Interpolation

Differentiation

Integration

"Quadrature"

Convergence

$$\tilde{f}(x) = a_1 \varphi_1(x) + \dots + a_n \varphi_n(x)$$

$$f'(x) \approx \tilde{f}'(x) = a_1 \varphi_1'(x) + \dots + a_n \varphi_n'(x)$$

$$f'(x) = \left[\varphi_1'(x), \dots, \varphi_n'(x)\right] \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}$$

$$V' = \begin{array}{c} \\ x_1 \\ \vdots \\ x_n \end{array} \begin{array}{c} \begin{array}{ccc} \varphi_1' & \quad\quad & \varphi_n' \end{array} \\ \begin{bmatrix} \varphi_1'(x_1) & \cdots & \varphi_n'(x_1) \\ \vdots & & \\ \varphi_1(x_n) & & \end{bmatrix} \end{array}$$

Given points $f(\vec{x})$

would like $f'(\vec{x})$

$$\tilde{f}'(\vec{x}) = V' V^{-1} f(\vec{x})$$

About Differentiation Matrices

How could you find coefficients of the derivative in the original basis (φ_i)?

$V^{-1} \underbrace{V' V^{-1} f(x)}$ → coeffs

Give a matrix that finds the second derivative.

$V' V^{-1} V' V^{-1}$ $V' V^{-1}$ → differentiation matrix

↑ x-coordinates + $(\varphi)_i$, $(\varphi')_i$

$y' = V' V^{-1} y$

↑ approximation of $f'(x)$ at each x

degree n-1 polynomial at n-points

$$V^{-1} \; V' \; V^{-1} y$$

coeffs of f' in orig. basis

coefficients of f' in diff. basis

"

coefficient of f in orig. basis

values of f' at each point $x_1 \ldots x_n$

Demo: Taking derivatives with Vandermonde matrices

Finite Difference Formulas

It is possible to use the process above to find 'canned' formulas for taking derivatives. Suppose we use three points equispaced points $(x-h, x, x+h)$ for interpolation (i.e. a degree-2 polynomial).

- What is the resulting differentiation matrix?
- What does it tell us for the middle point?

$$V = \begin{bmatrix} 1 & x-h & (x-h)^2 \\ 1 & x & x^2 \\ 1 & x+h & (x+h)^2 \end{bmatrix} \qquad V' = \begin{bmatrix} 0 & 1 & 2(x-h) \\ 0 & 1 & 2x \\ 0 & 1 & 2(x+h) \end{bmatrix}$$

$$D = V'V^{-1} = \begin{bmatrix} & \cdots & \\ \frac{-1}{2h} & 0 & \frac{1}{2h} \\ & \cdots & \end{bmatrix}$$

$$D \cdot V = V'$$

$$\underbrace{\begin{bmatrix} \frac{-1}{2h} & 0 & \frac{1}{2h} \end{bmatrix}} \begin{bmatrix} 1 & x-h & (x-h)^2 \\ 1 & x & x^2 \\ 1 & x+h & (x+h)^2 \end{bmatrix}$$

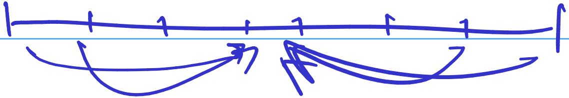
$$= \begin{bmatrix} 0 & 1 & 2x \end{bmatrix}$$

$$\frac{2xh}{2h} + \frac{2xh}{2h}$$

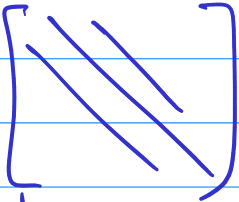
$$-\frac{\cancel{x}-h}{2h} + \frac{\cancel{x}+h}{2h} \qquad -\frac{(x-h)^2}{2h} + \frac{(x+h)^2}{2h}$$

$$D \cdot \begin{bmatrix} f(x-h) \\ f(x) \\ f(x+h) \end{bmatrix} = \begin{bmatrix} \cdots \\ \dfrac{f(x+h) - f(x-h)}{2h} \\ \cdots \end{bmatrix}$$

$$O(h^2)$$



$$D_{\text{tridiagonal}} \begin{bmatrix} \ddots & & & & \\ -\frac{1}{2h} & 0 & \frac{1}{2h} & & \\ & -\frac{1}{2h} & 0 & \frac{1}{2h} & \\ & & -\frac{1}{2h} & 0 & \frac{1}{2h} \\ & & & & \ddots \end{bmatrix} =$$



$$f(x) - \frac{df}{dx}(x) = b$$

Can we use a similar process to compute (approximate) integrals of a function f?

