最优化10

加兴华 18300290007

附件含问题分析及迭代函数代码

1

1. Suppose f is a general (may be nonconvex) continuous differentiable real valued function whose gradient is Lipschitz continuous with Lipschitz constant L. Show the convergence results (with convergence rate) with both exact minimization and Armijo rule line search for $\{\min_{i=0,\dots,k}\|\nabla f(x_i)\|\}_k$.

1、精确线搜索：

$$f(x_0)-f^* \ge f(x_0)-f(x_k) = \sum_{i=0}^{k} f(x_i)-f(x_{i+1}) \ge t(1-\frac{Lt}{2})\sum_{i=0}^{k}\|\nabla f(x_i)\|^2$$

$$\ge t(1-\frac{Lt}{2})(k+1)\min_i\|\nabla f(x_i)\|^2$$

记 $M = t(1-\frac{Lt}{2})$. $M_{max}=\frac{1}{2L}$ at $t=\frac{1}{L}$

$$\Rightarrow \min_i\|\nabla f(x_i)\|^2 \le \frac{f(x_0)-f^*}{(k+1)M}$$

收敛速率 $O(\sqrt{\frac{1}{k+1}})$

系数 $\frac{f(x_0)-f^*}{2L}$

A回溯线搜索：

同样有 $f(x_0)-f^* \ge M\cdot\min_i\|\nabla f(x_i)\|^2$. 但此时t基于$\alpha$和$\{t_k\}$获得：

$$f(x_0)-f^* \ge \sum_{0}^{k} f(x_i)-f(x_{i+1}) \ge \sum_{i=0}^{k}\alpha t_i\|\nabla f(x_i)\|^2 \ge \alpha\sum_{j=0}^{k}t_j\cdot\min_i\|\nabla f(x_i)\|^2.$$

$\Rightarrow M \le \alpha\cdot(k+1)\cdot\max_j\{t_j\} \Rightarrow M_{max}$ at $t_{min} =: \min\{\hat{t}, \frac{2(1-\alpha)\beta}{L}\}$. 注：$t_j=\hat{t}\cdot\beta^{s_j+1}$

收敛速度 $O(\sqrt{\frac{1}{k+1}})$

系数 $\frac{f(x_0)-f^*}{t_{min}(1-\frac{Lt_{min}}{2})}$

2

2. Show, using the definition, that the sequence $1+k^{-k}$ converges superlinearly to 1.

2. $\lim_{k\to\infty}[1+k^{-k} = 1+e^{-k\ln k}] = 1$

$\therefore e_k \triangleq k^{-k}$

$$\frac{e_{k+1}}{e_k} = \frac{(k+1)^{-k-1}}{k^{-k}} = \frac{1}{k+1}\cdot(1+\frac{1}{k})^{-k} \to 0 \quad \text{as} \begin{cases} \frac{1}{k+1}\to 0 \\ (1+\frac{1}{k})^{-k}\to e \end{cases}$$

$\therefore 1+k^{-k}$ 超线性收敛

3

3. Solve the following problems in MATLAB with gradient descent methods and damped Newton's methods, all with Armijo rule line search. Try two different initial points for each method.

(a)

(a) Solve the following problem

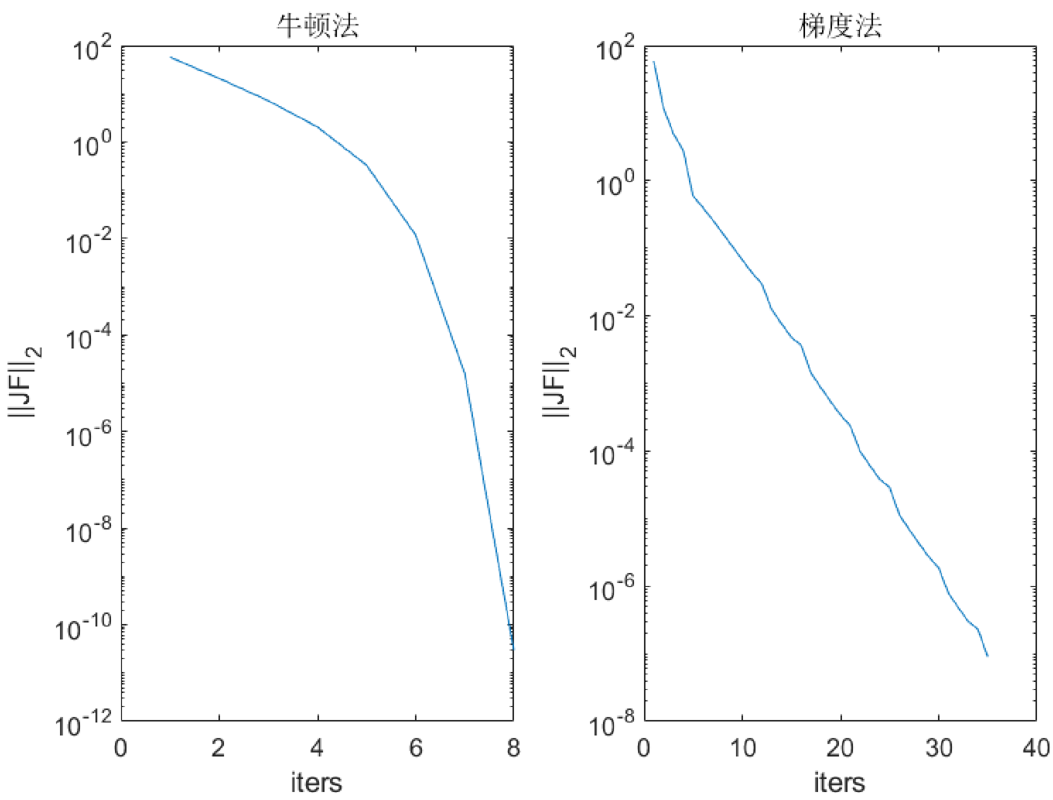
$$\min_{x_1,x_2} f(x_1,x_2) = \exp(x_1+3x_2-0.1)+\exp(x_1-3x_2-0.1)+\exp(-x_1-0.1)+0.1x^T x$$

You should set the stopping criterion as $\|\nabla f(x)\| \leq$ 1e-7. Plot figures to show the logarithm of the Euclidean norm of the gradient versus integration number. (You may use the semilogy function to plot figures.)

