
Table of Contents

1a) 1.2;1.2	1
1a) -1.2;1	1
Steepest descent	2
1b) BFGS	5

1a) 1.2;1.2

```
close all;
clear;
clc;
epsilon = 0.001;
x0 = [1.2; 1.2];

k = 1;
syms x1 x2 real;
x = [x1, x2];

f = @(x1, x2) 100*(x2-x1^2)^2 + (1-x1)^2;
xk = x0;
grad_f(x1, x2) = gradient(f, [x1,x2]);
hessian_f(x1,x2) = hessian(f, [x1, x2]);

p_k = newton_step(hessian_f, grad_f);
N = 20;

grad_value_k = double(subs(grad_f, [x1, x2], xk'));
num_alpha_eq_1_easy = 0;

while norm(grad_value_k,2) > epsilon
    x_easy(:,k) = xk;
    step = double(subs(p_k, [x1, x2], xk'));
    alpha_1 = backtracking_line_search(f, simplify(grad_f), xk,step);
    xk = xk + alpha_1*step;
    grad_value_k = double(subs(grad_f, [x1, x2], xk'));
    if alpha_1 == 1
        num_alpha_eq_1_easy = num_alpha_eq_1_easy + 1;
    end
    k = k+1;
    alphas_easy(k-1) = alpha_1;
end
```

1a) -1.2;1

```
x0 = [-1.2; 1];
k = 1;

syms x1 x2 real;
```

```

x = [x1, x2];

f = @(x1, x2) 100*(x2-x1^2)^2 + (1-x1)^2;
xk = x0;
grad_f(x1, x2) = gradient(f, [x1,x2]);
hessian_f(x1,x2) = hessian(f, [x1, x2]);

p_k = newton_step(hessian_f, grad_f);
N = 20;

grad_value_k = double(subs(grad_f, [x1, x2], xk'));
num_alpha_eq_1_difficult = 0;
while norm(grad_value_k,2) > epsilon
    x_difficult(:,k) = xk;
    step = double(subs(p_k, [x1, x2], xk'));
    alpha_1 = backtracking_line_search(f, simplify(grad_f), xk,step);
    xk = xk + alpha_1*step;
    k = k+1;
    grad_value_k = double(subs(grad_f, [x1, x2], xk'));
    if alpha_1 == 1
        num_alpha_eq_1_difficult = num_alpha_eq_1_difficult + 1;
    end
    alphas_difficult(k-1) = alpha_1;
end

```

Steepest descent

```

x0 = [-1.2; 1];
k = 1;

syms x1 x2 real;
x = [x1, x2];

f = @(x1, x2) 100*(x2-x1^2)^2 + (1-x1)^2;
xk = x0;
grad_f(x1, x2) = gradient(f, [x1,x2]);

N = 20;

grad_value_k = double(subs(grad_f, [x1, x2], xk'));
while norm(grad_value_k,2) > epsilon
    x_steepest_descent(:,k) = xk;
    step = -grad_value_k;
    alpha_1 = backtracking_line_search(f, simplify(grad_f), xk,step);
    xk = xk + alpha_1*step;
    k = k+1;
    grad_value_k = double(subs(grad_f, [x1, x2], xk'));
    if k == 30
        break;
    end
    k
end
disp(k);

