Local Min found	-19.9644	$1.0378e-05$	29	/
O3	Local Min possible	-19.9644	0.00012999	12	10.523

2. $n = 100$

Option	`fminunc` output description	Minimum value	Norm of gradient	iterations	cond(H)
O1	Local Min found	-1857.764	0.00082223	79	/
O2	Local Min found	-1857.764	0.00082223	29	/
O3	Local Min found	-1857.764	$1.8685e-07$	11	100.0717

3. $n = 1000$

Option	`fminunc` output description	Minimum value	Norm of gradient	iterations	cond(H)
O1	Local Min found	-184123.6576	0.010364	203	/
O2	Local Min found	-184123.6576	0.010364	203	/
O3	Local Min possible	-184123.6576	0.0010339	16	999.9952

For every input (i.e. $n = 10, 100, 1000$) `fminunc` produced the same result independent of the options used. Differences can be observed in the norms of the gradients at the approximated minimizers and the amount of iterations needed. Here, the most striking contrast can be found for $n = 1000$: While O1 and O2 require 2 about 200 iterations, O3 terminates after 16 iterations while also producing an approximated minimizers where the norm of the gradient is smaller than the output of O1,O2 by a factor of 10. Concerning iterations, O3 seems to perform better allround. For $n = 100$ we also obtain an approximate minimizer where the norm of the gradient is smaller by a factor of 10^3 compared to what O1 and O2 outputs. The fact that for $n = 10, 1000$ O3 outputs "local minimum possible" may again be attributed to positive semidefiniteness of the Hessian at the approximated minimizer.

2.4.4 `f_d`

1. $n = 10$

Option	`fminunc` output description	Minimum value	Norm of gradient	iterations	cond(H)
O1	Local Min found	-16769.7156	0.029042	54	/
O2	Local Min found	-16769.7156	0.029042	54	/
O3	Solver stopped prematurely; exceeded it limit	-16758.8021	11.9204	401	1254.7567

2. $n = 100$

Option	`fminunc` output description	Minimum value	Norm of gradient	iterations	cond(H)
O1	Local Min found	-10987678753.1027	13.4558	78	/
O2	Local Min found	-10987678753.1027	13.4558	78	/
O3	Solver stopped prematurely; exceeded it limit	-10968504409.5875	86246.1883	401	15458.5502

3. $n = 1000$

Option	`fminunc` output description	Minimum value	Norm of gradient	iterations	cond(H)
O1	Local Min found	$-1.051366255854342e+16$	25147.6496	152	/
O2	Local Min found	$-1.051366255854342e+16$	25147.6496	152	/
O3	Local Min possible	$-1.051215141638858e+16$	3110412089.4636	253	203328.5383

What makes the test of function `f_d` above stand out in comparison to the tests beforehand is that `fminunc` terminates prematurely after having calculated 400 iterations without meeting the default convergence criteria. Observe, that the approximated minimum values of O1 and O2 seem to coincide for all inputs, as does the norm of the gradient and the number of required iterations. We remark, that for $n = 100$ and especially for $n = 1000$, the norm of the gradient at the calculated minimizer is no longer close to zero, meaning we are no longer "close" to a stationary point, yet `fminunc`'s output description states "local minimum found". What is even more interesting is the fact, that O3 stops prematurely, needing more than the default maximal number of iterations, when O1 and O2 (without using the Hessian) terminate after a reasonable amount of iterations. This effect might be caused by making use of a badly conditioned Hessian, slowing down convergence. The calculated minimum values of O3 seem to be coinciding to the outputs of O1 and O2 up to some significant digit, increasing the default maxiter bound would allow for more accurate approximation. Striking is the astronomical norm of the gradient in O3 with $n = 1000$, yet `fminunc` describes the output to be a possible local minimum.

3 Exercise 3

3.1 Source Code

```
fprintf('========== (a) ==========\n');
%minimum should be at (0,0), f(min) should be 0
x0 = [10, -3];
tic
[x, fval, exitflag, output] = fminsearch(@f_aH, x0);
toc
if exitflag == 0
    fprintf('Iterations or function evaluations exceeded options.')
    return
end
fprintf('\nSolution found at: [ ');
fprintf(' %g ',x);
fprintf(' ]\nFunction value: %.8f\n\n', fval);
output

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

fprintf('========== (b) ==========\n');
%minimum should be at (1,1), f(min) should be 0
x0 = [-1, 2];
tic
[x, fval, exitflag, output] = fminsearch(@f_bH, x0);
toc
if exitflag == 0
    fprintf('Iterations or function evaluations exceeded options.')
    return
end
fprintf('\nSolution found at: [ ');
fprintf(' %g ',x);
fprintf(' ]\nFunction value: %.8f\n\n', fval);
output

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

options = optimset('MaxFunEvals', 1000000, 'MaxIter', 1000000);
fprintf('========== (c) ==========\n');
for i = 1:3
    fprintf('SIZE: n = 10^%i\n',i);
    x0 = 5*ones(10^i, 1);
    tic
    [x, fval, exitflag, output] = fminsearch(@f_cH, x0, options);
    toc
    if exitflag == 0
        fprintf('Iterations or function evaluations exceeded options.')
        return
    end
    if i < 3
        fprintf('\nSolution found at: [\n');
        fprintf('%g\n',x);
        fprintf(']');
    end
    fprintf('\nFunction value: %.8f\n', fval);
    output
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

options = optimset('MaxFunEvals', 1000000, 'MaxIter', 1000000);
fprintf('========== (d) ==========\n');
for i = 1:3
    fprintf('SIZE: n = 10^%i\n',i);
    x0 = [zeros(1,10^i-1),10^(i+1)];
    tic
    [x, fval, exitflag, output] = fminsearch(@f_dH, x0, options);
    toc
    if exitflag == 0
        fprintf('Iterations or function evaluations exceeded options.')
```

```
        return
    end
    if i == 1
        fprintf('\nSolution found at: [\n');
        fprintf('%g\n',x);
        fprintf(']');
    end
    fprintf('\nFunction value: %.8f\n', fval);
    output
end
```

