Introduction to (Bayesian) Estimation
MAE 5020

Nonlinear Least-squares estimation[1]

Oklahoma State University

August 16, 2023

[1] adapted from
http://www.seas.ucla.edu/~vandenbe/133A/lectures/nlls.pdf

Nonlinear Least-squares estimation (NLSE)

Consider N measurements: $Z(k) = F(\theta, k) + V(k)$, $\theta \in \mathbb{R}^n$

- Find $\hat{\theta}(k)$ that minimizes (unweighted $W = I$)

$$J(\hat{\theta}(k)) = \|Z(k) - F(\hat{\theta}(k), k)\|^2$$

- Expanded form:

$$\|Z(k) - F(\hat{\theta}(k), k)\|^2 = \sum_{i=1}^{k} \|z(i) - F_i(\hat{\theta}(k))\|^2$$

- Define $f_i(\hat{\theta}) = z(i) - F_i(\hat{\theta})$ and rewrite

$$J(\hat{\theta}(k)) = \sum_{i=1}^{k} \|f_i(\hat{\theta}(k))\|^2 = \|{\color{red} f(\hat{\theta}(k))}\|^2$$

Definitions

- $F_i(\theta)$, $i = 1, \cdots, k$, are differentiable functions of a vector $\theta \in \mathbb{R}^n$
- $f(\theta)$ is a function with components $f_i(\theta)$:
- Linear least squares when $f(\theta) = H\theta - Z$.

Applications of NLSE: Localization from range measurements

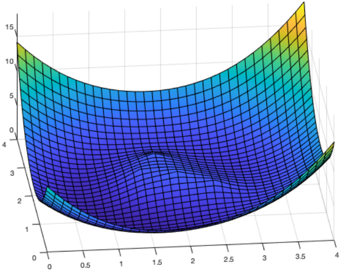
- θ represent an unknown location in 2-D or 3-D
- Distance to known points $a_1, \cdots, a_k$ is measured:

$$r_i = \|\theta - a_i\| + v_i, \quad i = 1, \cdots, k$$

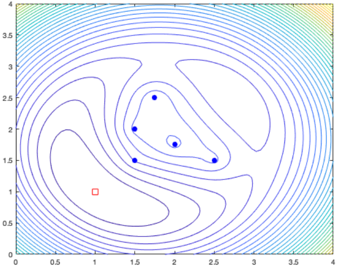
NLSE: estimate θ by $\hat{\theta}$ that minimizes

$$J(\hat{\theta}) = \sum_{i=1}^{k} (\|\theta - a_i\| - r_i)^2$$

Example



plot of $\|f(\hat{\theta})\|$



contour plof $\|f(\hat{\theta})\|^2$

- True location $(1, 1)$ marked in red
- Five points marked in blue

Application: Model fitting

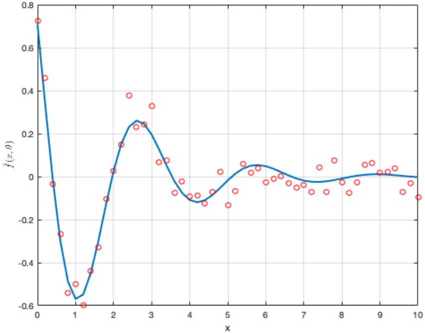
$$\min_{\theta} \sum_{i=1}^{k} \|z(i) - F(x(i), \theta)\|^2$$

- A nonlinear model $F(x(i), \theta)$ (e.g., a NN) parameterized by θ (e.g., weights, biases)
- Data points $(x(1), z(1)), \cdots, (x(k), z(k))$
- Loss function in ML: mean square loss
- Recall the linear model fitting example: $F(x, \theta) = \sum_i \theta_i f_i(x)$
- We now allow F to be nonlinear

Illustration

Second order system response

$$F(x,\theta) = \theta_1 e^{\theta_2 x} \cos(\theta_3 x + \theta_4)$$



$$\min \sum_{i=1}^{k} (\theta_1 e^{\theta_2 x(i)} \cos(\theta_3 x(i) + \theta_4) - z(i))^2$$

Derivatives/Gradient

Gradient of a differentiable function $g : \mathbb{R}^n \to \mathbb{R}$ at $z \in \mathbb{R}^n$ is a column vector

$\nabla g(z) =$

Linearization around z allows approximation of $g(\cdot)$

$$\begin{aligned} g(x) \approx \hat{g}(x) &= g(z) + \frac{\partial g}{\partial x_1}(z)(x - z_1) + \\ &= \end{aligned}$$

