14. Nonlinear least-squares

- definition
- Newton's method
- Gauss-Newton method

Nonlinear least-squares

$$\text{minimize} \quad \sum_{i=1}^{m} r_i(x)^2 = \|r(x)\|^2$$

- r_i is a nonlinear function of the n-vector of variables x
- $r(x) = (r_1(x), r_2(x), \ldots, r_m(x))$
- reduces to linear least-squares if $r(x) = Ax - b$
- $g(x) = \|r(x)\|^2$ may have multiple local minima; finding the global minimum is usually very hard
- a nonlinear minimization problem; can apply Newton's method

Interpretation as overdetermined nonlinear equations

m nonlinear equations in n variables

$$\begin{aligned} r_1(x_1, x_2, \ldots, x_n) &= 0 \\ r_2(x_1, x_2, \ldots, x_n) &= 0 \\ &\vdots \\ r_m(x_1, x_2, \ldots, x_n) &= 0 \end{aligned}$$

usually there is no x that satisfies $r(x) = 0$

instead we can calculate the vector x that minimizes

$$\|r(x)\|^2 = \sum_{i=1}^{m} r_i(x)^2$$

Inductor modeling example

50 nonlinear equations in 5 variables $x_1, \ldots, x_5$

$$\exp(x_1) n_i^{x_2} w_i^{x_3} d_i^{x_4} D_i^{x_5} \approx L_i, \quad i = 1, \ldots, 50$$

method 1 (exercise 8.4): suppose we are free to choose the error criterion

if we choose to minimize the sum of squared errors on a *logarithmic* scale,

$$\text{minimize} \quad \sum_{i=1}^{50} \left(\log(\exp(x_1) n_i^{x_2} w_i^{x_3} d_i^{x_4} D_i^{x_5}) - \log L_i\right)^2,$$

we obtain a *linear* least-squares problem: minimize $\|Ax - b\|^2$ where

$$A = \left[\begin{array}{ccccc} 1 & \log n_1 & \log w_1 & \log d_1 & \log D_1 \\ 1 & \log n_2 & \log w_2 & \log d_2 & \log D_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \log n_{50} & \log w_{50} & \log d_{50} & \log D_{50} \end{array}\right], \qquad b = \left[\begin{array}{c} \log L_1 \\ \log L_2 \\ \vdots \\ \log L_{50} \end{array}\right]$$

method 2: minimize the sum of squared errors on a linear scale

$$\text{minimize} \quad \sum_{i=1}^{50} \left(\exp(x_1) n_i^{x_2} w_i^{x_3} d_i^{x_4} D_i^{x_5} - L_i\right)^2$$

this is a *nonlinear* least-squares problem: minimize $\sum_{i=1}^{50} r_i(x)^2$ where

$$\begin{aligned} r_i(x) &= \exp(x_1) n_i^{x_2} w_i^{x_3} d_i^{x_4} D_i^{x_5} - L_i \\ &= \exp(x_1 + x_2 \log n_i + x_3 \log w_i + x_4 \log d_i + x_5 \log D_i) - L_i \end{aligned}$$

- much harder than linear least-squares (may have multiple local minima)
- requires an iterative method
- can use method 1 to find a starting point

Navigation from range measurements

estimate position (u, v) from distances to m beacons at locations (p_k, q_k)

measured distances:

