E0 230: Assignment 1

Sayak Das
SR No: 24235
MTech AI

September 13, 2024

Question 1 : Convex and Coercive functions

1. Convex Functions

a. Testing Convexity

Convexity was tested using the following inequality,

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y) \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function if for each x, y and each $\alpha \in [0, 1]$ the inequality holds.

b. Strict Convexity and values of x^* and $f(x^*)$

`f1` is convex and `f2` is strictly convex. For strictly convex function in inequality in Eq.(1) is strong. We have tested convexity and strict convexity of `f1` and `f2` in `Python` using three `for` loops. First loop for x , second loop for y and third loop for α as in Eq.(1).

The function `f1` has $f(x^*) = 0.0064$ and $x^* \in [-0.31, 0.31]$. So, x^* is not unique for `f1.`

The function `f1` has a unique minima at $x^* = -2$ and $f(x^*) = 0.21$.

We have generated $f(x)$ where $x \in [-2, 2]$ using discrete sampling and used it to report the results.

2. Coercive Functions

a. Testing Coercivity

By definition of coercive functions, $\lim f(x) \to \infty$ as $||x|| \to \infty$. So, we have sampled x between $[-10000, 10000]$ with an increment of 10 and tested if the values, $f(x)$ follow the definition of coercive functions, i.e., as x increases in either positive or negative direction, $f(x)$ must increase.

`f3` is coercive.

b. Stationary points and Roots

We first explain how we evaluated roots and stationary points. We found roots by seeing at which x, sign of $f(x)$ changes. If $f(x_t)$ has negative sign and $f(x_{t+1})$ has positive sign, and vice-versa, then root is at $\frac{f(x_t)+f_(x_{t+1})}{2}$.

Now, for stationary points, we first calculate approximate value $f'(x)$ using the definition,

$$f'(x) = \frac{f(x+h) - f(x)}{h}$$

We round this value to 2 decimal places because the approximate value along with floating point errors will never be exactly 0. So, all stationary points has $f'(x) = 0$. We distinguish between the stationary points by using the approximate value of $f''(x)$,

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

If $f''(x) > 0$ then x is a minima, and if $f''(x) < 0$ then x is a maxima.

Roots of `f3`: -2.00, -0.40, 0.69 and 2.41. Minimas of `f3`: -1.42 and 1.79. Local Maximas of `f3`: 0.14.

The points are presented in Figure (1).

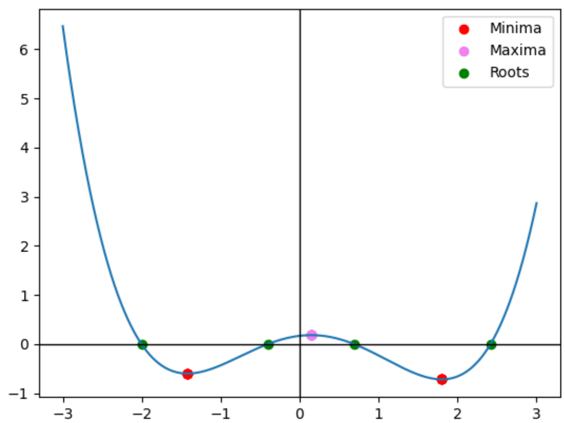


Figure 1: Stationary points and Roots of `f3`

Question 2 : Gradient Descent

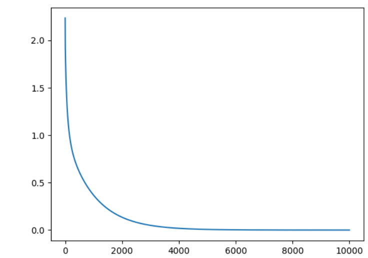
a. Gradient Descent with fixed step-size

We have run gradient descent with constant step size ($\alpha = 10^{-5}$) for 10000 iterations. After 10000 iterations, we get $\mathbf{x}^* = [-4.12 \times 10^{-5}, -5.00 \times 10^{-4}, -1.00 \times 10^{-3}, -2.00 \times 10^{-3}, -9.99 \times 10^{-3}]$, and $f(\mathbf{x}^*) \approx -0.006$. Different plots are presented in Figure (2).

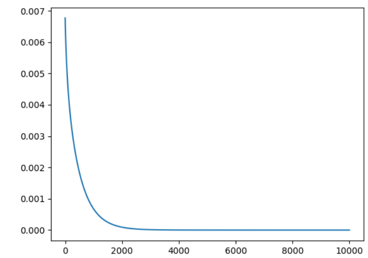
b. Gradient Descent with diminishing step-size

We have run gradient descent with decreasing step size such that $\alpha_k = \alpha_0/(k+1)$ and $\alpha_0 = 10^{-5}$ for 10000 iterations. After 10000 iterations, we get $\mathbf{x}^* = [-3.76 \times 10^{-5}, -8.90 \times 10^{-5}, -9.33 \times 10^{-5}, -9.55 \times 10^{-5}, -9.74 \times 10^{-5}]$, and $f(\mathbf{x}^*) \approx -0.0003$. Different plots are presented in Figure (3).

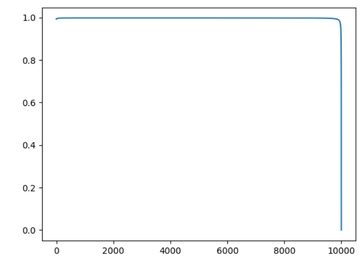
No the answer is not matching with the previous one. We got a better result using diminishing step-size. If our step-size is bigger than the interval we are trying to minimize, we can never get a x^t such that $f(x^t) \leq f(x^{t-1})$. This is why gradient descent with fixed step-size got a worse result than gradient descent with diminishing step-size.



