CS368 MATLAB Programming

Lecture 13

Young Wu

Based on lecture slides by Michael O'Neill and Beck Hasti

April 27, 2022

Algebraic Equations

Math

- An algebraic equation, also called a polynomial equation, are ones in the form,

$$\sum_{i=0}^{n} a_i x^i = 0.$$

- Root finding is the process of numerically finding one or all x's that satisfy the above equation.
- There are in general n solutions or roots (possibly complex or repeated) to the above equation.

Non-linear Equations

Math

- In general, non-linear equations in the form $f(x) = 0$ are solved using iterative methods.
- Start with a random guess x_0, and compute a sequence $x_1, x_2, \ldots$ with the property that $x^\star = \lim_{n\to\infty} x_i$ satisfies $f(x^\star) = 0$.

Optimization

Math

- Optimization problems are often solved by finding the roots to the first derivative condition (or gradient condition when there are multiple variables).
- To find the maximum or minimum value of a continuous and differentiable function $f(x)$ in an interval $[\underline{x}, \bar{x}]$, find the roots of $f'(x) = 0$, say $x_1, x_2, ..., x_n$, then find the maximum or minimum among $f(\underline{x}), f(\bar{x}), f(x_1), f(x_2), ..., f(x_n)$.

Intermediate Value Theorem

Math

- Intermediate Value Theorem says given a continuous function f, for any u between $f(a)$ and $f(b)$, there exists an $x \in [a, b]$ such that $f(x) = u$.
- IVT implies that if $f(a) \geqslant 0$ and $f(b) \leqslant 0$, then there exists an $x \in [a, b]$ such that $f(x) = 0$.
- Bisection method uses this observation to iteratively reduce the interval [a, b] that contains the root by a half until a and b are close enough.

Intermediate Value Theorem Diagram

Math

Search

Math

- Bisection method can be used to find a root of $f(x) = 0$ in an interval $x \in [x_0, x_1]$.

1. Start with $[x_0, x_1]$ and $x = \frac{1}{2}(x_0 + x_1)$.
2. If $f(x)$ and $f(x_0)$ has different signs, the solution is between x_0 and x, use bisection method on $[x_0, x]$.
3. If $f(x)$ and $f(x_1)$ has different signs, the solution is between x and x_1, use bisection method on $[x, x_1]$.
4. Stop when $f(x) = 0$ or x_0 and x_1 are close enough.

Bisection Diagram

Math

Search

Code

- Code for bisection search.

```
function x = bisection(f, x0, x1)
  x = 0.5 * (x0 + x1); % Find midpoint.
  if x1 - x0 < 0.0001 % Solution is close to x.
    return
  elseif f(x0) * f(x) <= 0 % Solution is in [x_0, x].
    x = bisection(f, x0, x);
  else % Solution is in [x, x_1].
    x = bisection(f, x, x1);
  end
end
```

Newton's Method

Math

- Newton's method can be used to find a root of $f(x) = 0$, given $f'(x)$, starting from initial guess x_0, preferably close to the solution.

1. Start with the initial guess $x = x_0$.
2. Repeat using Newton's formula $x = x - \dfrac{f(x)}{f'(x)}$.
3. Stop when $f(x)$ is close enough to 0 (or the number of iterations is too large).

Newton's Method Diagram

Math

Newton's Method

Code

- Code for Newton's Method

```
function x = newton(f, fp, x0)
    if abs(f(x0)) < 0.0001 % Solution is close to x0
      x = x0;
    else % Newton's update
      x = newton(f, fp, x0 - f(x0) / fp(x0));
    end
end
```

Non-Convergence

Math

- Newton's method could get stuck when $f'(x) = 0$.
- In that case, start with a different random initial guess.
- Newton's method could also diverge around an unstable root.
- In that case, a variation of Newton's method need to be used.

Secant Method

Math

- Secant method is used instead of Newton's method when the derivative function is unknown or costly to compete.
- Two initial guesses are required, x_0 and x_1, and the Newton's update is replaced by $x = x - \frac{f(x)}{\frac{f(x) - f(x')}{x - x'}} = \frac{x' f(x) - x f(x')}{f(x) - f(x')}$, where x' is the x in the previous iteration.

Secant Method Diagram

Math

Secant Method

Math

- Code for Newton's Method

```
function x = secant(f, x1, x0)
    if abs(f(x1)) < 0.0001 % Solution is close to x1
        x = x1;
    else % Secant update
        x2 = (x0 * f(x1) - x1 * f(x0)) / (f(x1) - f(x0))
        x = secant(f, x2, x1);
    end
end
```

Comparison with Newton's Method

Math

- Secant method is not the same as Netwon's method with the numerical derivative computed using finite differences, but when x and x' are close, a step using Secant method does approximate a step using Newton's method.
- In general, Newton's method usually takes fewer iterations.
- If it is costly to evaluate $f'(x)$, the secant method could be faster than Newton's method.

MATLAB Solver

Code

- *fzero(f, [x0, x1])* searches for the solution of $f(x) = 0$ between x_0 and x_1, assuming $f(x_0) f(x_1) \leqslant 0$.
- *fzero(f, x0)* starts at x_0 and search for the solution of $f(x) = 0$ using a variation of the secant method.

Extension to System of Equations

- Both Newton's method and Secant method can be extended to solving a system of non-linear equations $F(x) = 0$. The Jacobian matrix is used in place of the derivative. The updates are given by $x = x - J_F^{-1}(x) F(x)$.

Blank Slide