Project 2: Unconstrained Optimization

Nilesh Kumar & Naureen Hoque
CISC820 - Quantitative Foundations
Rochester Institute of Technology (RIT)
{nk4856, nh9502}@rit.edu

Project Objective: In this project, we were given the three test objective functions. Our goal was to implement the following optimization algorithms and then apply each of these to the provided 3 test functions: 1) Gradient Descent Method (GD), 2) Newton Method, 3) Quasi-Newton Method (QN), and 4) Conjugate Gradient Method (CG). As we will see in the results that the convergence of Gradient Descent can be slow so we use Conjugate Gradient method to accelerate the convergence and at the same time avoid the computational costs needed in methods like Newton. Linear CG method makes use of conjugate vectors, this conjugacy of vectors can be used to show how linear CG takes at most n steps. We have implemented Fletcher-Reeves variation of the original CG method which is common adaption of CG on non-linear functions. It is also economical because it generates the new direction based on previous direction and makes sure that generated vector is conjugate to all other vectors used before.

Our Findings:

1) **Function 1:** $f(x) = \sum_{i=1}^{n}(i.x_i^2)$ where $n = 100$. Global minimum $f(x)_{min} = 0$.
We applied same initial parameter values to explore these four algorithms: $\rho = 0.5$, maximum number of iteration = 1000, tolerance = 0.000001 and $\alpha = 1$.

Method	Min Value of $f(x)$	No of Iterations
Gradient Descent	3.8502e-05	438
Newton	0	2 (1 for exit)
Quasi-Newton	4.9343e-15	116
Conjugate Gradient	2.9129e-04	132

TABLE I: Min Value of $f(x)$ and No of Iterations with $x = [100 : 100 : 10000]$

Method	Min Value of $f(x)$	No of Iterations
Gradient Descent	5.2525e-05	532
Newton	0	2 (1 for exit)
Quasi-Newton	7.2355e-17	117
Conjugate Gradient	7.7344e-05	251

TABLE II: Min Value of $f(x)$ and No of Iterations with $x = [1000 : 1000 : 100000]$