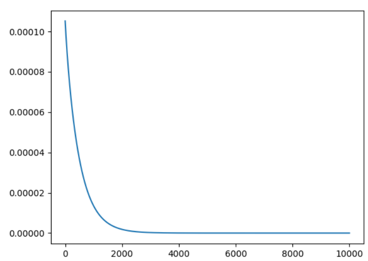
(a) $||\nabla f(\mathbf{x}_k)||_2$



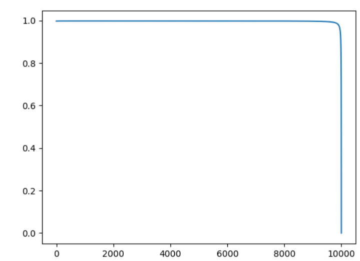
(b) $f(\mathbf{x}_k) - f(\mathbf{x}_T)$



(c) $f(\mathbf{x}_k) - f(\mathbf{x}_T)/f(\mathbf{x}_{k-1}) - f(\mathbf{x}_T)$

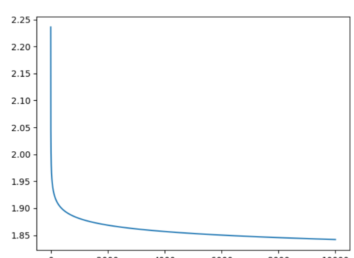


(d) $||\mathbf{x}_k - \mathbf{x}_T||_2^2$

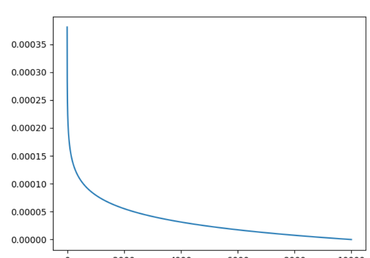


(e) $||\mathbf{x}_k - \mathbf{x}_T||_2^2/||\mathbf{x}_{k-1} - \mathbf{x}_T||_2^2$

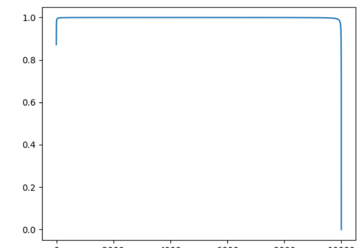
Figure 2: Constant Gradient Descent : Graph of different quantities at each iteration.



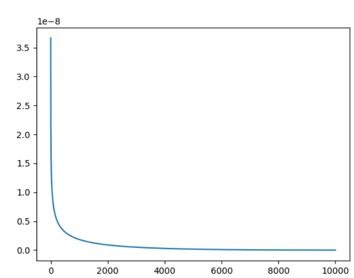
(a) $||\nabla f(\mathbf{x}_k)||_2$



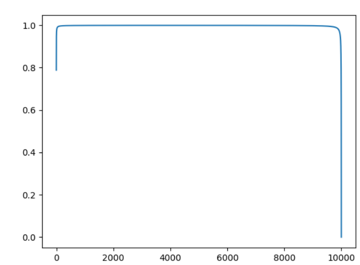
(b) $f(\mathbf{x}_k) - f(\mathbf{x}_T)$



(c) $f(\mathbf{x}_k) - f(\mathbf{x}_T)/f(\mathbf{x}_{k-1}) - f(\mathbf{x}_T)$

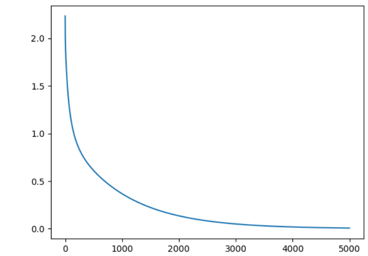


(d) $||\mathbf{x}_k - \mathbf{x}_T||_2^2$

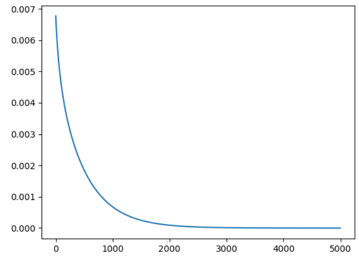


(e) $||\mathbf{x}_k - \mathbf{x}_T||_2^2/||\mathbf{x}_{k-1} - \mathbf{x}_T||_2^2$

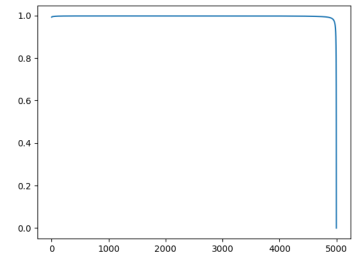
Figure 3: Diminishing Gradient Descent : Graph of different quantities at each iteration.



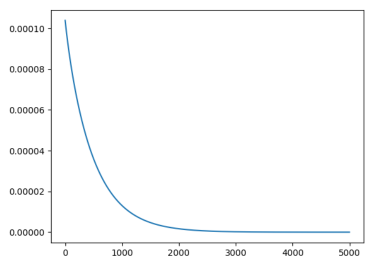
(a) $||\nabla f(\mathbf{x}_k)||_2$



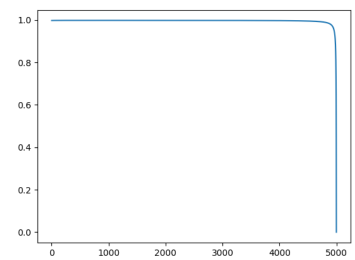
(b) $f(\mathbf{x}_k) - f(\mathbf{x}_T)$



(c) $f(\mathbf{x}_k) - f(\mathbf{x}_T)/f(\mathbf{x}_{k-1}) - f(\mathbf{x}_T)$



(d) $||\mathbf{x}_k - \mathbf{x}_T||_2^2$



(e) $||\mathbf{x}_k - \mathbf{x}_T||_2^2/||\mathbf{x}_{k-1} - \mathbf{x}_T||_2^2$