```

$$k =$$

$$2$$

$$k =$$

$$3$$

$$k =$$

$$4$$

$$k =$$

$$5$$

$$k =$$

$$6$$

$$k =$$

$$7$$

$$k =$$

$$8$$

$$k =$$

$$9$$

$$k =$$

$$10$$

$$k =$$

$$11$$

$$k =$$

$$12$$

$$k =$$

$$13$$

$$k =$$

$$14$$

$$k =$$

$$15$$

$$k =$$

$$16$$

$$k =$$

$$17$$

$$k =$$

$$18$$

$$k =$$

$$19$$

$$k =$$

$$20$$

$$k =$$

$$21$$

$$k =$$

$$22$$

$$k =$$

23

$k =$

24

$k =$

25

$k =$

26

$k =$

27

$k =$

28

$k =$

29

30

1b) BFGS

```
%Algorithm 6.1

x0 = [-1.2; 1];
rho = 0.9;
dim = size(x0, 1);
k = 1;
H0 = eye(dim); %inverse hessian approximation
inverse_hessian = H0;
alpha_k = 1;
xk = x0;
x_bfgs = x0;
grad_value_k = double(subs(grad_f, [x1, x2], xk'));
num_alpha_eq_1_bfgs = 0;
while norm(grad_value_k,2) > epsilon
    step = -inverse_hessian*grad_value_k;
    alpha_1 = backtracking_line_search(f, simplify(grad_f), xk,step);
```

```

x_kp1 = xk + alpha_1*step;
sk = x_kp1-xk;
grad_value_k = double(subs(grad_f, [x1, x2], xk'));
grad_value_kp1 = double(subs(grad_f, [x1, x2], x_kp1'));

yk = grad_value_kp1-grad_value_k;

%compute next hessian
inverse_hessian = inv_hessian(sk, yk, inverse_hessian);

k = k+1
xk = x_kp1
grad_value_k = grad_value_kp1;
x_bfgs = [x_bfgs, xk];
if alpha_1 == 1
    num_alpha_eq_1_bfgs = num_alpha_eq_1_bfgs + 1;
end
alphas_bfgs(k-1) = alpha_1;
end

f1 = figure();
subplot(411);
T = 1:size(x_easy, 2);
plot(T, x_easy);
subplot(412);
T = 1:size(x_difficult, 2);
plot(T, x_difficult);
subplot(413);
T_bfgs = 1:size(x_bfgs, 2);
plot(T_bfgs, x_bfgs);
title('x-values');
legend('x_easy (newton)', 'x_difficult (newton)', 'x_bfgs');

f2 = figure();
plot_iter_rosenbrock(x_easy);
title('x_easy (newton)');

f3 = figure();
plot_iter_rosenbrock(x_difficult);
title('x_difficult (newton)');

f4 = figure();
plot_iter_rosenbrock(x_bfgs);
title('x_bfgs');
f5 = figure();
plot_iter_rosenbrock(x_steepest_descent);
title('x_steepest_descent');

movegui(f1, 'west');
movegui(f2, 'north');
movegui(f3, 'east');
movegui(f4, 'south');
movegui(f5, 'center');