$$\rho_i = \sqrt{(u - p_i)^2 + (v - q_i)^2} + w_i, \quad i = 1, \ldots, m$$

where w_i is range error, unknown but small

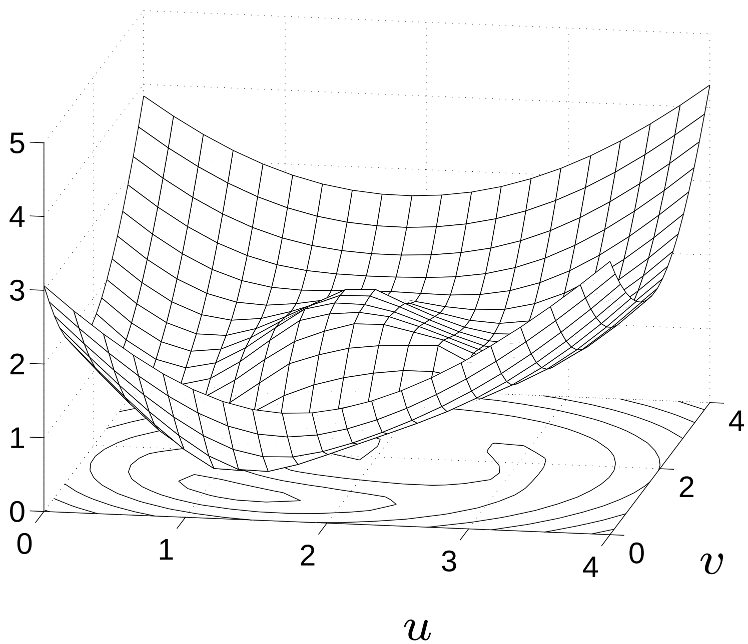
nonlinear least-squares estimate: choose estimates (u, v) by minimizing

$$\begin{aligned} g(u, v) &= \sum_{i=1}^{m} r_i(u, v)^2 \\ &= \sum_{i=1}^{m} \left(\sqrt{(u - p_i)^2 + (v - q_i)^2} - \rho_i \right)^2 \end{aligned}$$

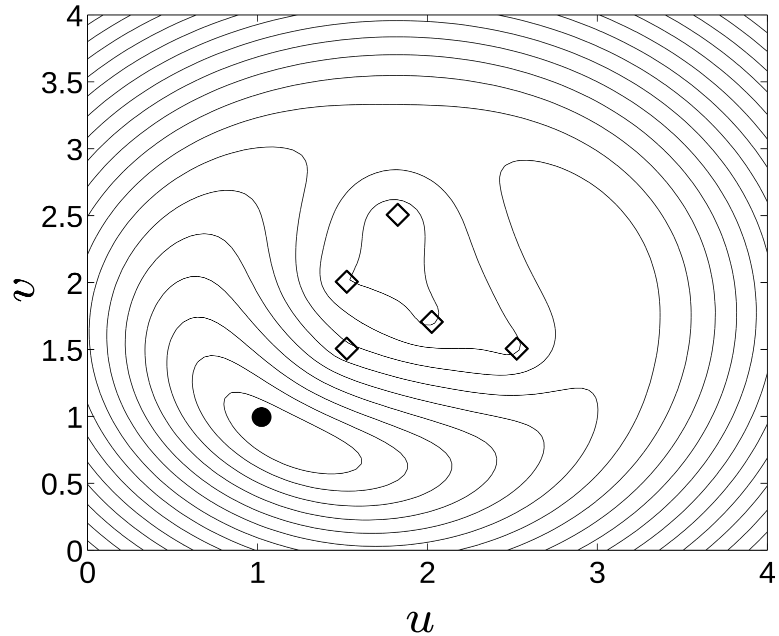
example

- correct position is $(1, 1)$ (•)
- $m = 5$ beacons at positions $\diamond$
- range measurements accurate to ± 0.2

graph of $g(u, v)$



contour lines of $g(u, v)$



(global) minimum at $(1.18, 0.82)$
local minimum at $(2.99, 2.12)$, local maximum at $(1.94, 1.87)$

Newton's method for nonlinear least-squares

apply method of page 13-25 to $g(x) = \|r(x)\|^2 = \sum_{i=1}^{m} r_i(x)^2$

first and second derivatives of g:

$$\begin{aligned}
\frac{\partial g(x)}{\partial x_k} &= 2\sum_{i=1}^{m} r_i(x)\frac{\partial r_i(x)}{\partial x_k} \\
\frac{\partial^2 g(x)}{\partial x_j \partial x_k} &= 2\sum_{i=1}^{m}\left(r_i(x)\frac{\partial^2 r_i(x)}{\partial x_j \partial x_k} + \frac{\partial r_i(x)}{\partial x_j}\frac{\partial r_i(x)}{\partial x_k}\right)
\end{aligned}$$