Figure 4: Inexact Line Search with Wolfe Conditions : Graph of different quantities at each iteration.

c. Inexact Line Search

We have run gradient descent with where step size is determined by Wolfe conditions for 5000 iterations. Where $c_1 = 10^{-4}$, $c_2 = 1 - c_1$ and $\gamma = 10^{-5}$. After 5000 iterations, we get $\mathbf{x}^* = [-4.12 \times 10^{-5}, -5.00 \times 10^{-4}, -1.00 \times 10^{-3}, -2.00 \times 10^{-3}, -9.93 \times 10^{-3}]$, and $f(\mathbf{x}^*) \approx -0.006$. Different plots are presented in Figure (4).

d. Exact Line Search

We have run gradient descent with exact step size for 100 iterations. After 100^{th} iterations, we get $\mathbf{x}^* = [-3.65 \times 10^{-5}, -5.00 \times 10^{-4}, -1.00 \times 10^{-3}, -1.99 \times 10^{-3}, -9.15 \times 10^{-3}]$, and $f(\mathbf{x}^*) \approx -0.006$. Different plots are presented in Figure (5).

We compute $\mathbf{A}$ as follows, in the denominator we need, $\mathbf{p_k^T A p_k}$ where $\mathbf{p_k} = -\nabla f(x_k)$. So, $\mathbf{p_k^T A p_k} = (-\nabla f(x_k))^T \mathbf{A}(-\nabla f(x_k))$.

We see that,

$$\nabla f(-\nabla f(x_k)) = \mathbf{A}(-\nabla f(x_k)) + b \tag{2}$$

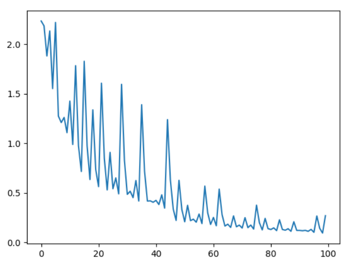
Let $-\nabla f(x_k) = 0$ and $\nabla f(x_k) = 0$ Then,

$$\mathbf{A}x_k + b = 0 \tag{3}$$

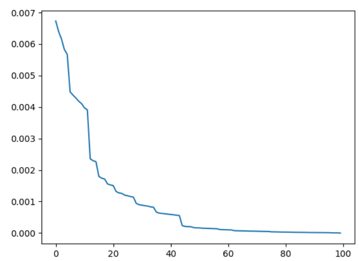
$$\mathbf{A}(-\nabla f(x_k)) + b = 0 \tag{4}$$

Adding Eq.(3) and Eq.(4) we get,

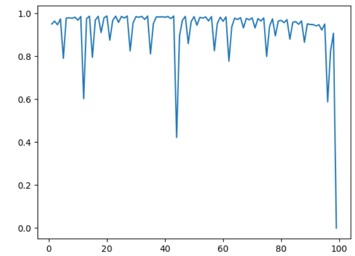
$$\begin{aligned} b &= -[\mathbf{A}(x_k - \nabla f(x_k) + b] \\ &= -\nabla f(x_k - \nabla f(x_k)) \end{aligned} \tag{5}$$



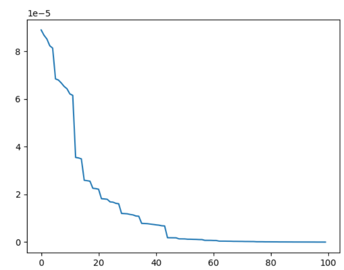
(a) $||\nabla f(\mathbf{x}_k)||_2$



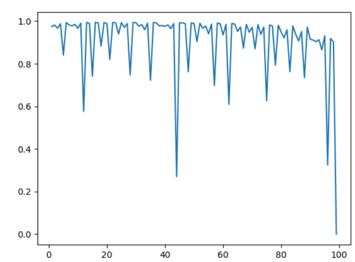
(b) $f(\mathbf{x}_k) - f(\mathbf{x}_T)$



(c) $f(\mathbf{x}_k) - f(\mathbf{x}_T)/f(\mathbf{x}_{k-1}) - f(\mathbf{x}_T)$



(d) $||\mathbf{x}_k - \mathbf{x}_T||_2^2$



(e) $||\mathbf{x}_k - \mathbf{x}_T||_2^2/||\mathbf{x}_{k-1} - \mathbf{x}_T||_2^2$

Figure 5: Exact line search : Graph of different quantities at each iteration.

From Eq.(2),

$$\mathbf{A}(-\nabla f(x_k)) = \nabla f(-\nabla f(x_k)) + \nabla f(x_k - \nabla f(x_k)) \tag{6}$$

Hence, without knowing explicitly what $\mathbf{A}$ is, we calculate the denominator as,

$$\mathbf{p_k^T A p_k} = (-\nabla f(x_k))^T \cdot [\nabla f(-\nabla f(x_k)) + \nabla f(x_k - \nabla f(x_k))]$$

Question 3 : Perturbed Gradient Descent

1. Finding Stationary points

We have to find and categorize all stationary points of the function,

$$f(x, y) = e^{xy}$$

. We know at stationary points the gradient of the function is 0. So,

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\frac{\partial e^{xy}}{\partial x} = ye^{xy} \quad \text{and} \quad \frac{\partial e^{xy}}{\partial y} = xe^{xy}$$