src/Pract1Ex3.m

3.2 Output

```
========== (a) ==========
Elapsed time is 0.012591 seconds.

Solution found at: [  -3.79534e-05   1.22752e-05  ]
Function value: 0.00000000


output =

  struct with fields:

    iterations: 52
     funcCount: 97
     algorithm: 'Nelder-Mead simplex direct search'
       message: 'Optimization terminated:    the current x satisfies the termination
    criteria using OPTIONS.TolX of 1.000000e-04     and F(X) satisfies the convergence
    criteria using OPTIONS.TolFun of 1.000000e-04    '

========== (b) ==========
Elapsed time is 0.010781 seconds.

Solution found at: [  0.999991   0.999983  ]
Function value: 0.00000000


output =

  struct with fields:

    iterations: 105
     funcCount: 195
     algorithm: 'Nelder-Mead simplex direct search'
       message: 'Optimization terminated:    the current x satisfies the termination
    criteria using OPTIONS.TolX of 1.000000e-04     and F(X) satisfies the convergence
    criteria using OPTIONS.TolFun of 1.000000e-04    '

========== (c) ==========
SIZE: n = 10^1

Exiting: Maximum number of function evaluations has been exceeded
         - increase MaxFunEvals option.
         Current function value: 53.708279

Elapsed time is 0.183353 seconds.
Iterations or function evaluations exceeded options.

==> increase MaxFunEvals (default: 200*#variables) to 100000
Now:

Exiting: Maximum number of iterations has been exceeded
         - increase MaxIter option.
```

```
         Current function value: 53.625224

==> increase MaxIter (same default) to 100000
Now:

SIZE: n = 10^1
Elapsed time is 0.140587 seconds.

Solution found at: [
0.394952
0.381119
0.376755
0.374441
0.373128
0.372251
0.37164
0.371131
0.370765
0.370491
]
Function value: -19.96439719

output =

  struct with fields:

    iterations: 3883
     funcCount: 5422
     algorithm: 'Nelder-Mead simplex direct search'
       message: 'Optimization terminated:    the current x satisfies the termination
    criteria using OPTIONS.TolX of 1.000000e-04     and F(X) satisfies the convergence
    criteria using OPTIONS.TolFun of 1.000000e-04    '

SIZE: n = 10^2

Exiting: Maximum number of function evaluations has been exceeded
         - increase MaxFunEvals option.
         Current function value: 28071.812796

Elapsed time is 279.552686 seconds.
Iterations or function evaluations exceeded options.

==> increase both Max-values to 1000000
Now:

SIZE: n = 10^2
Elapsed time is 783.989220 seconds.

Function value: 27997.73143250

output =

  struct with fields:

    iterations: 358765
     funcCount: 385746
     algorithm: 'Nelder-Mead simplex direct search'
       message: 'Optimization terminated:    the current x satisfies the termination
    criteria using OPTIONS.TolX of 1.000000e-04     and F(X) satisfies the convergence
    criteria using OPTIONS.TolFun of 1.000000e-04    '

SIZE: n = 10^3
Operation terminated by user during fminsearch (line 322)

==================== (d) ================
SIZE: n = 10^1
Elapsed time is 0.012700 seconds.