$i.e.$, the gradient and Hessian of g are

$$\begin{aligned}
\nabla g(x) &= 2\sum_{i=1}^{m} r_i(x)\nabla r_i(x) \\
\nabla^2 g(x) &= 2\sum_{i=1}^{m}\left(r_i(x)\nabla^2 r_i(x) + \nabla r_i(x)\nabla r_i(x)^T\right)
\end{aligned}$$

example (inductor problem of page 14-4)

- method 1 (linear least-squares):

$$x_1 = -7.25, \quad x_2 = 1.38, \quad x_3 = -0.48, \quad x_4 = 0.28, \quad x_5 = 1.21$$

- method 2: apply Newton's method to

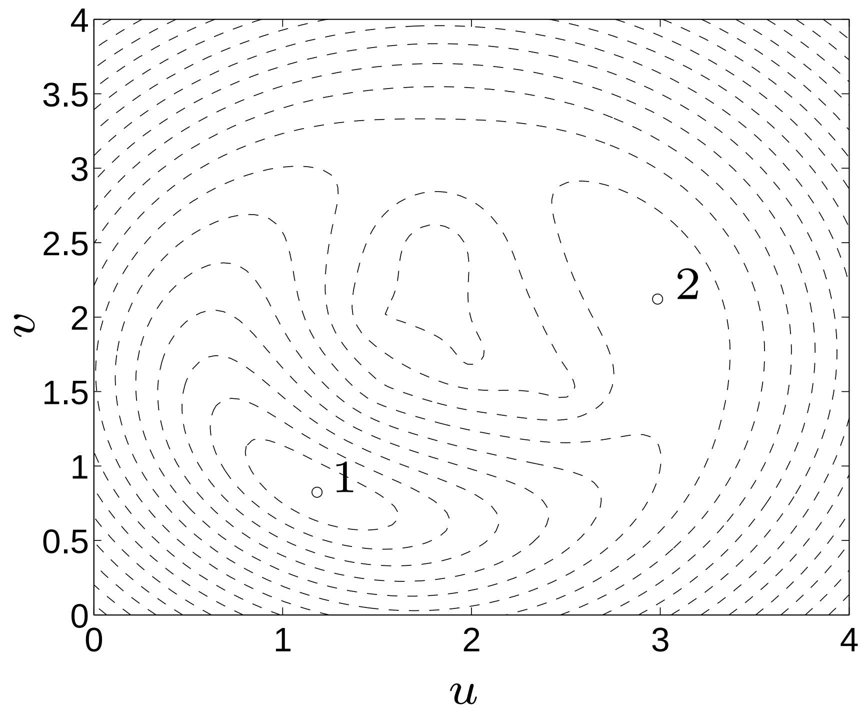
$$g(x) = \sum_{i=1}^{50} \left(\exp(x_1 + x_2 \log n_i + x_3 \log w_i + x_4 \log d_i + x_5 \log D_i) - L_i\right)^2$$