Now,

$$ye^{xy} = 0 \quad \text{So, either } y = 0 \text{ or } e^{xy} = 0.$$
$$\Rightarrow y = 0 \quad [\text{Because, } e^{xy} \text{ can not be zero.}]$$

Replacing $y = 0$ in the other partial derivative we get,

$$xe^{x \cdot 0} = 0 \quad \implies x = 0$$

So, we have a stationary point at $(x, y) = (0, 0)$. To find if $(0, 0)$ is a minima, maxima or stationary point, we have to check definiteness of the Hessian matrix of f.

$$\nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} = \begin{bmatrix} y^2 e^{xy} & e^{xy} + xye^{xy} \\ e^{xy} + xye^{xy} & x^2 e^{xy} \end{bmatrix}$$

at $x = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ we have,

$$\nabla^2 f(\begin{bmatrix} 0 \\ 0 \end{bmatrix}) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

The eigenvalues of $\nabla^2 f(\begin{bmatrix} 0 \\ 0 \end{bmatrix})$ are,

$$\mathbf{det}(\nabla^2 f(\begin{bmatrix} 0 \\ 0 \end{bmatrix}) - \lambda I) = \mathbf{det}(\begin{bmatrix} 0 - \lambda & 1 \\ 1 & 0 - \lambda \end{bmatrix}) = 0$$

$$\Rightarrow \lambda^2 - 1 = 0 \Rightarrow \lambda = \pm 1$$

Because the eigenvalues of $\nabla^2 f(x)$ at $x = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ is both positive and negative, $\nabla^2 f(x)$ is indefinite at that point. Which means $(x, y) = (0, 0)$ is a saddle point. $\square$

2. Gradient Descent stays on the line $x = y$

The update rule for gradient descent is as follows,

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \cdot \nabla f(\mathbf{x}^{(k)})$$

Where $\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}$. We know from *Question 3.1* that,

$$\nabla f(\mathbf{x}^{(k)}) = \begin{bmatrix} y^{(k)} e^{x^{(k)} y^{(k)}} \\ x^{(k)} e^{x^{(k)} y^{(k)}} \end{bmatrix} \tag{7}$$

Now we prove using induction, it is given that $x^{(0)} = y^{(0)}$. From (7) we create $x^{(1)}$ and $y^{(1)}$ as follows,

$$\begin{aligned} x^{(1)} &= x^{(0)} - \alpha_0 \cdot y^{(0)} e^{x^{(0)} y^{(0)}} \\ &= x^{(0)} - \alpha_0 \cdot x^{(0)} e^{x^{(0)} x^{(0)}} \\ &= x^{(0)} - \alpha_0 \cdot x^{(0)} e^{(x^{(0)})^2} \end{aligned}$$

$$\begin{aligned} y^{(1)} &= y^{(0)} - \alpha_0 \cdot x^{(0)} e^{x^{(0)} y^{(0)}} \\ &= x^{(0)} - \alpha_0 \cdot x^{(0)} e^{x^{(0)} x^{(0)}} \\ &= x^{(0)} - \alpha_0 \cdot x^{(0)} e^{(x^{(0)})^2} \end{aligned}$$

We see that $x^{(1)} = y^{(1)}$. At each step the relation between x and y holds, i.e., $x^{(k)} = y^{(k)}$. Hence, starting with $x = y$ implies that new points will lie on the line $x = y$. $\square$

3. Contour plot

The contour plot of $f(x, y) = e^{xy}$ is shown in Figure 6.

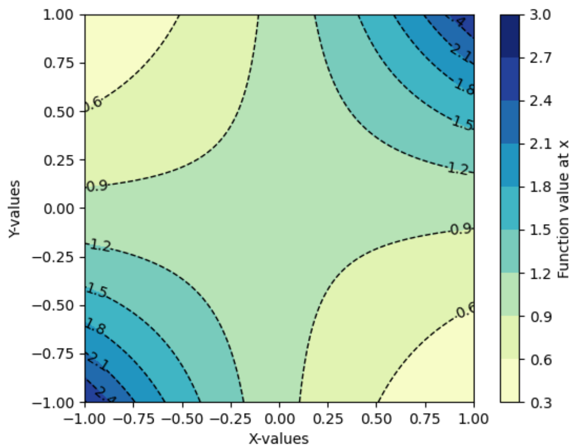


Figure 6: Contour plot of $f(x, y) = e^{xy}$ with $-1 \leq x, y \leq 1$.

4. Gradient Descent with fixed step-size

After running gradient descent with constant step size $\alpha = 10^{-4}$ with a starting point $\mathbf{x} = (x, y) = (1.2, 1.2)$ for 100000 iterations, we get $\mathbf{x}^* = (3.21 * 10^{-5}, 3.21 * 10^{-5})$ and $f(\mathbf{x}^*) = 1$. In Figure (7a) we present function value at each iteration and in Figure (7b) we present the trajectory of gradient descent. From Figure (7b) it is clear that gradient descent indeed follows the straight line $x = y$ when starting at point $x = y$.