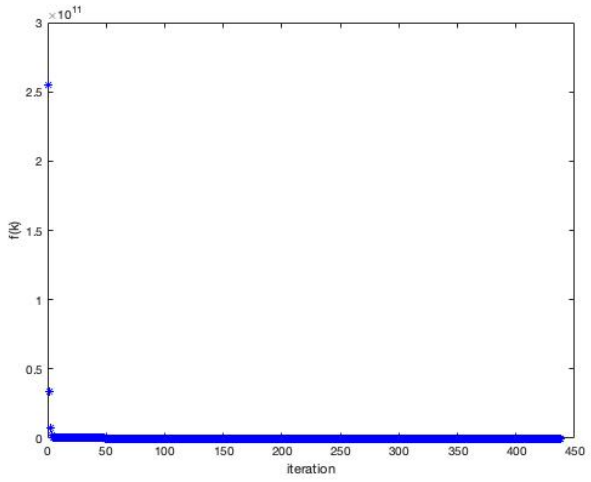


Fig. 1: Function 1: Gradient Descent

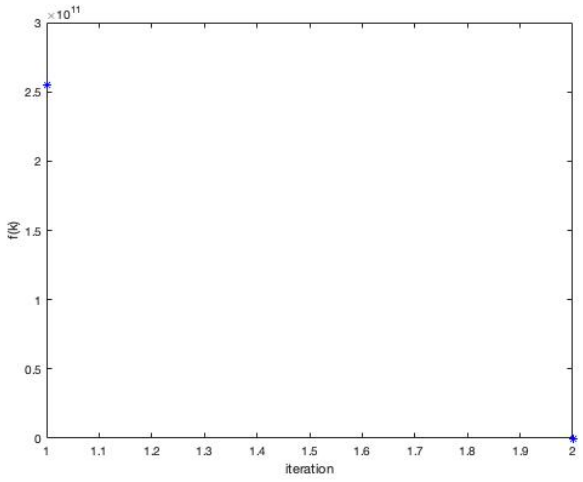


Fig. 2: Function 1: Newton

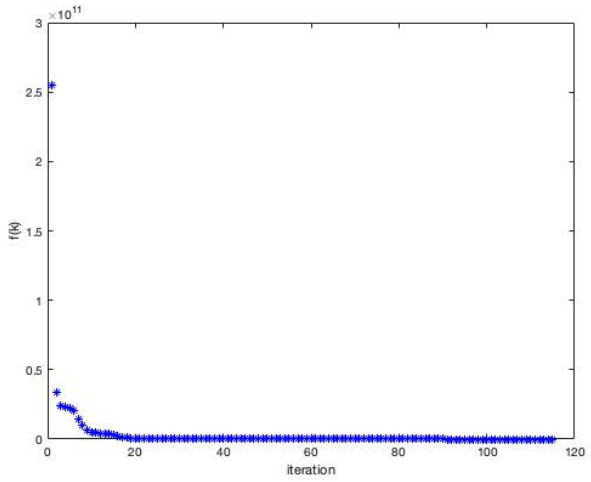


Fig. 3: Function 1: Quasi Newton

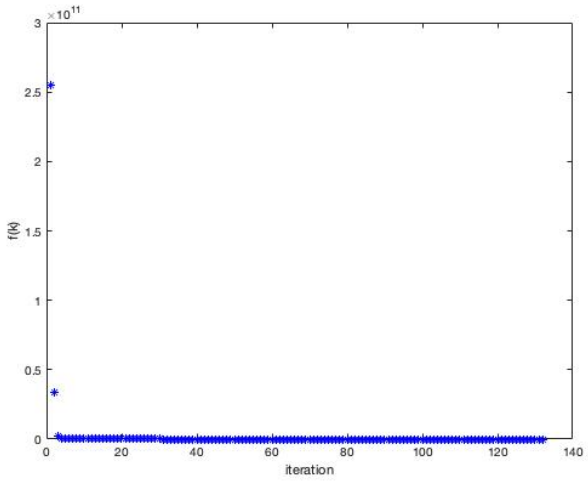


Fig. 4: Function 1: Conjugate Gradient

Analysis of the Algorithms on the Function: On function 1 (quadratic) all four methods perform according to their expectations. For Newton it converges in just one step, irrespective of the starting point, Quasi-Newton is the second fastest and is also robust to the initialization. Gradient Descent depends a lot on the initialization because of its zigzag behavior and alignment with minimum value when approaching a minimum. Gradient Descent's poor performance can also be attributed to the bad scaling of the function as the i increase the sensitivity of function with respect to that variable also increases. Conjugate gradient is also expected to lie somewhere in between Gradient Descent and Newton/Quasi-Newton as it accelerates the convergence rate of Gradient Descent but also avoids the high computational cost of Newton methods.

2) **Function 2:** $f(x) = c^T x - \sum_{i=1}^{m} log(b_i - a_i^T x)$ where $m = 500$ and $n = 100$.

We applied same initial parameter values to explore these four algorithms: $x = zeros(100, 1)$, $\rho = 0.5$, maximum number of iteration = 1000, $\alpha = 1$, and tolerance = 0.000001.

Method	Min Value of $f(x)$	No of Iterations
Gradient Descent	-2.4432e+03	1000
Newton	-2.4432e+03	10
Quasi-Newton	-2.4432e+03	182
Conjugate Gradient	-1.8463e+03	7
Conjugate Gradient (tighter convergence)	-2.4432+03	1000

TABLE III: Min Value of $f(x)$ and No of Iterations with α =1

Analysis of the Algorithms on the Function: This is a convex function, but not quadratic. It is also a barrier with an implicit constraint that Ax is greater than b. We try to solve this problem by using original method with a slightly modified backtrack and keep reducing alpha till we get alpha which does not take our x out of feasible region and we always start with a x in the feasible region. As GD takes zigzag path closer to minima (which never happens at CG), it takes more iterations than the CG. Our results supports the theory. NW and QN algorithms use the curvature information to move faster which allows them to reach the destination very quickly. We can see that CG couldn't reach the minimum value similar to the other 3 algorithms. Our assumption is that it is because of the learning rate. It might reach to the minimum (or close to the minimum) value if smaller learning rate was used.