运行question3_1.m，结果如下:

```
opt_x_newton =
  -0.321455337609409    0.000000000002318
opt_solution_newton =
   2.570407447255078
opt_x_GD =
  -0.321455311311982    0.000000004382634
opt_solution_GD =
   2.570407447255080
```

3-1



(b)

(b) Solve the following logistic regression problem:

$$\min_{w\in\mathbb{R}^n, c\in\mathbb{R}} \frac{1}{m}\sum_{i=1}^{m} \log(1+\exp(-b_i(w^T a_i + c))) + 0.01(w^T w + c^2)$$

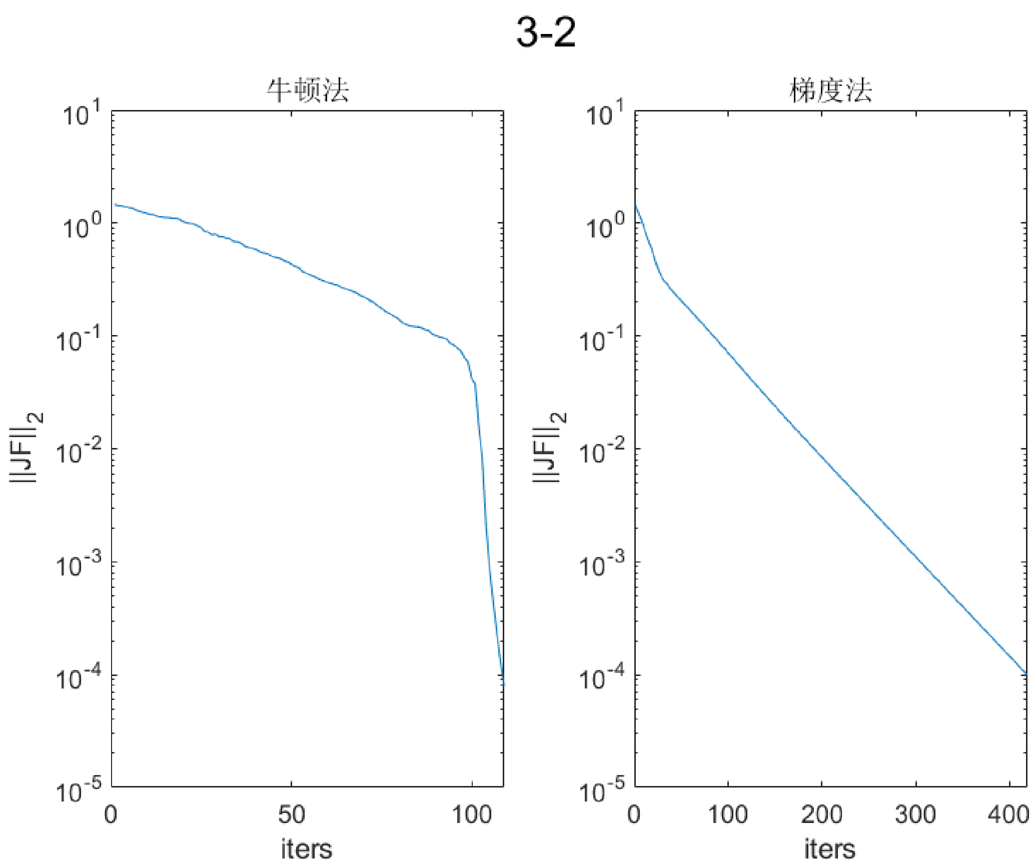
where a_i, b_i are given data. Use the following MATLAB code to generate the data:

```
m = 500; n = 1000;
A=[a1,...,am] = randn(n,m); b = sign(rand(m,1)-0.5);
```

Terminate your code when the Euclidean norm of the gradient is smaller than 10^{-4}. Plot figures to show the logarithm of the Euclidean norm of the gradient versus integration number.

运行question3_2.m,结果如下:

```
opt_solution_newton =
   0.117256405318718
opt_solution_GD =
   0.117256609404719
```



4

4. Use BFGS methods **with backtracking line search** and BB-step size gradient method **with backtracking line search** (you need to implement the algorithm with two updates of t_k as given in the course slides) method to solve

$$\min \frac{1}{2}x^T Ax + b^T x$$

for A and b generated by

```
rc=1:10:1000;A=sprandsym(100,0.1,rc);b=randn(100,1);
```

Use the all one vector (`ones(n,1)`) as the starting point. Terminate the problem after 1000 iterations or the norm of gradient is less than 1e-6. Note that the optimal value f^* can be computed analytically by the first order optimality condition. Compute it and plot the evolution for $\log(f(x^k) - f^*)$.

运行question4.m，结果如下:

```
opt_solution_BFGS =
  -0.353493187888508
opt_solution_BB =
  -0.353493187888497
```