5. Gradient Descent with decreasing step-sizes

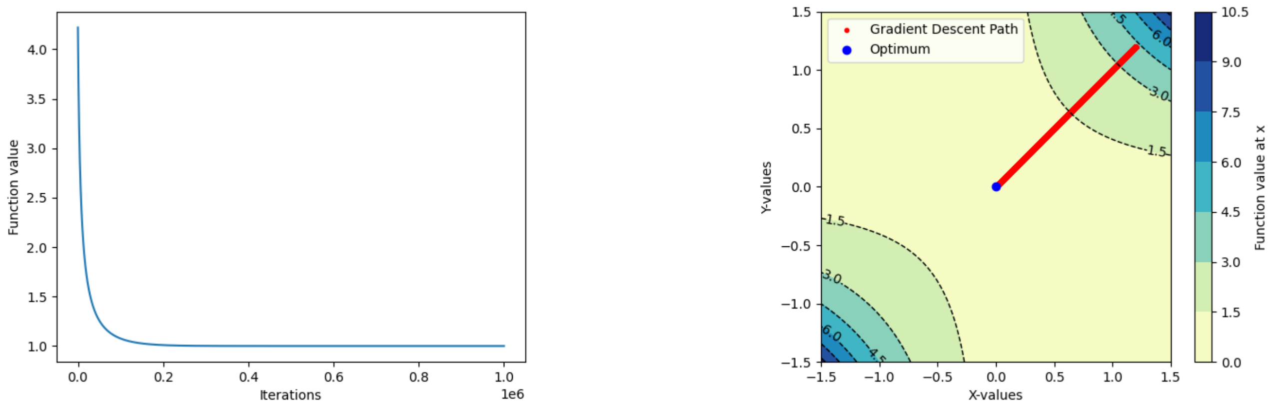
After running gradient descent with decreasing step-sizes $\alpha = 10^{-3}$ with a starting point $\mathbf{x} = (x, y) = (1.2, 1.2)$ for 100000 iterations, we get $\mathbf{x}^* = (1.144, 1.144)$ and $f(\mathbf{x}^*) = 3.701$. In Figure (8a) we present function value at each iteration and in Figure (8b) we present the trajectory of gradient descent.

6. Perturbed Gradient Descent with fixed step-size and noise-variance

After running gradient descent with constant step size $\alpha = 10^{-3}$ and constant noise-variance ($\zeta^{(t)} \sim \mathcal{N}(0, \sigma_t^2, \mathcal{I})$ where $\sigma^2 = 1$) with a starting point $\mathbf{x} = (x, y) = (1.2, 1.2)$ for 100000 iterations, we get $\mathbf{x}^* = (-2.677, 2.461)$ and $f(\mathbf{x}^*) = 0.001$. In Figure (9a) we present function value at each iteration and in Figure (9b) we present the trajectory of gradient descent.

7. Perturbed Gradient Descent with fixed step-size and decreasing noise-variance

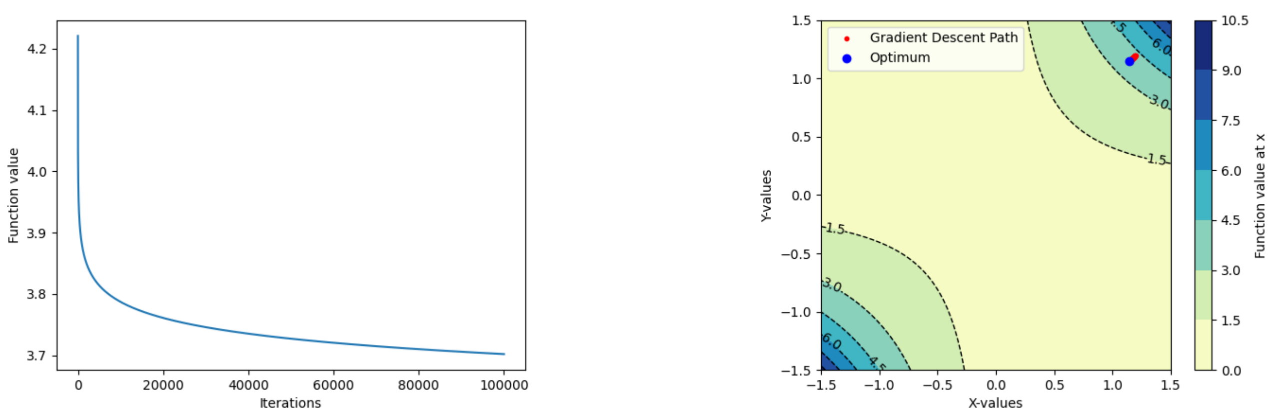
After running gradient descent with constant step size $\alpha = 10^{-3}$ and decreasing noise-variance ($\zeta^{(t)} \sim \mathcal{N}(0, \sigma_t^2, \mathcal{I})$ where $\sigma^2 = 1$) with a starting point $\mathbf{x} = (x, y) = (1.2, 1.2)$ for 100000 iterations, we get $\mathbf{x}^* = (2.641, -2.638)$ and $f(\mathbf{x}^*) = 0.0009$. In Figure (10a) we present function value at each iteration and in Figure (10b) we present the trajectory of gradient descent.



(a) Function value at each iteration

(b) Trajectory of gradient descent.

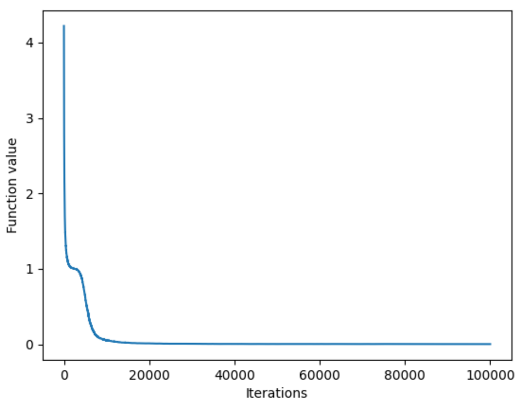
Figure 7: Graphs for gradient descent with constant step size.