Jacobian matrix

Jacobian of a differentiable function $f : \mathbb{R}^n \to \mathbb{R}^k$ at $z \in \mathbb{R}^n$:

$$Df(z) = \frac{\partial f(x)}{x}(z) = \begin{pmatrix} \frac{\partial f_1(x)}{x_1}(z) & \cdots & \frac{\partial f_1(x)}{x_n}(z) \\ \vdots & & \vdots \\ \frac{\partial f_k(x)}{x_1}(z) & \cdots & \frac{\partial f_k(x)}{x_n}(z) \end{pmatrix} = \begin{pmatrix} \nabla f_1(z)^T \\ \vdots \\ \nabla f_k(z)^T \end{pmatrix}$$

Linearization leads to approximation:

$$f(x) \approx \hat{f}(x) = f(z) + Df(z)(x - z)$$

Gradient of $J(\hat{\theta})$

$$J(\hat{\theta}) = \sum_{i=1}^{k} \|f_i(\hat{\theta})\|^2 = \|f(\hat{\theta})\|^2 \triangleq g(\hat{\theta})$$

- Derivative of g w.r.t. $\hat{\theta}_j$ (assume $f_i(\hat{\theta})$ is a scalar):

$$\frac{\partial g}{\partial \hat{\theta}_j} = 2 \sum_{i=1}^{k} f_i(\hat{\theta}) \frac{\partial f_i}{\partial \hat{\theta}_j}$$

- Gradient of g at $\hat{\theta}$

$$\nabla g(\hat{\theta}) = \begin{pmatrix} \frac{\partial g}{\partial \hat{\theta}_1} \\ \vdots \\ \frac{\partial g}{\partial \hat{\theta}_n} \end{pmatrix} = 2 \sum_{i=1}^{k} f_i(\hat{\theta}) \nabla f_i(\hat{\theta}) = 2 Df(\hat{\theta})^T f(\hat{\theta})$$

Necessary optimality condition

$$\min g(\hat{\theta}) = \sum_{i=1}^{k} \|f_i(\hat{\theta})\|^2 = \|f(\hat{\theta})\|^2$$

- Necessary condition: if $\hat{\theta}$ minimizes $g(\hat{\theta})$, it must satisfy

$$\nabla g(\hat{\theta}) = 2Df(\hat{\theta})^T f(\hat{\theta}) = 0$$

- For the linear case, where $f(\hat{\theta}) = H\hat{\theta} - Z$, we have

$$\nabla g(\hat{\theta}) =$$

- For general nonlinear f, this condition is only necessary, not sufficient for optimality.

Algorithms to find possible $\hat{\theta}$

- Gradient descent

- Gauss-Newton method

- Levenberg-Marquardt method

Gauss-Newton algorithm

$$\min g(\hat{\theta}) = \sum_{i=1}^{k} \|f_i(\hat{\theta})\|^2 = \|f(\hat{\theta})\|^2$$

Start with some initial guess $\hat{\theta}^{(1)}$, and repeat for $\ell = 1, 2, \cdots$:

1. Linearize $f(\hat{\theta})$ around $\hat{\theta}^{(\ell)}$:

$$\hat{f}(\hat{\theta}, \hat{\theta}^{(\ell)}) = f(\hat{\theta}^{(\ell)}) + Df(\hat{\theta}^{(\ell)})(\hat{\theta} - \hat{\theta}^{(\ell)})$$

2. Use $\hat{f}(\hat{\theta}, \hat{\theta}^{(\ell)})$ as an approximation to $f(\hat{\theta})$ and minimize $\|\hat{f}(\hat{\theta}, \hat{\theta}^{(\ell)})\|^2$
3. Set $\hat{\theta}^{(\ell+1)}$ to the solution from Step 2

Step 2

$$\min \|f(\hat{\theta}^{(\ell)}) + Df(\hat{\theta}^{(\ell)})(\hat{\theta} - \hat{\theta}^{(\ell)})\|^2$$

- Given $\hat{\theta}^{(\ell)}$, this is a linear LSE problem.
- If $Df(\hat{\theta}^{(\ell)})$ has linearly independent columns,

$$\hat{\theta}^{(\ell+1)} = \hat{\theta}^{(\ell)} - \left(Df(\hat{\theta}^{(\ell)})^T Df(\hat{\theta}^{(\ell)})\right)^{-1} Df(\hat{\theta}^{(\ell)})^T f(\hat{\theta}^{(\ell)})$$

- Δx^{ℓ} is the same as $-\frac{1}{2}\left(Df(\hat{\theta}^{(\ell)})^T Df(\hat{\theta}^{(\ell)})\right)^{-1} \nabla g(\hat{\theta}^{(\ell)})$

What if columns of $Df(\hat{\theta}^{(\ell)})$ are linearly dependent?

$$\hat{\theta}^{(\ell+1)} = \hat{\theta}^{(\ell)} - \left(Df(\hat{\theta}^{(\ell)})^T Df(\hat{\theta}^{(\ell)}) + \lambda^{(\ell)} I\right)^{-1} Df(\hat{\theta}^{(\ell)})^T f(\hat{\theta}^{(\ell)})$$

- Levenberg-Marquardt update: λ is a regularization parameter
- Strategies to update λ^{ℓ} are possible.
- Trust-region: min $\|\hat{f}(\hat{\theta}, \hat{\theta}^{(\ell)})\|^2$ subject to $\|\hat{\theta} - \hat{\theta}^{(\ell)}\| < \gamma$.

