Newton’s method for nonlinear equations
Example: solving a nonlinear electrical circuit

Newton’s method for unconstrained minimization
Solving the first-order optimality conditions
Minimizing the second-order approximation
Relationship with the Gauss/Newton method
Relationship with iteratively reweighted least-squares
Example: $\ell_p$ regression
Newton’s method for nonlinear equations

given $f : \mathbb{R}^n \to \mathbb{R}^n$, find $x \in \mathbb{R}^n$ such that

$$f(x) = 0$$

first-order Taylor approximation at current guess:

$$f(x^{(k)}) + Df(x^{(k)})(x - x^{(k)}) \approx 0$$

next guess is solution of first-order approximation:

$$x^{(k+1)} = x^{(k)} - Df(x^{(k)})^{-1}f(x^{(k)})$$

(assuming $Df(x^{(k)})$ is nonsingular)
Graphical intuition

Newton’s method for solving nonlinear equations:

\[ f(x) = 0 \]
Shockley diode equation:

\[ I = I_s \left( \exp\left(\frac{V}{nV_T}\right) - 1 \right) \]

where
- \( I \) is current through diode
- \( V \) is voltage across diode
- \( I_s \) is saturation current of diode (physical parameter of diode)
- \( V_T \) is thermal voltage (temperature-dependent parameter)
- \( n \) is ideality factor (physical parameter of diode)
A diode circuit

voltage seen by resistor is $V_{out} = x_1 - x_2$
Circuit equations

Kirchoff’s current law entering node 1:

\[ I_s \left( \exp \left( \frac{V_{in} - x_1}{nV_T} \right) - 1 \right) + I_s \left( \exp \left( \frac{-x_1}{nV_T} \right) - 1 \right) + \frac{x_2 - x_1}{r} = 0 \]

Kirchoff’s current law leaving node 2:

\[ I_s \left( \exp \left( \frac{x_2 - V_{in}}{nV_T} \right) - 1 \right) + I_s \left( \exp \left( \frac{x_2}{nV_T} \right) - 1 \right) + \frac{x_2 - x_1}{R} = 0 \]
Applying Newton’s method

find root of function:

\[ f(x) = \begin{bmatrix}
    I_s \left( \exp \left( \frac{V_{in} - x_1}{nV_T} \right) - 1 \right) + I_s \left( \exp \left( \frac{-x_1}{nV_T} \right) - 1 \right) + \frac{x_2 - x_1}{R} \\
    I_s \left( \exp \left( \frac{x_2 - V_{in}}{nV_T} \right) - 1 \right) + I_s \left( \exp \left( \frac{x_2}{nV_T} \right) - 1 \right) + \frac{x_2 - x_1}{R}
\end{bmatrix} \]

with derivative:

\[ Df(x) = \begin{bmatrix}
    -\frac{I_s}{nV_T} \exp \left( \frac{V_{in} - x_1}{nV_T} \right) \\
    -\frac{I_s}{nV_T} \exp \left( \frac{-x_1}{nV_T} \right) - \frac{1}{R} \\
    -\frac{l_s}{nV_T} \exp \left( \frac{x_2 - V_{in}}{nV_T} \right) + \frac{l_s}{nV_T} \exp \left( \frac{x_2}{nV_T} \right) + \frac{1}{R}
\end{bmatrix} \]
Initial guess for Newton’s method

simplified circuits with ideal diodes:

$V_{\text{in}} < 0$

$V_{\text{in}} > 0$

initial point for Newton’s method:

$x^{(0)} = \begin{bmatrix} \max(V_{\text{in}}, 0) \\ \min(V_{\text{in}}, 0) \end{bmatrix}$
input/output curve for full-wave rectifier

\[ I_s = 1 \text{fA}, \ V_T = 26 \text{mV}, \ n = 1, \ R = 1 \text{M\Omega} \]
Newton’s method for nonlinear equations