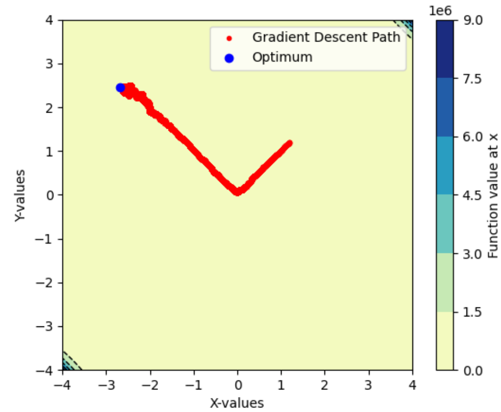
(a) Function value at each iteration

(b) Trajectory of gradient descent.

Figure 8: Graphs for gradient descent with constant step size.

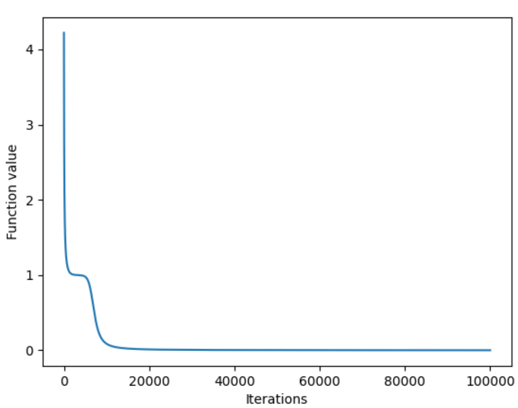


(a) Expected function value at each iteration

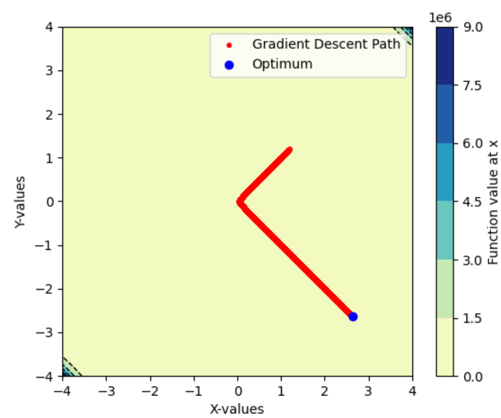


(b) Trajectory of gradient descent.

Figure 9: Graphs for gradient descent with fixed step size and noise variance.

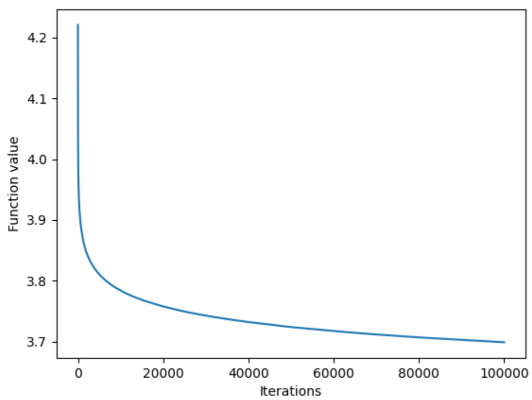


(a) Expected function value at each iteration

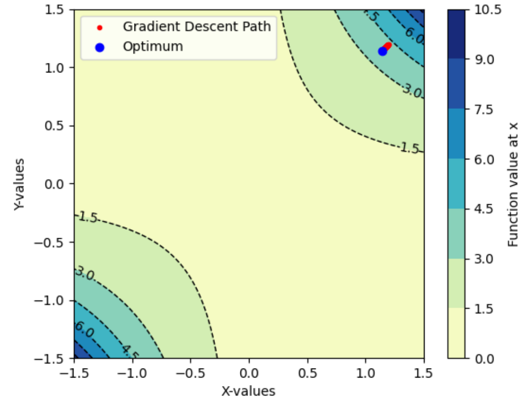


(b) Trajectory of gradient descent.

Figure 10: Graphs for gradient descent with fixed step size and decreasing noise variance.

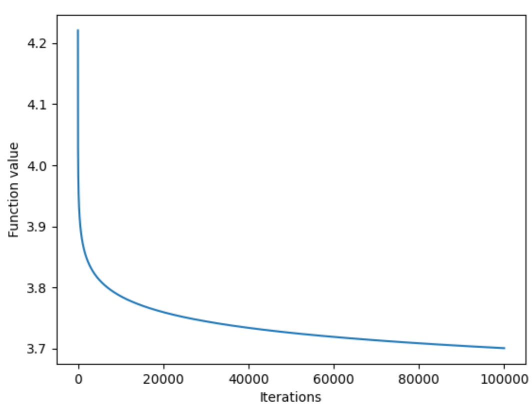


(a) Expected function value at each iteration

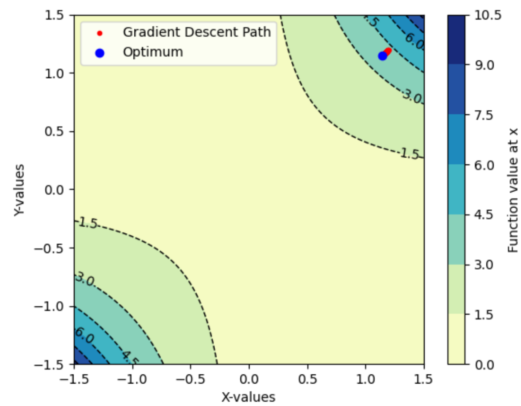


(b) Trajectory of gradient descent.

Figure 11: Graphs for gradient descent with decreasing step size and constant noise variance.



(a) Expected function value at each iteration



(b) Trajectory of gradient descent.

Figure 12: Graphs for gradient descent with decreasing step size and decreasing noise variance.

