Finite Difference Approximations

Introduction

Finite differences are a way to approximate derivatives using discrete points instead of exact calculus. They form the backbone of numerical methods for solving differential equations, simulating physical systems, and engineering computations. The idea is simple: replace derivatives with ratios of function values at nearby points.

Key Concepts

- First Derivative Approximation
- Second Derivative Approximation
- Example
- Generalization

First Derivative Approximations

So we want $f'(x)$, and we know $f(x)$ at discrete points.

Forward Difference

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

This is simply just the definition of the derivative. Instead of taking the limit, we just take tiny steps of h computationally.

Another way we can do this:

Backward Difference

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

This is more precise than forward/backward for the same h.

Central Difference

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

Second Derivative Approximation

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

This is the standard stencil for discretizing second derivatives (e.g. in heat transfer, diffusion problems).

Example

Let $f(x) = e^x$ at $x = 0$, with $h = 0.1$. Exact derivative: $f'(0) = 1$.

- Forward: $\frac{f(0.1)-f(0)}{0.1} = \frac{1.10517-1}{0.1} \approx 1.0517$
- Backward: $\frac{f(0)-f(-0.1)}{0.1} = \frac{1-0.90484}{0.1} \approx 0.9516$
- Central: $\frac{f(0.1)-f(-0.1)}{0.2} = \frac{1.10517-0.90484}{0.2} \approx 1.0016$

Central difference nails it much better (error ~ 0.0016).

Generalization

Higher-order formulas can be built by expanding Taylor series further. For example:

$$f'(x) \approx \frac{-f(x+2h) + 8f(x+h) - 8f(x-h) + f(x-2h)}{12h} + O(h^4)$$

This is a **fourth-order** central difference scheme.

Recap

- Finite differences approximate derivatives using discrete values.
- Forward/backward differences are first-order accurate.
- Central differences are second-order accurate and generally better.
- Step size h is a tradeoff: smaller h reduces truncation error but increases floating-point roundoff error.

Finite differences are the backbone of numerical PDE solvers (heat equation, Navier–Stokes, etc.), so mastering them is essential.