Numerical Methods (CS 357)

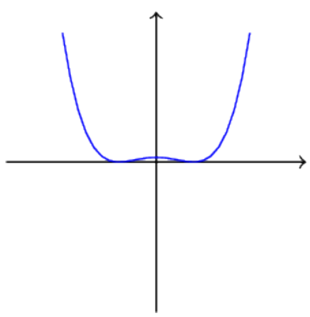
Worksheet

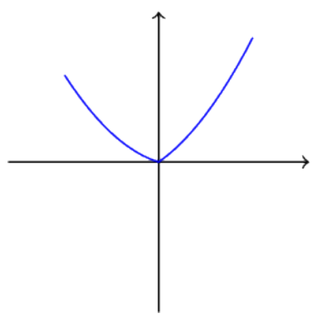
Part 1. Objectives

- Be able to find a minimum of a function in 1D using Golden Section Search and Newton
- Be able to find a minimum of a function in n dimensions using Steepest Descent and Newton
- Know the convergence rates of these methods and their behavior on different types of functions
- Be able to apply Lagrange multipliers for constrained optimization

Part 2. Unimodality

Which of these functions is unimodal?

A: 

B: 

(A) Function A

(B) Function B

(C) Function A and B

(D) Neither

Part 3. Entries of the Hessian matrix

Consider the function

$$f(x, y, z) = x^2 + 3xy + 5z^2.$$

What is the entry $H_{2,1}$ in the second row and first column of the Hessian matrix of f?

Part 4. Golden Section Search

Complete the code doing Golden Section Search for function minimization below. Stop the iteration when your brackets are less than 10^{-5} wide.

The supplied plotting code shows the evolution of your brackets. **Observe** how one of the bracket midpoints stays the same from one iteration to the next.

INPUT:

- `f`, a function to minimize. *Note:* `f` is not unimodal.
- `a` and `b`, the left and right ends of the starting bracket.

OUTPUT:

- `a`, `b`, the final ends of your bracket.

```
import numpy as np
brackets = []

while ....
    gs = (np.sqrt(5)-1)/2
    m2 = a + gs*(b-a)
    m1 = a + (1-gs)*(b-a)

    brackets.append([a, m1, m2, b])

    ...



# plotting code below, no need to modify
import matplotlib.pyplot as pt
x = np.linspace(-10, 10)
pt.plot(x, f(x))

brackets = np.array(brackets)
for i in range(4):
    pt.plot(brackets[:, i], 3*np.arange(len(brackets)), "o-")
```