```

```
Solution found at: [
0.482444
0.863983
1.8512
11.5311
7.55222
-7.94144
-0.775348
-0.633636
-5.43603
31.9502
]
Function value: -16468.38764339

output =

  struct with fields:

    iterations: 1098
     funcCount: 1642
     algorithm: 'Nelder-Mead simplex direct search'
       message: 'Optimization terminated:    the current x satisfies the termination
    criteria using OPTIONS.TolX of 1.000000e-04     and F(X) satisfies the convergence
    criteria using OPTIONS.TolFun of 1.000000e-04    '

SIZE: n = 10^2
Elapsed time is 7.013846 seconds.

Function value: -10409439951.64850044

output =

  struct with fields:

    iterations: 142685
     funcCount: 152272
     algorithm: 'Nelder-Mead simplex direct search'
       message: 'Optimization terminated:    the current x satisfies the termination
    criteria using OPTIONS.TolX of 1.000000e-04     and F(X) satisfies the convergence
    criteria using OPTIONS.TolFun of 1.000000e-04    '

SIZE: n = 10^3

Exiting: Maximum number of function evaluations has been exceeded
         - increase MaxFunEvals option.
         Current function value: -8563658281094496.000000

Elapsed time is 3127.988139 seconds.
Iterations or function evaluations exceeded options.
```

src/Pract1Ex3 output.txt

3.3 Interpretation

Function	n	`fminsearch` output description	Function value
f_a	-	termination criteria satisfied	0
f_b	-	termination criteria satisfied	0
f_c	10	termination criteria satisfied	53.625224
f_c	100	termination criteria satisfied	783.989220
f_c	1000	terminated because it took too long	-
f_d	10	termination criteria satisfied	16468.38764339
f_d	100	termination criteria satisfied	10409439951.64850044
f_d	1000	Maximum funevals exceeded	-

Disclaimer: fminsearch doesn't use the Hessian NOR the gradient, so for better efficiency we added a

"if nargout > 1" clause so that the gradient wouldn't be evaluated every time.

Both f_a and f_b converge to the desired solution very fast, though they need significantly more iterations than fminunc.

Initially, f_c with n=10 did not terminate with a solution since it immediately put out the message that the maximum number of function evaluations had been exceeded - "increase MaxFunEvals option". After setting this option from the default value (200 times the number of variables = 2000) to 100000, the same message was shown for the maximum number of iterations, so MaxIter was also increased. Now we got a solution, and even the same one as with fminunc.

With n=100, we had to increase both MaxFunEvals and MaxIter to 1.000.000 to get a result (it needed around 350.000 iterations and 380.000 evaluations). However, the function value was around 28000 in contrast to the -1850 from fminunc, so that can't be correct. Since for n=100, it already didn't even calculate the correct result, it took significantly longer than for n=10, we aborted the calculation for n=1000 after an hour.

In comparison to fminunc, f_d gave us different solution vectors for x, but the function values came really close to the ones from fminunc, at least until n=100. For n=1000, Matlab reached the maximum number of iterations, 1.000.000, and returned the current value, which was almost -1.e+16, the returned function value from fminunc.

4 Exercise 4

4.1 Source Code

```
options1 = optimoptions('fminunc','GradObj','off');
options2 = optimset('GradObj','off');

f = @(x) x(1)+10*max(x(1)^2+2*x(2)^2-1,0);
x0=[1,1];
s1 = fminunc(f,x0,options1)
s2 = fminsearch(f,x0,options2)

e1 = norm(s1-[-1,0])
e2 = norm(s2-[-1,0])
```

src/Prog1Ex4.m

```
>> Prog1Ex4

Local minimum possible.

fminunc stopped because it cannot decrease the objective function
along the current search direction.

<stopping criteria details>

s1 =

   -0.9997   -0.0168


s2 =

   -1.0000    0.0000


e1 =

    0.0168


e2 =

   4.3007e-05
```

src/test4_1.txt

4.2 Solution of `fminunc`

When applying `fminunc` to the problem considering the given non-continuously differentiable function

$$\min_{x \in \mathbb{R}^2} x_1 + 10 \max\{x_1^2 + 2x_2^2 - 1, 0\}$$

we get that $x = (x_1, x_2) \approx (-0.9997, -0.0168)$ solves the above equation where the exact solution should correspond to the vector $\overline{x} = (x_1, x_2) = (-1, 0)$. This implies a numerical error of $\|x - \overline{x}\| \approx 0.0168$ as can be obtained by the `MATLAB` source code in section 4.1.

4.3 Solution of `fminsearch`

If instead of `fminunc` the `MATLAB` command `fminsearch` is applied to the same problem as in section 4.2 we get the result of $x = (x_1, x_2) \approx (-1.0000, 0.0000)$. Consequently, we also get a numerical error much smaller in size and given by $\|x - \overline{x}\| \approx 4.3007 \cdot 10^{-5}$.

4.4 Interpretation

In the `MATLAB` output it states that `fminunc` stopped because it cannot decrease the objective function along the current search direction any further. Though, if stopping criteria details are displayed we get the following additional information.

```
1 Optimization stopped because the objective function cannot be decreased in the
2 current search direction. Either the predicted change in the objective function,
3 or the line search interval is less than eps.
```

src/test4_2.txt

Consequently, it must be the case that the line search algorithm in `fminunc` which makes use of numerical respectively analytic gradients is not appropriate for the considered problem. On the opposite, the simplex search method of `fminsearch` does not estimate according gradients using finite differences and therefore provides an adequate line search interval and ultimately a more reliable result.