$$\tilde{f}(x) = a_1 \varphi_1(x) + \dots a_2 \varphi_2(x)$$

$$\int_a^b f(x)\,dx \approx \int_a^b \tilde{f}(x)\,dx = a_1 \int_a^b \varphi_1(x)\,dx + \dots + a_n \underbrace{\int_a^b \varphi_n(x)\,dx}$$

loss of accuracy

$$w = \begin{bmatrix} \int_a^b \varphi_1(x)\,dx \\ \vdots \\ \int_a^b \varphi_n(x)\,dx \end{bmatrix}$$

vector of weights

$Va = y$

$w^T \cdot a$ coeffs

Example: Building a Quadrature Rule

Demo: Computing the Weights in Simpson's Rule

Suppose we know

$$f(x_0) = 2 \qquad f(x_1) = 0 \qquad f(x_2) = 3$$

$$x_0 = \cancel{1}\,0 \qquad x_1 = \frac{1}{2} \qquad x_2 = 1$$

How can we find an approximate integral? $\int_0^1 f(x)\,dx$

$$w = \begin{bmatrix} \int_0^1 1\,dx \\ \vdots \\ \int_0 x^2\,dx \end{bmatrix} = \begin{bmatrix} 1 \\ 1/2 \\ 1/3 \end{bmatrix}$$

$$V = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1/2 & 1/4 \\ 1 & 1 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} x_0^0 & x_0^1 & x_0^2 \\ x_1^0 & x_1^1 & x_1^2 \\ x_2^0 & x_2^1 & x_2^2 \end{bmatrix}$$

$$a = V^{-1} \begin{bmatrix} 2 \\ 0 \\ 3 \end{bmatrix} = \begin{bmatrix} 2 \\ \cdot \\ \cdot \\ \cdot \end{bmatrix}$$

$$\text{integral} = w^T \cdot a \qquad (x, y)$$

$$= \underbrace{w^T V^{-1}}_{\text{weights}(x)} y$$

Facts about Quadrature

What does Simpson's rule look like on $[0, 1/2]$?

What does Simpson's rule look like on $[5, 6]$?

How accurate is Simpson's rule with n points and functions?

Accuracy with n points/function

Interpolation error $O(h^{n+1})$

Integration error $O(h^{n+2})$

Differentiation error $O(h^{n})$

Outline

What is linear convergence? quadratic convergence?

About Convergence Rates

Demo: Rates of Convergence

Characterize linear, quadratic convergence in terms of the 'number of accurate digits'.

Outline

Solving Nonlinear Equations

What is the goal here?

Bisection Method

Assume there is a zero on the interval $[a, b]$ and that f is continuous, perform binary search.

Demo: Bisection Method

What's the rate of convergence? What's the constant?

Newton's Method

Derive Newton's method.

Demo: Newton's method
Demo: Convergence of Newton's Method

What are some **drawbacks** of Newton?

Secant Method

What would Newton without the use of the derivative look like?

Demo: Secant Method

In-class activity: Nonlinear equations in 1D

Outline

Solving Nonlinear Equations

What is the goal here?

Newton's method

What does Newton's method look like in n dimensions?

Newton: Example

Set up Newton's method to find a root of

$$f(x, y) = \begin{pmatrix} x + 2y - 2 \\ x^2 + 4y^2 - 4 \end{pmatrix}.$$

Demo: Newton's method in n dimensions

Secant in n dimensions?

What would the secant method look like in n dimensions?

Outline

Optimization

State the problem.

Optimization: What could go wrong?

What are some potential problems in optimization?

Optimization: What is a solution?

How can we tell that we have a (at least local) minimum? (Remember calculus!)

Newton's Method

Let's steal the idea from Newton's method for equation solving:
Build a simple version of f and minimize that.

Demo: Newton's method in 1D

In-class activity: Optimization Methods

Golden Section Search

Would like a method like bisection, but for optimization.
In general: No invariant that can be preserved.
Need *extra assumption.*

Demo: Golden Section Search Proportions

Outline

Optimization in n dimensions: What is a solution?

How can we tell that we have a (at least local) minimum? (Remember calculus!)

Steepest Descent

Given a scalar function $f : \mathbb{R}^n \to \mathbb{R}$ at a point $\boldsymbol{x}$, which way is down?

Demo: Steepest Descent

Newton's method (nD)

What does Newton's method look like in n dimensions?

Demo: Newton's method in n dimensions

Demo: Nelder-Mead Method

Nonlinear Least Squares/Gauss-Newton

What if the f to be minimized is actually a 2-norm?

$$f(\boldsymbol{x}) = \|\boldsymbol{r}(\boldsymbol{x})\|_2, \qquad \boldsymbol{r}(\boldsymbol{x}) = \boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x})$$

Demo: Gauss-Newton