use as starting point the linear least-squares solution

converges in four iterations to

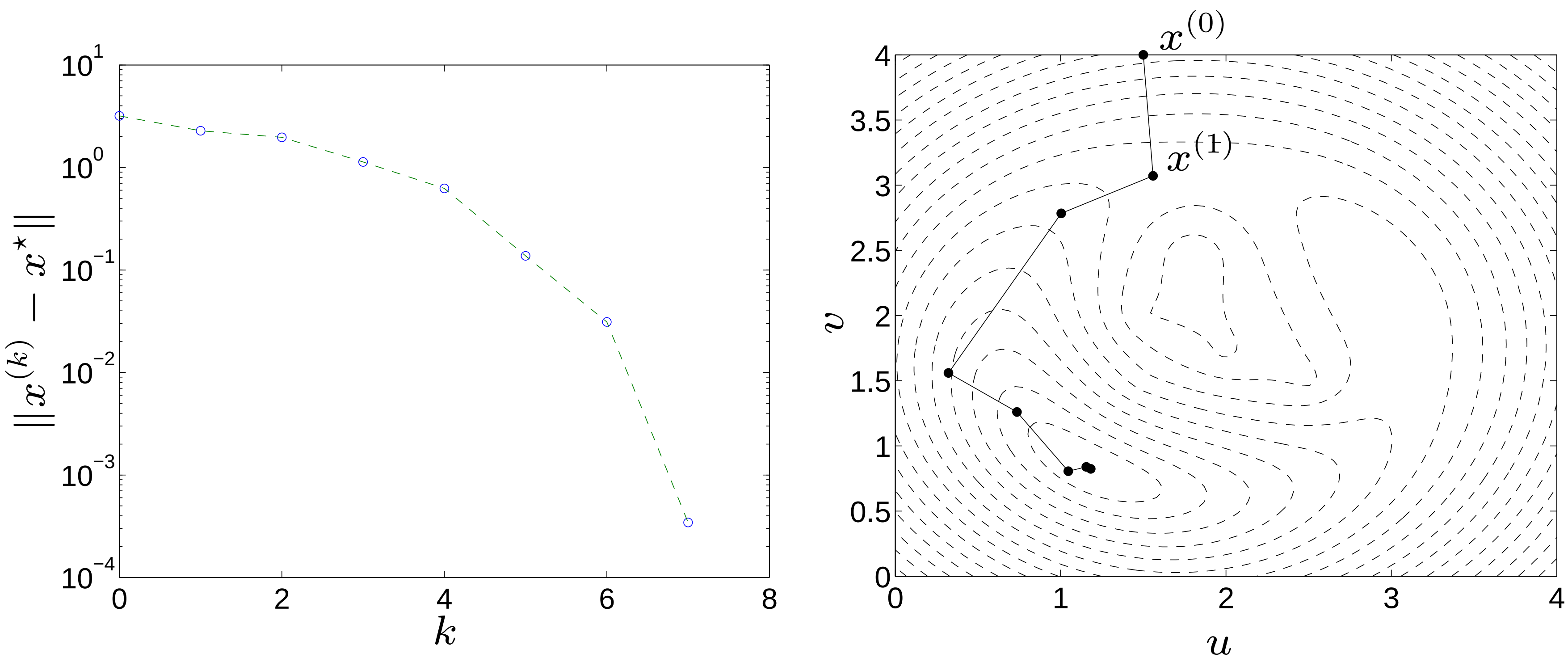
$$x_1 = -7.14, \quad x_2 = 1.32, \quad x_3 = -0.53, \quad x_4 = 0.22, \quad x_5 = 1.27$$

example (navigation problem of page 14-6)



- from starting points $(0,0)$, $(0,4)$, $(4,0)$, converges to 1 (minimum)
- from starting points $(4,4)$, $(2,2)$, converges to point 2 (local minimum)

convergence from starting point $x^{(0)} = (1.5, 4)$



- converges to minimum at $(1.18, 0.82)$
- 2nd and 3rd iteration use negative gradient direction; other iterations use Newton direction

Gauss-Newton method for nonlinear least-squares

a simpler alternative to Newton's method for minimizing

$$g(x) = \sum_{i=1}^{m} r_i(x)^2$$

- start at some initial guess $x^{(0)}$
- at iteration k, linearize $r_i(x)$ around current guess $x^{(k)}$:

$$r_i(x) \approx r_i(x^{(k)}) + \nabla r_i(x^{(k)})^T (x - x^{(k)})$$

new guess $x^{(k+1)}$ is the minimizer of

$$\sum_{i=1}^{m} \left(r_i(x^{(k)}) + \nabla r_i(x^{(k)})^T (x - x^{(k)}) \right)^2$$

$i.e.$, sum of the squares of the linearized residuals to find $x^{(k+1)}$ from $x^{(k)}$, solve a linear least-squares problem

$$\text{minimize} \quad \|A^{(k)}x - b^{(k)}\|^2$$

with

$$A^{(k)} = \begin{bmatrix} \nabla r_1(x^{(k)})^T \\ \nabla r_2(x^{(k)})^T \\ \vdots \\ \nabla r_m(x^{(k)})^T \end{bmatrix}, \qquad b^{(k)} = \begin{bmatrix} \nabla r_1(x^{(k)})^T x^{(k)} - r_1(x^{(k)}) \\ \nabla r_2(x^{(k)})^T x^{(k)} - r_2(x^{(k)}) \\ \vdots \\ \nabla r_m(x^{(k)})^T x^{(k)} - r_m(x^{(k)}) \end{bmatrix}$$