```

$k =$

2

$xk =$

1.5552
2.1246

$k =$

3

$xk =$

1.5165
2.5692

$k =$

4

$xk =$

1.5200
2.3117

$k =$

5

$xk =$

1.5152
2.2960

$k =$

6

$xk =$

1.3754
1.8570

$k =$

7

$xk =$

1.4516
2.1029

$k =$

8

$xk =$

1.4197
2.0087

$k =$

9

$xk =$

1.2866
1.6245

$k =$

10

$xk =$

1.3533
1.8252

$k =$

11

$xk =$

1.3245

1.7484

$k =$

12

$xk =$

1.1856

1.3812

$k =$

13

$xk =$

1.2441

1.5462

$k =$

14

$xk =$

1.2099

1.4615

$k =$

15

$xk =$

1.1008

1.1942

$k =$

16

$xk =$

1.1506
1.3224

$k =$

17

$xk =$

1.1247
1.2629

$k =$

18

$xk =$

1.0364
1.0630

$k =$

19

$xk =$

1.0753
1.1555

$k =$

20

$xk =$

1.0568
1.1157

$k =$

21

$xk =$

1.0110
1.0188

$k =$

22

$xk =$

1.0170
1.0337

$k =$

23

$xk =$

1.0083
1.0165

$k =$

24

$xk =$

1.0008
1.0014

$k =$

25

$xk =$

1.0001
1.0002

$k =$

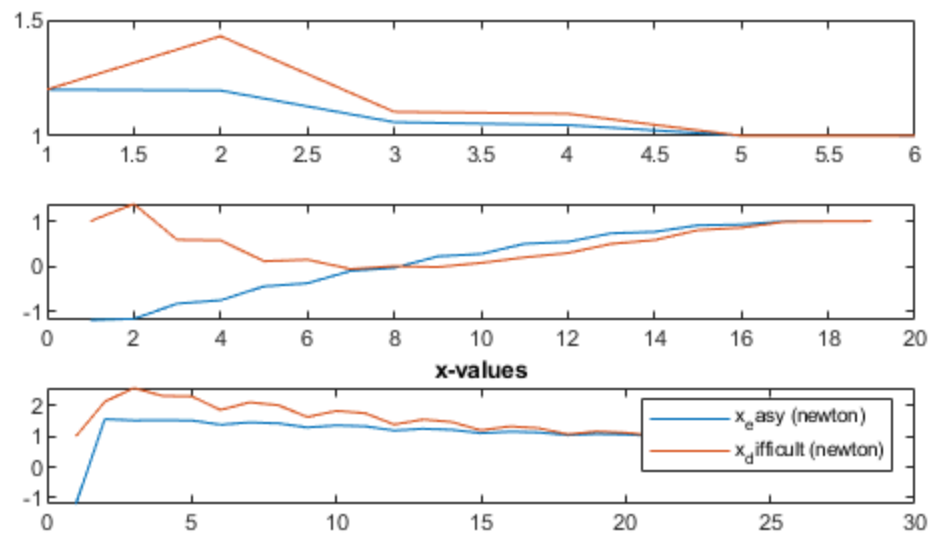
26

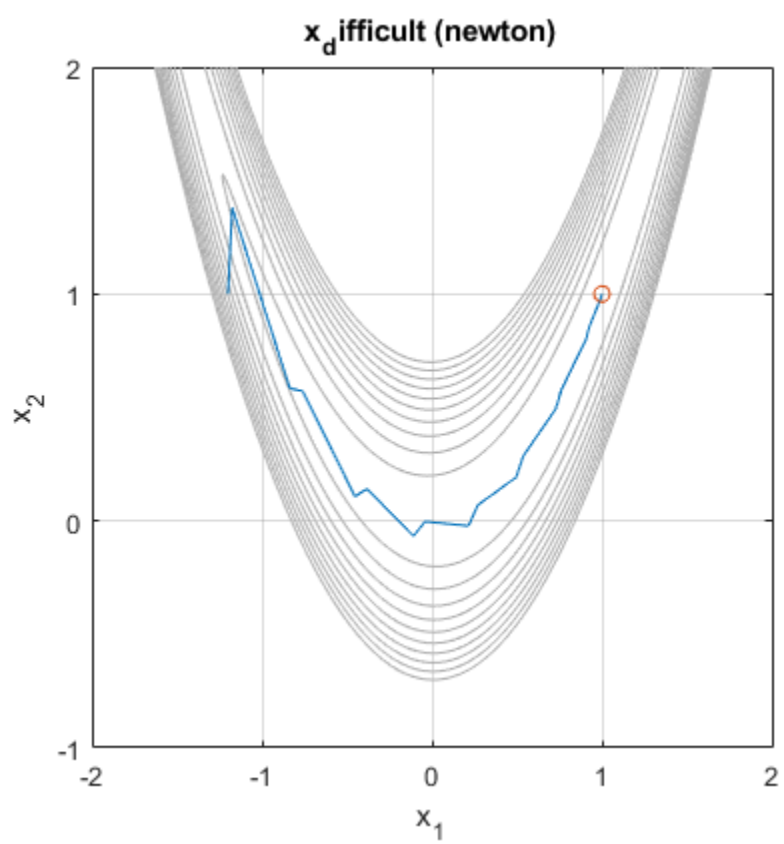
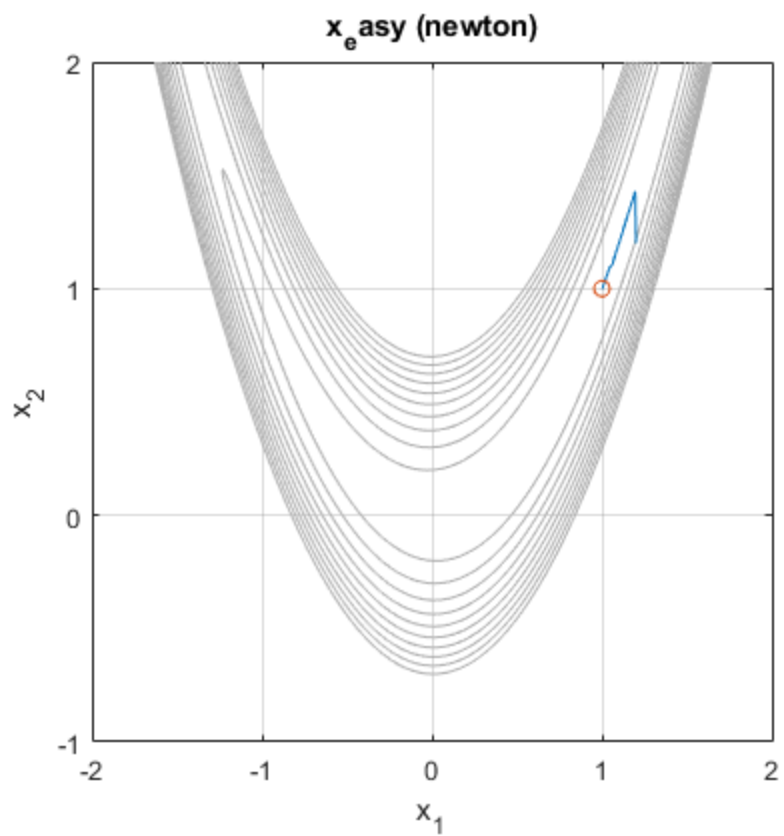
$x_k =$