- Solving the first-order optimality conditions
- Minimizing the second-order approximation
- Relationship with the Gauss/Newton method
- Relationship with iteratively reweighted least-squares
- Example: $\ell_p$ regression
Solving the first-order optimality conditions

given $f : \mathbb{R}^n \rightarrow \mathbb{R}$, solve unconstrained optimization problem:

$$
\begin{align*}
\text{minimize} & : f(x) \\
\text{subject to} & : x \in \mathbb{R}^n
\end{align*}
$$

first-order optimality condition:

$$
\nabla f(x) = 0
$$

use Newton’s method to solve first-order optimality condition:

$$
\begin{align*}
x^{(k+1)} &= x^{(k)} - D \nabla f(x^{(k)})^{-1} \nabla f(x^{(k)}) \\
&= x^{(k)} - \nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})
\end{align*}
$$
Minimizing the second-order approximation

second-order Taylor approximation at current guess $x^{(k)}$:

$$f(x) \approx f(x^{(k)}) + \nabla f(x^{(k)})^T (x - x^{(k)})$$
$$+ \frac{1}{2} (x - x^{(k)})^T \nabla^2 f(x^{(k)})(x - x^{(k)})$$

if approximation is convex, unique minimizer is

$$x^{(k+1)} = x^{(k)} - \nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})$$
Graphical intuition

Newton’s method for unconstrained minimization:

\[ f(x) \]

\[ x(0) \quad x(1) \quad x(2) \]

Example: \( \ell_p \) regression
Newton’s method with sum-of-squares objective

sum-of-squares objective:

\[ f(x) = \frac{1}{2} \sum_{i=1}^{m} r_i(x)^2 \]

first and second partial derivatives:

\[ \frac{\partial f}{\partial x_j} = \sum_{i=1}^{m} r_i(x) \frac{\partial r_i}{\partial x_j}, \quad \frac{\partial^2 f}{\partial x_j \partial x_k} = \sum_{i=1}^{m} \left( \frac{\partial r_i}{\partial x_j} \frac{\partial r_i}{\partial x_k} + r_i(x) \frac{\partial^2 r_i}{\partial x_j \partial x_k} \right) \]

gradient and Hessian:

\[ \nabla f(x) = Dr(x)^T r(x), \quad \nabla^2 f(x) = Dr(x)^T Dr(x) + \sum_{i=1}^{m} r_i(x) \nabla^2 r_i(x) \]
An approximate Hessian

\[ \nabla^2 f(x) = Dr(x)^T Dr(x) + \sum_{i=1}^{m} r_i(x) \nabla^2 r_i(x) \]

\[ \approx Dr(x)^T Dr(x) \]

approximation works well if
- \( r_i(x) \) is small, or
- \( r_i(x) \) is approximately affine
Newton’s method for nonlinear equations
Newton’s method for unconstrained minimization

The Gauss/Newton method

Newton’s method: with approximate Hessian:

\[ x^{(k+1)} = x^{(k)} - \nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)}) \]
\[ \approx x^{(k)} - (Dr(x^{(k)})^T Dr(x^{(k)}))^{-1} Dr(x^{(k)})^T r(x^{(k)}) \]
\[ = x^{(k)} - Dr(x^{(k)})^\dagger r(x^{(k)}) \]

Gauss/Newton update equation!
Newton’s method for sum of penalties of linear functions

sum of penalties of linear functions:

\[ f(x) = \sum_{i=1}^{m} \phi(A_i x - y_i) \]

first and second partial derivatives:

\[
\frac{\partial f}{\partial x_j} = \sum_{i=1}^{m} \phi'(A_i x - y_i) A_{ij}, \quad \frac{\partial^2 f}{\partial x_j \partial x_k} = \sum_{i=1}^{m} \phi''(A_i x - y_i) A_{ij} A_{ik}
\]

gradient and Hessian:

\[
\nabla f(x) = A^\top \phi'(Ax - y), \quad \nabla^2 f(x) = A^\top \text{diag}(\phi''(Ax - y)) A
\]

(\( \phi \) and its derivatives overloaded for vectors)
Newton’s method as iteratively reweighted least squares

Newton update:

\[
x^{(k+1)} = x^{(k)} - \nabla^2 f(x^{(k)})^{-1} \nabla f(x^{(k)})
\]

\[
= x^{(k)} - (A^T \text{diag}(\phi''(Ax^{(k)} - y))A)^{-1} A^T \phi'(Ax^{(k)} - y)
\]

\[
= (A^T W^{(k)} A)^{-1} A^T W^{(k)} z^{(k)}
\]

- weight matrix:

\[
W^{(k)} = \text{diag}(\phi''(Ax^{(k)} - y))
\]

- adjusted response vector:

\[
z^{(k)} = Ax^{(k)} - \text{diag}(\phi''(Ax^{(k)} - y))^{-1} \phi'(Ax^{(k)} - y)
\]
Assumptions for iteratively reweighted least squares

assume

- no penalty for zero residual:
  \[ \phi(0) = 0 \]