3) **Function 3:** $f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$.

This is a variant of Rosenbrock function (non-convex) which is defined by:
$f(x,y) = \sum_{i=1}^{N-1} 100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2$, where $x = [x_1, x_2, ..., x_N] \in \mathbb{R}^N$.
This function has a global minimum at $(1, 1)$ for $N = 2$. Thus, for all four algorithms, we got exact same result with $x = [1; 1]$ (the initial value of x), as expected.
We applied same initial parameter values to explore these four algorithms: $\rho = 0.5$, maximum number of iteration = 1000, tolerance = 0.000001 and $\alpha = 1$.

α_0	$f(x)$	No of Iterations
1	0	1
0.1	0	1
0.01	0	1

TABLE IV: Initial $x = [1; 1]$ with Different α

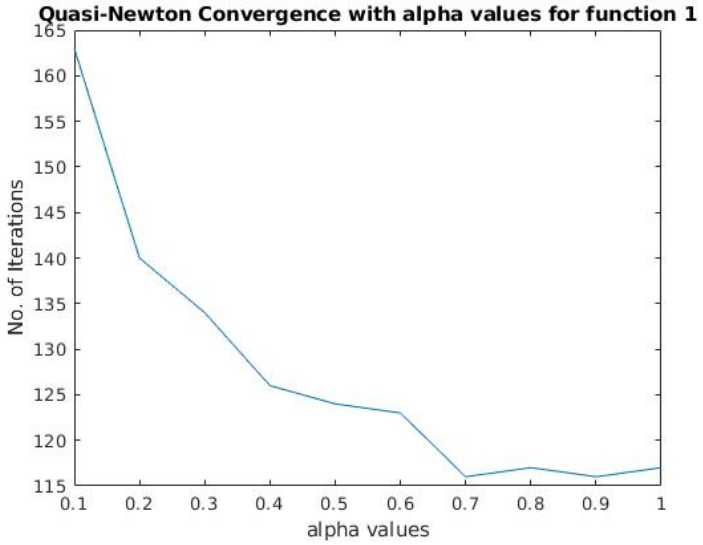
Method	Min Value of $f(x)$	No of Iterations
Gradient Descent	54.4009	1500
Newton	9.9512e-21	163
Quasi-Newton	3.6572e-11	61
Conjugate Gradient	4.2978e+05	4
Conjugate Gradient (tighter convergence)	3.0815e-29	1500

TABLE V: Min Value of $f(x)$ and No of Iterations with $x = [70, 70]$

Method	Min Value of $f(x)$	No of Iterations
Gradient Descent	0.0968	1000
Newton	2.7382e-13	10
Quasi-Newton	1.3262e-08	13
Conjugate Gradient	0.0034	20

TABLE VI: Min Value of $f(x)$ and No of Iterations with $x = [1.5; 2]$

Analysis of the Algorithms on the Function: This is a Non-convex function and the minima for this function lies in a narrow valley. It is easier to get to the valley but getting to the minima takes time. Gradient descent crawls its way to the minima as compared to other three methods. This behavior of Gradient Descent is attributed to the fact that it does not consider the curvature (other functions approximate quadratic) and taking the steepest descent direction every time can lead to a zigzag path.



Analysis of the Step-Size on the Methods: We explored different initial α values over the algorithms. We found that convergence of Quasi-Newton gets faster with higher value of initial and same behavior was observed for Newton on function three. But for Gradient Descent and Conjugate Gradient there

is no such trend which means that choosing alpha for these two methods is a hyperparameter and can be inferred using validation strategies.