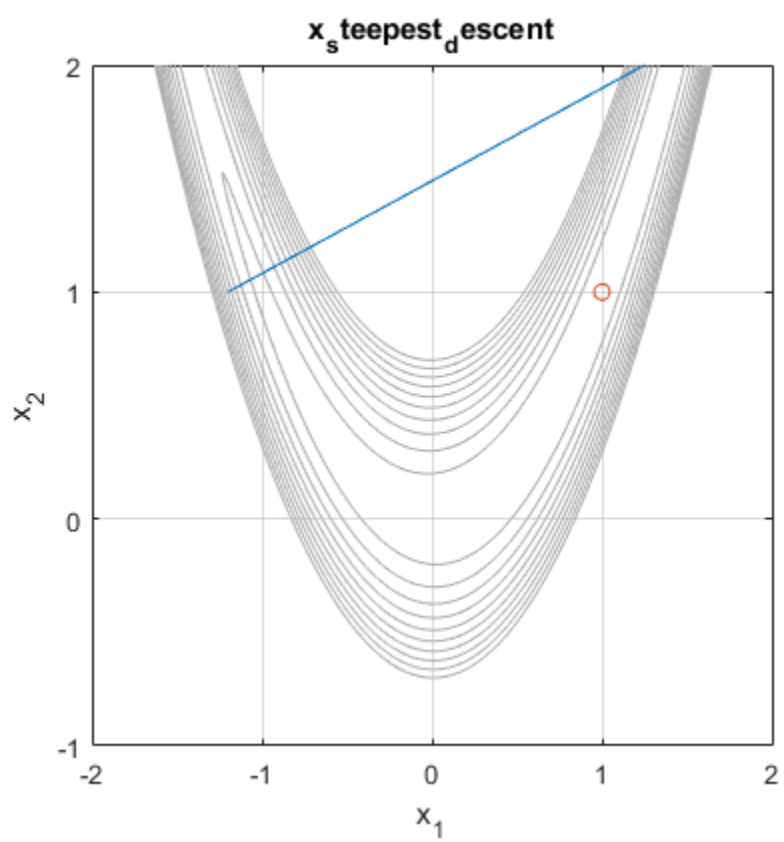
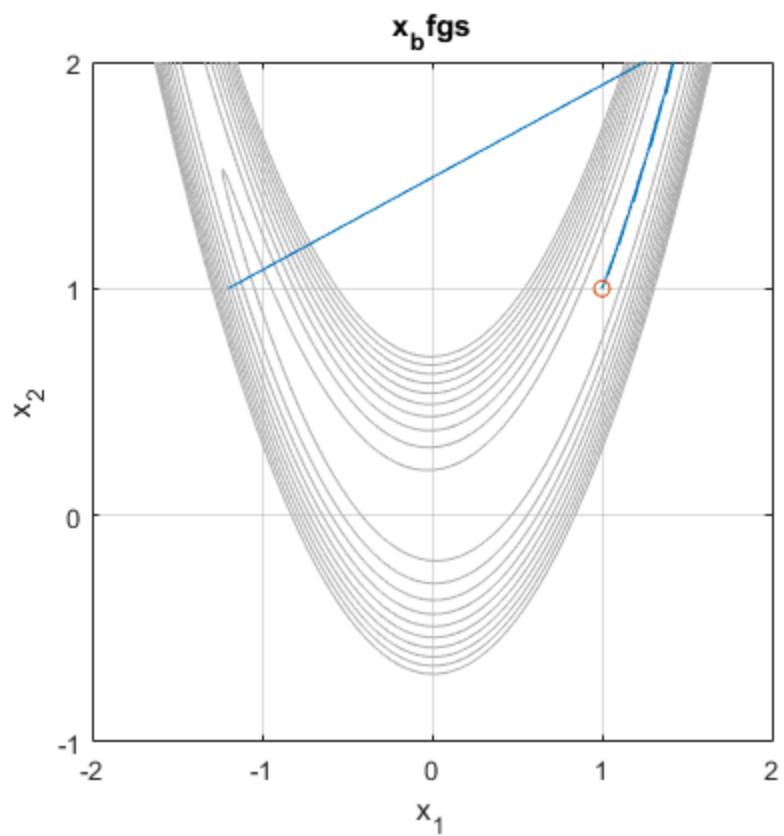
1.0000

1.0000

Warning: Ignoring extra legend entries.







Published with MATLAB® R2019b