Computational Science:
Computational Methods in Engineering

Nonlinear Regression

https://empossible.net/academics/emp4301_5301/

Outline

- Statement of the Problem
- Multiple-Parameter Taylor Series
- Formulation
- Algorithm
- MATLAB Implementation
  https://empossible.net/academics/emp4301_5301/

Statement of Problem

Sometimes it is desired to fit a set of M measured data points to a nonlinear function $f(x)$.

$$y = f\left(x; a_0, a_1, \ldots, a_N\right)$$

$$y \equiv \text{measured value}$$
$$x \equiv \text{parameter from which } f \text{ is evaluated}$$
$$a_n \equiv \text{coefficients for the function fit}$$

The function can be anything like sine's, logarithms, exponentials…

$$f(x) = A + B\sin(Cx) \qquad f(x) = A + Be^{-Cx^2} \qquad f(x) = A + B\ln(Cx)$$

EMPossible 3

Multiple-Parameter Taylor Series

Recall the Taylor series for a single parameter…

$$f(x) = f(\tilde{x}) + \frac{f'(\tilde{x})}{1!}(x-\tilde{x}) + \frac{f''(\tilde{x})}{2!}(x-\tilde{x})^2 + \frac{f'''(\tilde{x})}{3!}(x-\tilde{x})^3 + \cdots$$

The two-parameter Taylor series is

$$f(x,y) = f(\tilde{x},\tilde{y}) + \frac{1}{1!}\left[\frac{\partial f(\tilde{x},\tilde{y})}{\partial x}(x-\tilde{x}) + \frac{\partial f(\tilde{x},\tilde{y})}{\partial y}(y-\tilde{y})\right]$$
$$+\frac{1}{2!}\left[\frac{\partial^2 f(\tilde{x},\tilde{y})}{\partial x^2}(x-\tilde{x})^2 + 2\frac{\partial^2 f(\tilde{x},\tilde{y})}{\partial x \partial y}(x-\tilde{x})(y-\tilde{y}) + \frac{\partial^2 f(\tilde{x},\tilde{y})}{\partial y^2}(y-\tilde{y})^2\right]$$
$$\vdots$$

Ignore the higher-order terms.

The N-parameter Taylor series using only first-order derivatives is

$$f(x_1, x_2, \ldots, x_N) \approx f(\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_N) + \sum_{n=1}^{N} \frac{\partial f}{\partial x_n}\Delta x_n \qquad \Delta x_n = x_n - \tilde{x}_n$$

EMPossible 4

Formulation

https://empossible.net/academics/emp4301_5301/

Slide 5

Solution Using Gauss-Newton Method (1 of 5)

Write an equation for each measured sample.

$$
\begin{aligned}
y_1 &= f(x_1; a_0, a_1, \ldots, a_N) + e_1 \\
y_2 &= f(x_2; a_0, a_1, \ldots, a_N) + e_2 \\
&\vdots \\
y_M &= f(x_M; a_0, a_1, \ldots, a_N) + e_M
\end{aligned}
\quad \rightarrow \quad
\underbrace{
\begin{aligned}
y_1 &= f(x_1) + e_1 \\
y_2 &= f(x_2) + e_2 \\
&\vdots \\
y_M &= f(x_M) + e_M
\end{aligned}
}_{\text{Shorthand notation}}
$$

This cannot be written in matrix form because $f(x)$ is nonlinear.

EMPossible

6

Solution Using Gauss-Newton Method (2 of 5)

Convert the nonlinear equations into linear equations by expanding them into multi-parameter Taylor series.

$$
\begin{aligned}
y_1 &= f(x_1) + \frac{\partial f(x_1)}{\partial a_0}\Delta a_0 + \frac{\partial f(x_1)}{\partial a_1}\Delta a_1 + \cdots + \frac{\partial f(x_1)}{\partial a_N}\Delta a_N + e_1 \\
y_2 &= f(x_2) + \frac{\partial f(x_2)}{\partial a_0}\Delta a_0 + \frac{\partial f(x_2)}{\partial a_1}\Delta a_1 + \cdots + \frac{\partial f(x_2)}{\partial a_N}\Delta a_N + e_2 \\
&\vdots \\
y_M &= f(x_M) + \frac{\partial f(x_M)}{\partial a_0}\Delta a_0 + \frac{\partial f(x_M)}{\partial a_1}\Delta a_1 + \cdots + \frac{\partial f(x_M)}{\partial a_N}\Delta a_N + e_M
\end{aligned}
$$