- advantage (over Newton's method): no second derivatives of r_i needed
- disadvantage: convergence is slower

Summary: Gauss-Newton method

given initial x, tolerance $\epsilon > 0$.
repeat

1. evaluate $r_i(x)$ and $\nabla r_i(x)$ for $i = 1, \ldots, m$, and calculate

$$r := \begin{bmatrix} r_1(x) \\ r_2(x) \\ \vdots \\ r_m(x) \end{bmatrix}, \qquad A := \begin{bmatrix} \nabla r_1(x)^T \\ \nabla r_2(x)^T \\ \vdots \\ \nabla r_m(x)^T \end{bmatrix}, \qquad b := Ax - r.$$

2. **if** $\|2A^T r\| \leq \epsilon$, **return** x.
3. $x := (A^T A)^{-1} A^T b$.

until maximum number of iterations is exceeded

in step 2, note that $2A^T r = \nabla g(x)$

Interpretation as modified Newton method

(notation: $x = x^{(k)}$, $x^+ = x^{(k+1)}$)

in Gauss-Newton method (from normal eqs. of LS problem on page 14-13)

$$
\begin{aligned}
x^+ &= \left(\sum_{i=1}^m \nabla r_i(x) \nabla r_i(x)^T\right)^{-1} \left(\sum_{i=1}^m \nabla r_i(x)(\nabla r_i(x)^T x - r_i(x))\right) \\
&= x - \left(\sum_{i=1}^m \nabla r_i(x) \nabla r_i(x)^T\right)^{-1} \left(\sum_{i=1}^m r_i(x) \nabla r_i(x)\right)
\end{aligned}
$$

interpretation: take unit step in direction

$$v = -H^{-1} \nabla g(x)$$

where

$$H = 2 \sum_{i=1}^m \nabla r_i(x) \nabla r_i(x)^T$$

compare with the Newton direction at x:

$$v = \nabla^2 g(x)^{-1} \nabla g(x)$$

where (from page 14-8)

$$\nabla^2 g(x) = 2 \sum_{i=1}^{m} \left(\nabla r_i(x) \nabla r_i(x)^T + r_i(x) \nabla^2 r_i(x)\right)$$

interpretation of GN-method: replace $\nabla^2 g(x)$ in Newton's method by

$$H = 2 \sum_{i=1}^{m} \nabla r_i(x) \nabla r_i(x)^T$$

- $H \approx \nabla^2 g(x)$ if the residuals $r_i(x)$ are small
- advantage: no need to evaluate $\nabla^2 r_i(x)$
- H is always positive semidefinite

Gauss-Newton method with backtracking

---

given initial x, tolerance $\epsilon > 0$, parameter $\alpha \in (0, 1/2)$
repeat