LM method

Start with some initial guess $\hat{\theta}^{(1)}$ and $\lambda^{(\ell)}$, and repeat for $\ell = 1, 2, \cdots$:

1. Evaluate $f(\hat{\theta}^{(\ell)})$ and set $H = Df(\hat{\theta}^{(\ell)})$
2. Compute

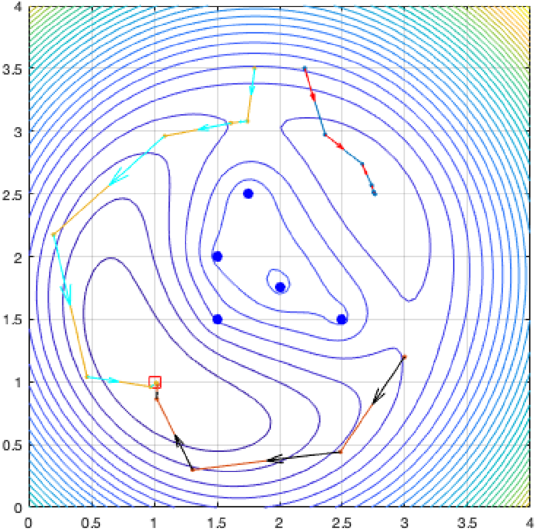
$$\hat{\Theta} = \hat{\theta}^{(\ell)} - (H^T H + \lambda^{(\ell)} I)^{-1} H^T f(\hat{\theta}^{(\ell)})$$

3. Set $\hat{\theta}^{(\ell+1)}$ and $\lambda^{(\ell+1)}$ as follows

$$\begin{cases} \hat{\theta}^{(\ell+1)} = \hat{\Theta}, \lambda^{(k+1)} = \beta_1 \lambda^{(k)} & \textit{if } \|f(\hat{\Theta})\|^2 \leq \|f(\hat{\theta}^{(\ell)})\|^2 \\ \hat{\theta}^{(\ell+1)} = \hat{\theta}^{(\ell)}, \lambda^{(k+1)} = \beta_2 \lambda^{(k)} & \textit{otherwise} \end{cases}$$

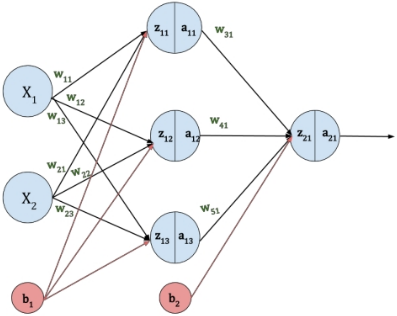
- β_1, β_2 are constants $0 < \beta_1 < 1 < \beta_2$
- Terminate if $\nabla g(\hat{\theta}^{(\ell)}) = 2H^T f(\hat{\theta}^{(\ell)})$ is small

Example: localization from range measurements



$\lambda^{(1)} = 0.1, \; \beta_1 = 0.8, \; \beta_2 = 2$

Model fitting using NN



A simple NN model.

First layer:

$$W^{[1]} = \begin{pmatrix} w_{11} & w_{21} \\ w_{12} & w_{22} \\ w_{13} & w_{23} \end{pmatrix} \quad b^{[1]} = \begin{pmatrix} b_1 \\ b_1 \\ b_1 \end{pmatrix}$$

Second layer:

$$W^{[2]} = \begin{pmatrix} w_{31} & w_{41} & w_{51} \end{pmatrix} \quad b^{[2]} = b_2$$

$$a_{21} = \sigma(\underbrace{W^{[2]}\,\underbrace{\sigma(\overbrace{W^{[1]}X + b^{[1]}}^{Z^{[1]}})}_{A^{[1]}} + b^{[2]}}_{Z^{[2]}})$$

$\triangleq f(\theta, X)$

θ: all the weights and biases

Backpropagation as gradient

We have data $(X(1), z(1)), \cdots, (X(k), z(k))$. Estimate θ to minimize $\sum_{i=1}^{k}(z(i) - f(\theta, X(i)))^2$.

- A NLSE problem.
- Requires $\nabla_\theta f(\theta, X)$

Example: $\frac{\partial f(\theta,X)}{\partial w_{12}}$

Backpropagation

$$\frac{\partial f(\theta, X)}{\partial w_{12}} = \frac{\partial \sigma(Z^{[2]})}{\partial Z^{[2]}} \frac{\partial Z^{[2]}}{\partial A^{[1]}} \frac{\partial A^{[1]}}{\partial Z^{[1]}} \frac{\partial Z^{[1]}}{\partial w_{12}}$$

Each partial derivative can be easily computed.