Newton Raphson Method

Swapnil Singh
Bank of Lithuania, KTU

December 4, 2024

Outline

- Introduction to Newton-Raphson Method
- Mathematical Foundations
- Line Search and Armijo Rule
- Convergence Properties
- Algorithm Implementation
- Practical Considerations
- Examples and Applications

Introduction

- **Purpose:** Find roots of equation $f(x) = 0$ or minimum/maximum of function
- **Key Idea:** Use local quadratic approximation
- **Advantages:**
  - Quadratic convergence when close to solution
  - Efficient for smooth functions
  - Works well for multi-dimensional problems
- **Disadvantages:**
  - Requires derivatives
  - Sensitive to starting point
  - May not converge for poor initial guesses

Mathematical Foundation

For Root Finding:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

For Optimization:

- Find where gradient is zero: $\nabla f(x) = 0$
- Update formula with line search:

$$\begin{aligned} x_{k+1} &= x_k + \alpha_k d_k \\ d_k &= -[H(x_k)]^{-1} \nabla f(x_k) \end{aligned}$$

where:

- $\nabla f(x_k)$ is the gradient
- $H(x_k)$ is the Hessian matrix
- α_k is the step size from line search

Quadratic Approximation

Taylor Series Expansion:

$$f(x_k + d) \approx f(x_k) + \nabla f(x_k)^T d + \frac{1}{2} d^T H(x_k) d$$

Finding the Minimum:

- Take derivative with respect to d and set to zero:

$$\nabla f(x_k) + H(x_k) d = 0$$

- Solve for the Newton direction:

$$d_k = -H(x_k)^{-1} \nabla f(x_k)$$

Line Search with Armijo Rule

Armijo Condition:

- Ensures sufficient decrease in function value
- Step size α_k must satisfy:

$$f(x_k + \alpha_k d_k) \leq f(x_k) + c\alpha_k \nabla f(x_k)^T d_k$$

where:

- c is typically 0.1 (Armijo parameter)
- α_k is reduced by factor β (typically 0.5)

Benefits:

- Guarantees descent property
- Improves global convergence
- Handles poor quadratic approximations

Convergence Properties

Local Convergence:

- Quadratic convergence near solution:

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

- Requires:
  - Smooth function
  - Good initial guess
  - Non-singular Hessian at solution

Global Convergence with Line Search:

- Armijo rule ensures:

$$f(x_{k+1}) < f(x_k)$$

- Convergence to local minimum under mild conditions

Algorithm Implementation

Input: Initial guess x_0, tolerance ϵ, max iterations N

Input: Armijo parameter c, reduction factor β

Output: Optimal point x^*

$k \leftarrow 0$ **while** $k < N$ **do**

Compute gradient: $g_k = \nabla f(x_k)$ Compute Hessian: $H_k = H(x_k)$ **if** $\|g_k\| < \epsilon$ **then**

return x_k

end

Solve: $H_k d_k = -g_k$ $\alpha_k \leftarrow 1$ **while** $f(x_k + \alpha_k d_k) > f(x_k) + c\alpha_k g_k^T d_k$ **do**

$\alpha_k \leftarrow \beta\alpha_k$ **if** $\alpha_k < 10^{-10}$ **then**

break

end

end

Update: $x_{k+1} = x_k + \alpha_k d_k$ $k \leftarrow k + 1$

Implementation Details

Key Components:

- **Function Evaluation:**
  - Define objective function
  - Compute gradient analytically or numerically
  - Compute Hessian analytically or numerically
- **Linear System Solution:**
  - Use `mldivide` (backslash) operator
  - Check condition number
  - Handle singular matrices
- **Line Search:**
  - Implement backtracking
  - Choose appropriate parameters
  - Handle convergence failures

Practical Considerations

Implementation Challenges:

- Choose appropriate stopping criteria
- Handle ill-conditioned Hessians
- Select good initial point
- Deal with non-convex functions
- Choose appropriate line search parameters

Line Search Parameters:

- Armijo parameter c (typically 0.1)
- Step reduction factor β (typically 0.5)
- Minimum step size threshold

Improvements:

- Strong Wolfe conditions
- Trust region methods
- Quasi-Newton methods (BFGS)
- Regularization for ill-conditioning

Example Applications

Optimization Problems:

- Maximum likelihood estimation
- Least squares problems
- Portfolio optimization
- Machine learning (model training)

Root Finding:

- Solving nonlinear equations
- Finding equilibrium points
- Solving boundary value problems
- Financial derivatives pricing

Summary

Key Points:

- Newton-Raphson combines quadratic approximation with line search
- Armijo rule ensures sufficient decrease
- Quadratic convergence near solution
- Practical implementation requires careful attention to details

When to Use:

- Smooth, well-behaved functions
- When derivatives are available
- When fast local convergence is needed
- When good initial guess is available