Homework #5

Implement Conjugate gradient Methods & Disscusion

GIST AI

20231126 황산하

\*코드는 Python으로 작성했고 Report와 함께 Zip으로 압축하여 제출하였습니다.

INDEX

1. Comparison of Conjugate Gradient Methods With result

2. Discussion - terminal condition

3. Appendix A : Code

**1.** **Comparison and Result**

본 과제는 Conjugate Gradient (CG)의 최적화 방법을 구현하고 Non Linear 방법인 FR, PR 그리고 HS 의 성능을 비교하는 과제이다. Linear를 가정하고 Local Minimum을 찾는 Linear CG 방법과 Non Linear를 가정하고 Local Minimum을 찾는 Non Linear CG 각각의 특징 때문에 함수의 특징을 고려해서 알고리즘을 사용해야 하는데, 그래서 Linear CG 방법에서 실험한 함수와 Non Linear에서 사용한 함수가 달라서 직접적인 비교는 어렵다. 따라서 본 과제에서는 Non linear CG 방법들 사이에서의 비교를 하고 Linear CG는 결과만 보고서에 기록하려고 한다. Non linear CG를 비교할 땐 명확한 비교를 위해 초기점을 (0.5,0.5)로 통일했고, 동일한 종결조건을 사용했다.

**1) Linear Conjugate Gradient Method (Result Only)**

[표 1 Linear Conjugate 결과표]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Linear | (x, y) | Iter | Convergence | f(x,y) |
| F1 | 2.01631, 0.965561 | 99 | Yes | 0.00777987 |

초기 점은 (0,0)으로 잡고 알고리즘을 구현했을 때 얻을 수 있는 결과이다. 이전 과제 4에서 했던 gradient based 알고리즘들과 마찬가지로 step size가 성능에 영향을 끼치는 것 같았다. Step size를 search 하는 알고리즘으로는 wolfe를 적용한 line search 와 inexact line search를 비교했다. 둘다 잘 수렴하는 모습을 보였다.

**2) Non-Linear Conjugate Gradient Method : FR**

[표 2 FR결과표]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FR | (x, y) | Iter (Time) | Convergence | f(x,y) |
| F2 | 0.327, 0.60554 | 3 iter (0.003sec) | Yes | 0.231 |
| F3 | 2.50, 0.319 | 6 iter (0.0049sec) | Yes | 0.08 |

초기값의 영향을 많이 받음

**3) Non-Linear Conjugate Gradient Method : PR**

[표 3 PR 결과표]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PR | (x, y) | Iter (Time) | Convergence | f(x,y) |
| F2 | 0.281, 0.633 | 2 iter (0.00199sec) | Yes | 0.855 |
| F3 | 0.744,-1.4533 | 611iter (0.00399sec) | Yes | 9.74 |

초기값 영향큼, 알맞은 terminal condition 있다면 더 좋을 듯

**4) Non-Linear Conjugate Gradient Method : HS**

[표 4 HS 결과표]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HS | (x, y) | Iter (Time) | Convergence | f(x,y) |
| F2 | 0.39147, 0.71453 | 6 iter (0.004sec) | Yes | 0.79 |
| F3 | 1.6585, -0.6311 | 40 iter (0.0451sec) | Yes | 3.32 |

terminal condition 있다면 더

**2.** **Disccusion – Terminal Condition**

terminal condition 있다면 더

**3.** **Appendix A : Code**

코드는 Python으로 구현했으며 제출될 폴더 안에는 method.py, result.py, utils.py가 있다. 이전 과제와 마찬가지로 utils.py에는 알고리즘 구현을 위해 필요한 gradient를 구하는 함수와 inexact line search 함수가 구현되어있다. Method.py안에 각각의 CG알고리즘들이 구현되어 있다. Result.py를 실행 시킴으로써 결과를 받아 볼 수 있는데 스크립트에서 함수와 알고리즘을 선택해서 실행시켜야 한다. 예를 들어 python result.py --mode=cg\_fr --x0=[0.5,0.5] 이런식으로 실행 시킬 수 있다.linear\_conjugate, cg\_fr, cg\_pr, cg\_hs 중에서 선택해서 mode를 고를 수 있다.