Numerical Methods (CS 357)

Worksheet

Part 1. Objectives

- Be able to apply the Newton and Secant methods in one dimension
- Be able to apply the Newton method in multiple dimension
- Understand how quickly these converge and how their convergence speed compares to the bisection method
- Understand necessary and sufficient conditions for optimality
- Be able to apply Newton's method for optimization

Part 2. Optimization: Sufficient and necessary conditions

Consider the problem of minimizing the following function in one dimension. At the marked point, which of the conditions for a local minimum are satisfied?

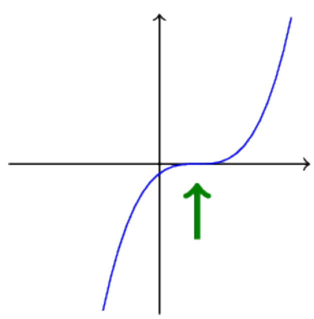

Figure 1: Function plot

(A) None of them

(B) Necessary

(C) Sufficient

(D) Necessary and Sufficient

Part 3. Optimization: Sufficient and necessary conditions

Consider the problem of minimizing the following function in one dimension. At the marked point, which of the conditions for a local minimum are satisfied?

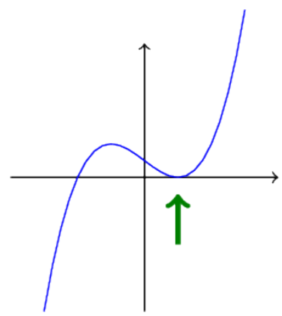

Figure 2: Function plot

(A) None of them

(B) Necessary

(C) Sufficient

(D) Necessary and Sufficient

Part 4. Error prediction

Let's say we have an iterative process whose error e_k at the kth iteration decreases following this estimate:

$$e_{k+1} \leq e_k^2.$$

Also suppose that the initial error e_0 is 0.1.

How many iteration will it take for the error to decrease to less than 10^{-10}?

Part 5. Secant method

Implement the secant method on the provided function `f`. Stop if your last two x values differ by less than 10^{-13}.

Print your x values as you go along.

INPUT:

- `f`, a function for which you should find a root
- `x0` and `x1`, two starting values

OUTPUT:

- `zero`, your approximation to the zero of the function
- A plot of `f` with your zero marked (already produced by the provided plotting code)

```
while ...




zero = ...

# plotting code below, no need to modify
import matplotlib.pyplot as pt
import numpy as np

plot_x = np.linspace(0, 3)
pt.plot(plot_x, f(plot_x))
pt.plot(zero, f(zero), "or")
```