- zero is local minimum of penalty function:
  \[ \phi'(0) = 0 \quad \text{and} \quad \phi''(0) > 0 \]

- current residual is small:
  \[ Ax^{(k)} \approx y \]
Approximate weight matrix and adjusted response vector

(vector multiplication and division defined componentwise)

second-order Taylor series near zero:

\[ \phi(Ax^{(k)} - y) \approx \phi(0) + \phi'(0)(Ax^{(k)} - y) + \frac{1}{2}\phi''(0)(Ax^{(k)} - y)^2 \]

\[ \approx \frac{1}{2}\phi''(Ax^{(k)} - y)(Ax^{(k)} - y)^2 \]

approximate weight matrix:

\[ W^{(k)} = \text{diag}(\phi''(Ax^{(k)} - y)) \approx 2 \text{diag}\left(\frac{\phi(Ax^{(k)} - y)}{(Ax^{(k)} - y)^2}\right) \]

approximate adjusted response vector:

\[ z^{(k)} = Ax^{(k)} - \text{diag}(\phi''(Ax^{(k)} - y))\phi'(Ax^{(k)} - y) \]

\[ \approx y - \text{diag}(\phi''(y - y))\phi'(y - y) = y \]
Iteratively reweighted least-squares

Newton’s method with approximate weight matrix and adjusted response vector:

\[ x^{(k+1)} \]

\[ = (A^T W^{(k)} A)^{-1} A^T W^{(k)} z^{(k)} \]

\[ = \left( \frac{1}{2} A^T W^{(k)} A \right)^{-1} A^T \left( \frac{1}{2} W^{(k)} \right) z^{(k)} \]

\[ \approx \left( A^T \text{diag} \left( \frac{\phi(Ax^{(k)} - y)}{(Ax^{(k)} - y)^2} \right) A \right)^{-1} A^T \text{diag} \left( \frac{\phi(Ax^{(k)} - y)}{(Ax^{(k)} - y)^2} \right) y \]

update equation for iteratively reweighted least squares!
Example: $\ell_p$ regression

$\ell_p$ regression problem:

$$\minimize_{x \in \mathbb{R}^n} : \left( \sum_{i=1}^{m} |(Ax - y)_i|^p \right)^{\frac{1}{p}}$$

- twice differentiable for $p = 2, 4, 6, 8, \ldots$
- can apply Newton’s method
Applying Newton’s method

equivalent objective function:

\[ J(x) = \frac{1}{p} \sum_{k=1}^{m} (Ax - y)_k^p \]

first and second derivatives:

\[ \frac{\partial J}{\partial x_j} = \sum_{k=1}^{m} (Ax - y)_k^{p-1} A_{kj}, \quad \frac{\partial^2 J}{\partial x_i \partial x_j} = (p - 1) \sum_{k=1}^{m} (Ax - y)_k^{p-2} A_{ki} A_{kj} \]

gradient and Hessian:

\[ \nabla J(x) = A^T (Ax - y)^{p-1}, \quad \nabla^2 J(x) = (p - 1) A^T \text{diag}((Ax - y)^{p-2}) A \]

where \( z^q = (z_1^q, \ldots, z_m^q) \) is componentwise vector exponentiation
Initial guess for Newton’s method

let \(\hat{x}^{(p)}\) be solution of \(\ell_p\) regression problem

- \(p = 2\) is linear least-squares problem, \(\hat{x}^{(2)} = A^\dagger y\)
- for \(p = 4, 6, 8, \ldots\), use \(\hat{x}^{(p-2)}\) as initial guess
Measurements with uniform noise

\[ N = 1000 \text{ randomly generated estimation problems:} \]

\[ y = A x + \epsilon, \]

\[ \begin{align*}
\text{\> } x & \in \mathbb{R}^n \text{ for } n = 3 \\
\text{\> } y & \in \mathbb{R}^m \text{ for } m = 100 \\
\text{\> } A_{ij}, x_j & \sim \mathcal{N}(0, 1), \\
\text{\> } \epsilon_i & \sim \mathcal{U}[-1, +1] \\
\text{\> } A_{ij}, x_j, \epsilon_i & \text{ independent} \\
\text{\> } \text{maximum-likelihood estimate of } x & \text{ minimizes } \| A x - y \|_\infty
\end{align*} \]
Mean estimation error

\[ \left\| x - \hat{x}(p) \right\| \]

\[ p \]

Example: $\ell_p$ regression