https://github.com/hyeonjang/numerical-optimization
20211046
Hyeonjang An

Homework 5

EC6301
Numerical Opimization
December 1, 2021

Problem

1. Implement linear Conjugate Gradient method for following function:
   (a) $f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)^2$
2. Implement nonlinear Conjugate Gradient methods for the following functions:
   (a) $f(x, y) = 40(y - x^2)^2 + (1 - x)^2$
   (b) $f(x, y) = (1.5 - x + xy)^2 + (2.25 - x + xy^2)^2 + (2.625 - x + xy^3)^2$
3. Discuss their performances between nonlinear CGs:

Implementation

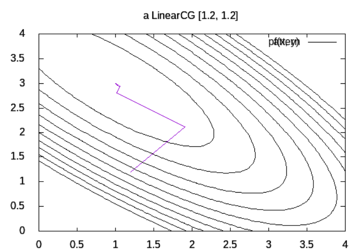
1. Implementation
   (a) **LinearCG**
      i. LinearCG is the method for linear function, which can be represented as $\frac{1}{2}x^T Ax - bx$
      ii. Therefore after deriving matrix A from the function, then we can apply this as input on a computing.
      iii. In the implementation, the constructor of LinearCG takes A and b as input argument
      iv. As termination condition, the residual r_k should be zero, when the convergence occrus. Because of a numerical calculation, it is not exact zero value, but is set as near zero, threshold $1e - 2$

(a) **NonlinearCG**

i. As similar as LinearCG, to terminate the criterion should be g_k is zero, but as same reason, currently set as $1e - 4$

Plotting

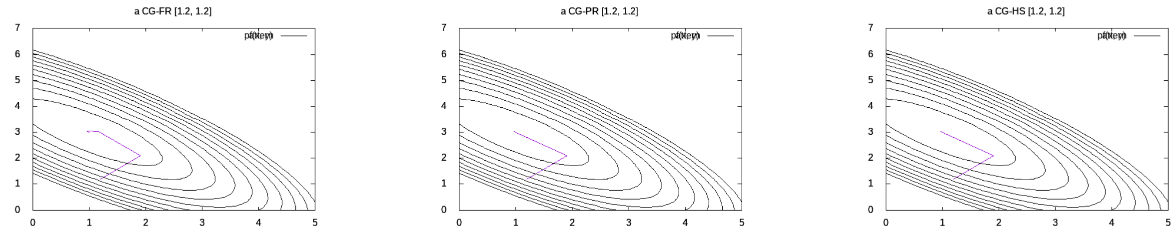
1. LinearCG behavior



(a) LinearCG

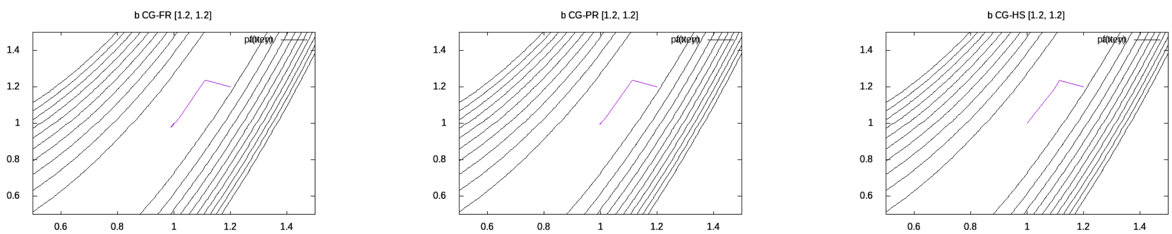
Figure 1: $f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)^2$

2. NonlinearCG behavior
   - According to Hager and Zhang [**hager2006algorithm**], in Fletcher-Reeves scheme, a jamming can occur, when the search direction nearly orthogonal to the gradient
   - CG-PR is the method to escape this problem has the fastest convergence speed
     (a) $\beta_k^{PR} = \frac{y_k^T g_{k+1}}{\|g_k\|^2}$, where $y_k = g_{k+1} - g_k$. When jamming occurs $g_{k+1} \approx g_k, \beta_k^{PR} \approx 0$, and $d_{k+1} \approx -g_{k+1}$
     (b) It means when jamming occurs, the search direction is not orthogonal to the gradient
   - This situation is observied apparently in *Figure2*, in the CG-FR cases, the plot show the moving purple line following the contour. But, not in the CG-PR and CG-HS cases.
   - Also, *Figure3* show this type of result, which in the case of CG-FR makes convergence failed.



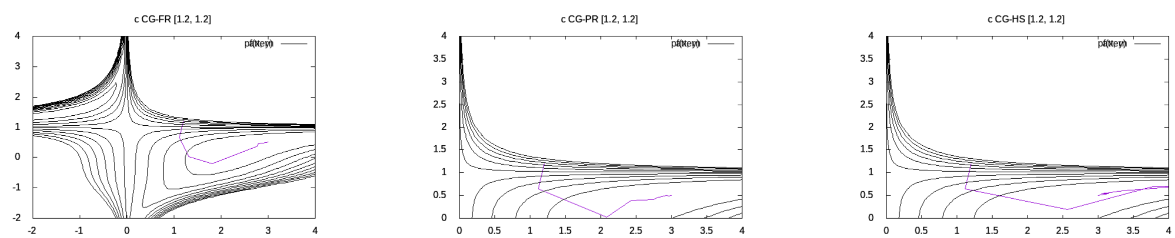
(a) CG-FR (b) CG-PR (c) CG-HS

Figure 2: $f(x, y) = (x + 2y - 7)^2 + (2x + y - 5)^2$



(a) CG-FR (b) CG-PR (c) CG-HS

Figure 3: $f(x, y) = 40(y - x^2)^2 + (1 - x)^2$



(a) CG-FR (b) CG-PR (c) CG-HS

Figure 4: $f(x, y) = (1.5 - x + xy)^2 + (2.25 - x + xy^2)^2 + (2.625 - x + xy^3)^2$

Performance

1. Convergence speed

   - As said before, CG-FR fails to converge, when the search direction is nearly orthogonal to gradient. And also, this affects Convergence speed
   - Especially, the function b case shows the failure case of convergence, which is

indicatied in *Figure4*

- Otherwise, CG-HS method has the fastest convergence speed

initial point	$f(x, y)$	Performance(x, y)		
		CG-FR	CG-PR	CG-HS
[1.2, 1.2]	(b)	62409445493 ns	320895182 ns	41624827 ns
	(c)	740809097 ns	349752676 ns	80612994 ns