1. evaluate $r_i(x)$ and $\nabla r_i(x)$ for $i = 1, \ldots, m$, and calculate

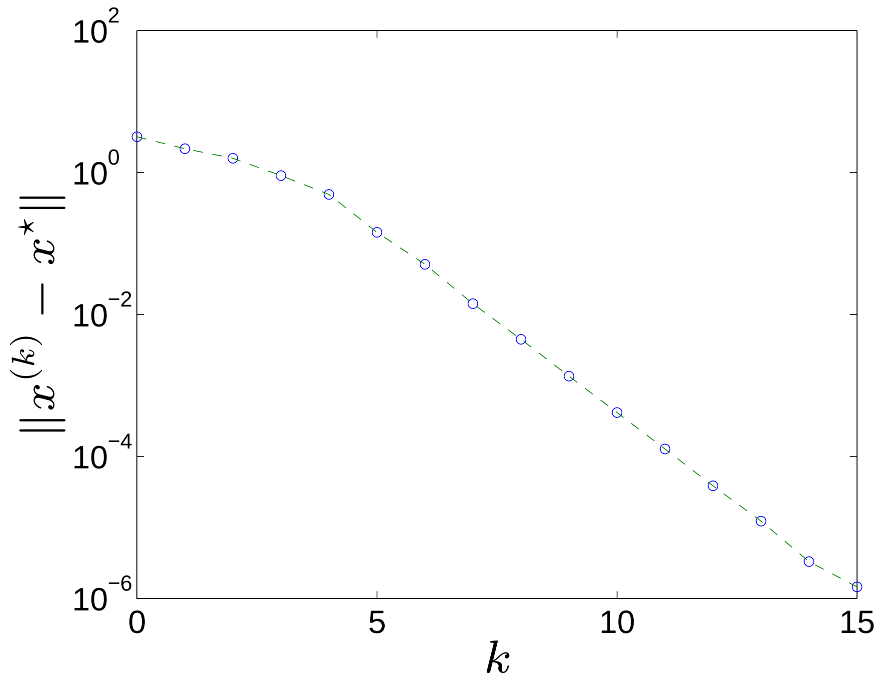
$$r := \begin{bmatrix} r_1(x) \\ \vdots \\ r_m(x) \end{bmatrix}, \qquad A := \begin{bmatrix} \nabla r_1(x)^T \\ \vdots \\ \nabla r_m(x)^T \end{bmatrix}.$$

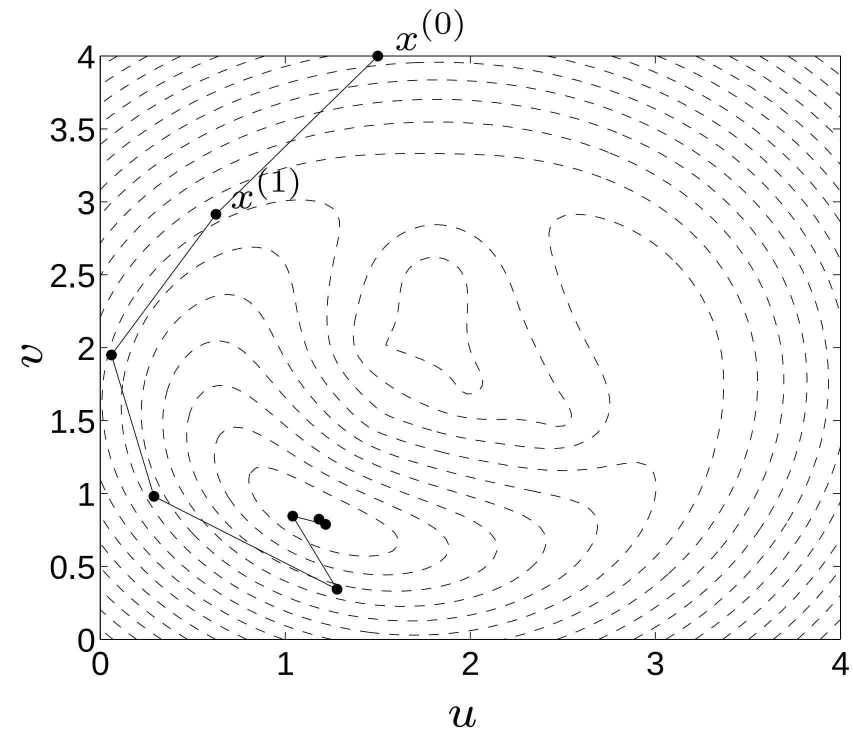
2. **if** $\|2A^T r\| \leq \epsilon$, **return** x.
3. $v := -(A^T A)^{-1} A^T r$.
4. $t := 1$
   while $\sum_{i=1}^m r_i(x + tv)^2 > \|r\|^2 + \alpha(2r^T Av)t$, $t := t/2$.
5. $x := x + tv$.

until maximum number of iterations is exceeded

---

example (same problem as page 14-6)





local convergence is slower than Newton's method