6. Perturbed Gradient Descent with decreasing step-size and fixed noise-variance

After running gradient descent with decreasing step size $\alpha = 10^{-3}$ and constant noise-variance ($\zeta^{(t)} \sim \mathcal{N}(0, \sigma_t^2, \mathcal{I})$ where $\sigma^2 = 1$) with a starting point $\mathbf{x} = (x, y) = (1.2, 1.2)$ for 100000 iterations, we get $\mathbf{x}^* = (1.143, 1.143)$ and $f(\mathbf{x}^*) = 3.6989$. In Figure (11a) we present function value at each iteration and in Figure (11b) we present the trajectory of gradient descent.

6. Perturbed Gradient Descent with decreasing step-size and noise-variance

After running gradient descent with decreasing step size $\alpha = 10^{-3}$ and decreasing noise-variance ($\zeta^{(t)} \sim \mathcal{N}(0, \sigma_t^2, \mathcal{I})$ where $\sigma^2 = 1$) with a starting point $\mathbf{x} = (x, y) = (1.2, 1.2)$ for 100000 iterations, we get $\mathbf{x}^* = (1.143, 1.144)$ and $f(\mathbf{x}^*) = 3.700$. In Figure (12a) we present function value at each iteration and in Figure (12b) we present the trajectory of gradient descent.

Question 4 : Zeroth-order Optimisation

1. Find all the extremas

Find all the extremas of $f(x) = x(x-1)(x-3)(x+2)$ *where* $x \in \mathbb{R}$.

First we simplify the functions $f(x)$,

$$\begin{aligned} f(x) &= x(x-1)(x-3)(x+2) \\ &= (x^2)(x^2 - x - 6) \\ &= x^4 - x^3 - 6x^2 - x^3 + x^2 + 6x \\ &= x^4 - 2x^3 - 5x^2 + 6x \end{aligned}$$

So, $f'(x)$ is,

$$f'(x) = 4x^3 - 6x^2 - 10x + 6$$

All stationary points has $f'(x) = 0$. Because, this is a cubic equation, we use rational root theorem to guess the first root. We check at $x = \frac{1}{2}$.

$$\begin{aligned} f'(\frac{1}{2}) &= 4 \cdot \frac{1}{8} - 6 \cdot \frac{1}{4} - 10 \cdot \frac{1}{2} + 6 \\ &= \frac{1}{2} - \frac{3}{2} - 5 + 6 \\ &= \frac{1}{2} - \frac{3}{2} + 1 \\ &= \frac{1 - 3 + 2}{2} \\ &= 0 \end{aligned}$$

So, $x = \frac{1}{2}$ is root of $f'(x)$. Now we find the other roots by dividing $(x - \frac{1}{2})$ from $f'(x)$,

$$\begin{array}{r|l} & 4x^2 - 4x - 12 \\ \hline x - \frac{1}{2} & 4x^3 - 6x^2 - 10x + 6 \\ & \underline{4x^3 - 2x^2} \\ & -4x^2 - 10x + 6 \\ & \underline{-4x^2 + 2x + 6} \\ & -12x + 6 \\ & \underline{-12x + 6} \\ & 0 \end{array}$$

So, $\frac{f'(x)}{(x - \frac{1}{2})} = 4x^2 - 4x - 12$. Now we can find root of this equation easily,

$$\begin{aligned} 4x^2 - 4x - 12 &= 0 \\ x^2 - x - 3 &= 0 \end{aligned}$$

$$x = \frac{1 \pm \sqrt{1 - 4 \cdot (-3)}}{2} = \frac{1 \pm \sqrt{13}}{2}$$

Hence, $f(x)$ has stationary point at $x = \frac{1}{2}, \frac{1+\sqrt{13}}{2}, \frac{1-\sqrt{13}}{2}$. $\square$

2.a. Golden Section Search

Number of iterations needed

In Golden Section search, the length of the interval gets reduced by a constant amount ρ in each iteration. We use this fact to obtain number of iterations needed to achieve a precision of $\epsilon = 10^{-4}$.

We know the length of the interval at the beginning is $(b-a)$ and $\rho = (\Phi - 1)$ where $\Phi = \frac{1+\sqrt{5}}{2}$ is the golden ratio. We see next how the length of the interval reduces in each iteration,

$$\begin{array}{cc} \text{Iteration: } 1 \Leftrightarrow & (b-a) \\ \text{Iteration: } 2 \Leftrightarrow & (b-a)\cdot\rho \\ \text{Iteration: } 3 \Leftrightarrow & (b-a)\cdot\rho^2 \\ \vdots & \vdots \\ \text{Iteration: } N \Leftrightarrow & (b-a)\cdot\rho^{N-1} \end{array} \tag{8}$$

We stop when the value $(b-a)\cdot\rho^{N-1}$ is less than ϵ. Say we reach this precision at iteration N. Then,

$$\begin{aligned} (b-a)\cdot\rho^{N-1} &< \epsilon \\ \rho^{N-1} &< \frac{\epsilon}{b-a} \\ (N-1)\cdot\ln\rho &< \ln\left(\frac{\epsilon}{b-a}\right) \\ N-1 &> \frac{\ln\left(\frac{\epsilon}{b-a}\right)}{\ln\rho} \\ N &> \frac{\ln\left(\frac{\epsilon}{b-a}\right)}{\ln\rho} + 1 \end{aligned}$$

In our case,

$$\begin{aligned} \ln\left(\frac{\epsilon}{b-a}\right) &= \ln\left(\frac{10^{-4}}{3-1}\right) \approx -9.9034 \\ \ln\rho &= \ln\Phi - 1 \approx -0.4812 \\ N &> \frac{-9.9034}{-0.4812} + 1 \approx 21.58 \end{aligned}$$

Therefore, we need at least 22 iterations to get a precision of 10^{-4}. $\square$

Tables and Graphs for Golden Section Search

The intervals at each iteration is presented in Table 1. The algorithm converges in 20 steps. Which is 2 less than what we predicted. We assume this error is generated because of floating-point arithmetic precision.

