Question - 1 analysis report

Jarpula Bhanu Prasad - AI21BTECH11015

***Abstract*—This report contains the analysis of unconstrained optimization problem using Gradient Descent method (with a fixed step size of 0.1) and Newton's method**

CODES

Jupyter notebook code → here

I. PROBLEM

Consider the following unconstrained optimization problem.

$$\min \quad \frac{1}{3}x_1^6 - 2.1x_1^4 + 4x_2^4 + 4(x_1^2 - x_2^2) + x_1x_2$$
$$\text{Sub.to} \quad \mathbf{x} \in \mathcal{R}^2 \tag{1}$$

Obtain the minima of the optimization problem using the Gradient Descent method (with a fixed step size of 0.1) and Newton's method. Use the initial point as $\mathbf{x}^0 = [1, -1]^T$

II. SOLUTION

A. Gradient Descent method

For the given optimization problem we defined epsilon $\epsilon = 10^{-6}$, step size $\alpha = 0.1$ and initial point $\mathbf{x}^0 = [1, -1]$

on running Gradient Descent algorithm we conclude that

- minima
  $\mathbf{x}^*$ = [0.08984201 -0.71265645]
- minimum value
  $f(\mathbf{x}^*)$ = -1.0316284534898608

fig(1) gives the visualization of function $f(\mathbf{x})$ and minimum value $f(\mathbf{x}^*)$

fig(2) gives the visualization of function $f(\mathbf{x})$ along with points after each iteration, alone points in normal view and points view from bottom view angle

from fig(1) we can see there are more than one minima (i.e., there exists local minimas) with our gradient descent algorithm we converge to one of minima's

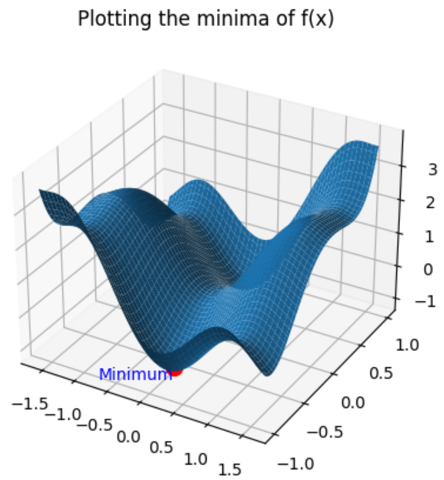

Figure 1. Visualization of $f(\mathbf{x}^*)$ for gradient descent

from fig(2) we see there is jitter in updation point (from bottom angle view) slowly the jitter reduces and converges to minima

B. Newton's method

For the given optimization problem we defined epsilon $\epsilon = 10^{-6}$ and initial point $\mathbf{x}^0 = [1, -1]$

on running Newton's algorithm we conclude that

- minima
  $\mathbf{x}^*$ = [1.10920534 -0.7682681]
- minimum value
  $f(\mathbf{x}^*)$ = 0.543718600978185

fig(3) gives the visualization of function $f(\mathbf{x})$ and minimum value $f(\mathbf{x}^*)$

fig(4) gives the visualization of function $f(\mathbf{x})$ along with points after each iteration, alone points in normal view and points view from bottom view angle

from fig(3) we can see there are more than one minima (i.e., there exists local minimas)

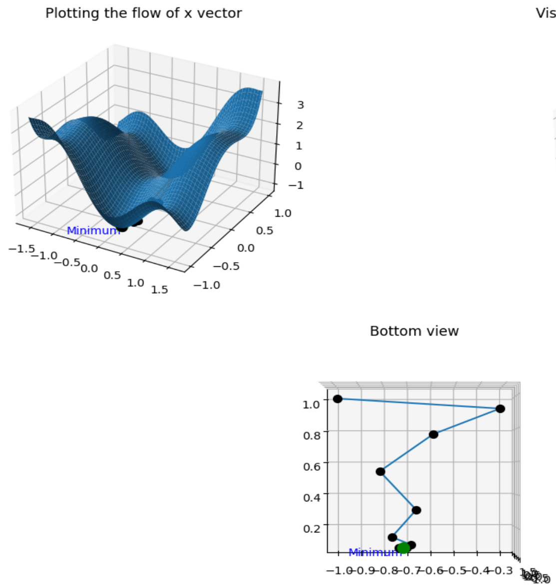

Figure 2. Visualization of $f(\mathbf{x})$ along with updated trial point after each iteration for gradient descent

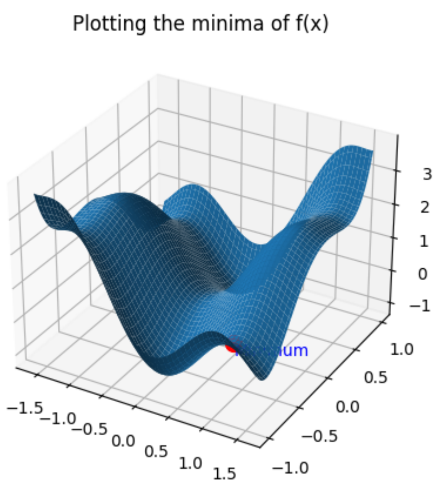

Figure 3. Visualization of $f(\mathbf{x}^*)$ using Newton method

with our gradient descent algorithm we converge to one of minima's

from fig(4) we see there is no jitter in updation point (from bottom angle view) but it rapidly converges to minima

III. Conclusion

From above we see even though we use same epsilon we see Gradient descent performs more efficient than Newtons method

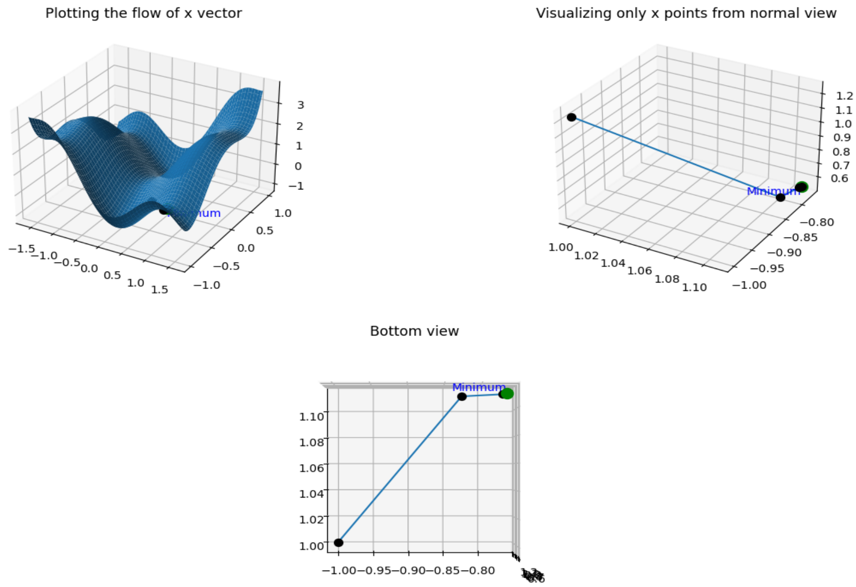

Figure 4. Visualization of $f(\mathbf{x})$ along with updated trial point after each iteration