Different plots for golden search method is presented in Figure (13).

Iterations	a	b	Interval Length
1	1	3	2
2	1.764	3	1.23607
3	1.764	2.528	0.76393
4	2.056	2.528	0.47214
5	2.236	2.528	0.2918
6	2.236	2.416	0.18034
7	2.236	2.348	0.11146
8	2.279	2.348	0.06888
9	2.279	2.321	0.04257
10	2.295	2.321	0.02631
11	2.295	2.311	0.01626
12	2.295	2.305	0.01005
13	2.299	2.305	0.00621
14	2.301	2.305	0.00384
15	2.301	2.303	0.00237
16	2.302	2.303	0.00147
17	2.303	2.303	0.00091
18	2.303	2.303	0.00056
19	2.303	2.303	0.00035
20	2.303	2.303	0.00021

Table 1: Interval lengths and boundaries at each iteration for the Golden Search Method.

2.b. Fibonacci Search

Number of iterations needed

In Fibonacci search, the final interval after T iterations is given by $\frac{b-a}{F_{T+1}}$. So, this must be less then 10^{-4}. Hence we get the following inequality,

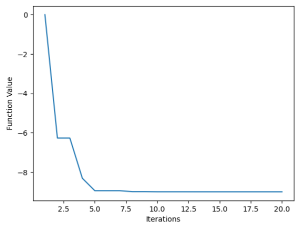
$$\begin{aligned}\frac{b-a}{F_{T+1}} &< 10^{-4}\\ F_{T+1} &> \frac{b-a}{10^{-4}}\\ &= \frac{2}{10^{-4}}\\ &= 20000\end{aligned}$$

We see that $F_{23} = 28657 > 20000$ is the first Fibonacci number. So we need at least 22 iterations to get a precision less that 10^{-4}. □

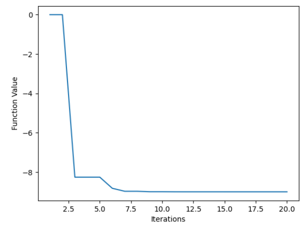
Tables and Graphs for Fibonacci Search

The intervals at each iteration is presented in Table 2. The same argument holds on why Fibonacci search converges in 20 iterations instead of 22 as predicted.

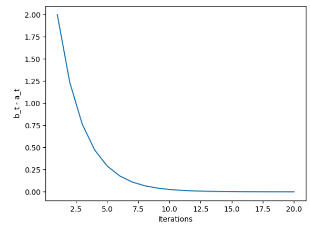
Different plots for Fibonacci search method is presented in Figure (14).



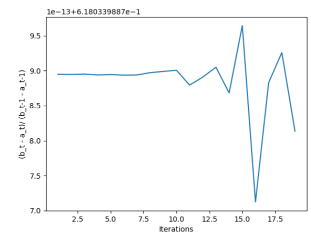
(a) $f(a)$ at each iteration.



(b) $f(b)$ at each iteration.

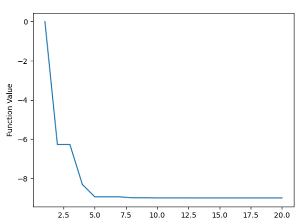


(c) $(b_t - a_t)$ at each iteration.

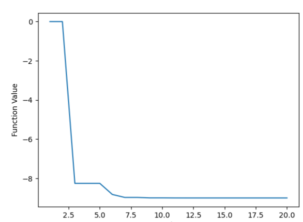


(d) $(b_t - a_t)/(b_{t-1} - a_{t-1})$ at each iteration.

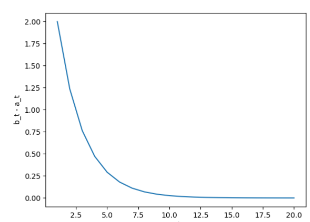
Figure 13: Graphs for Golden Search Method.



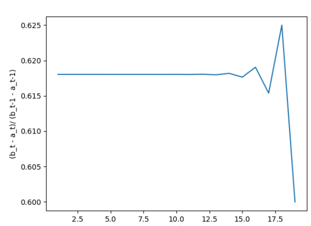
(a) $f(a)$ at each iteration.



(b) $f(b)$ at each iteration.



(c) $(b_t - a_t)$ at each iteration.



(d) $(b_t - a_t)/(b_{t-1} - a_{t-1})$ at each iteration.

Figure 14: Graphs for Fibonacci Search Method.

Iterations	a	b	Interval Length
1	1	3	2
2	1.764	3	1.23607
3	1.764	2.528	0.76393
4	2.056	2.528	0.47214
5	2.236	2.528	0.2918
6	2.236	2.416	0.18034
7	2.236	2.348	0.11146
8	2.279	2.348	0.06888
9	2.279	2.321	0.04257
10	2.295	2.321	0.02631
11	2.295	2.311	0.01626
12	2.295	2.305	0.01005
13	2.299	2.305	0.00621
14	2.301	2.305	0.00384
15	2.301	2.303	0.00237
16	2.302	2.303	0.00147
17	2.303	2.303	0.00091
18	2.303	2.303	0.00056
19	2.303	2.303	0.00035
20	2.303	2.303	0.00021

Table 2: Interval lengths and boundaries at each iteration for the Fibonacci Search Method.