EMPossible 7

Solution Using Gauss-Newton Method (3 of 5)

Now that the equations are linear, write them in matrix form.

$$
\begin{aligned}
y_1 &= f(x_1) + \frac{\partial f(x_1)}{\partial a_0}\Delta a_0 + \frac{\partial f(x_1)}{\partial a_1}\Delta a_1 + \cdots + \frac{\partial f(x_1)}{\partial a_N}\Delta a_N + e_1 \\
y_2 &= f(x_2) + \frac{\partial f(x_2)}{\partial a_0}\Delta a_0 + \frac{\partial f(x_2)}{\partial a_1}\Delta a_1 + \cdots + \frac{\partial f(x_2)}{\partial a_N}\Delta a_N + e_2 \\
&\vdots \\
y_M &= f(x_M) + \frac{\partial f(x_M)}{\partial a_0}\Delta a_0 + \frac{\partial f(x_M)}{\partial a_1}\Delta a_1 + \cdots + \frac{\partial f(x_M)}{\partial a_N}\Delta a_N + e_M
\end{aligned}
\Rightarrow
$$

$$
\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_{M-1} \\ y_M \end{bmatrix}
=
\begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \\ \vdots \\ f(x_{M-1}) \\ f(x_M) \end{bmatrix}
+
\begin{bmatrix}
\frac{\partial f(x_1)}{\partial a_0} & \frac{\partial f(x_1)}{\partial a_1} & \cdots & \frac{\partial f(x_1)}{\partial a_N} \\
\frac{\partial f(x_2)}{\partial a_0} & \frac{\partial f(x_2)}{\partial a_1} & \cdots & \frac{\partial f(x_2)}{\partial a_N} \\
\frac{\partial f(x_3)}{\partial a_0} & \frac{\partial f(x_3)}{\partial a_1} & \cdots & \frac{\partial f(x_3)}{\partial a_N} \\
\vdots & \vdots & & \\
\frac{\partial f(x_{M-1})}{\partial a_0} & \frac{\partial f(x_{M-1})}{\partial a_1} & \cdots & \frac{\partial f(x_{M-1})}{\partial a_N} \\
\frac{\partial f(x_M)}{\partial a_0} & \frac{\partial f(x_M)}{\partial a_1} & \cdots & \frac{\partial f(x_M)}{\partial a_N}
\end{bmatrix}
\begin{bmatrix} \Delta a_0 \\ \Delta a_1 \\ \vdots \\ \Delta a_N \end{bmatrix}
+
\begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_{M-1} \\ e_M \end{bmatrix}
$$

$$[y] = [f] + [Z][\Delta a] + [e]$$

EMPossible 8

Solution Using Gauss-Newton Method (4 of 5)

Solve the matrix equation for $[\Delta a]$ using least-squares.

$[y]=[f]+[Z][\Delta a]+\cancel{[e]}$	Drop the residual term since fit will be iterated.
$[y]-[f]=[Z][\Delta a]$	Bring $[f]$ to left-hand side of equation.
$[d]=[Z][\Delta a]$	Let $[d]=[y]-[f]$
$[Z]^T[d]=[Z]^T[Z][\Delta a]$	Follow least-squares recipe and premultiply by $[Z]^T$.
$[\Delta a]=\left([Z]^T[Z]\right)^{-1}[Z]^T[d]$	Solve for $[\Delta a]$.

$[d]$ is a new error function.
$[y]$ contains the measured values and $[f]$ contains the fit values from the function $f(x)$.

EMPossible

9

Solution Using Gauss-Newton Method (5 of 5)

The equation to calculate $[\Delta a]$ is

$$[\Delta a]=\left([Z]^T[Z]\right)^{-1}[Z]^T[d]$$

This will only convey how much to adjust the coefficients $[a]$ given an initial guess $[a]_0$.

The calculation must be iterated several times to find the actual coefficients of $[a]$.

EMPossible

10

Algorithm

https://empossible.net/academics/emp4301_5301/

Slide 11

Algorithm for Nonlinear Regression (1 of 2)

Step 0 – Derive analytical expressions for partial derivatives of $f(x)$.

$$\frac{\partial f(x)}{\partial a_0}, \frac{\partial f(x)}{\partial a_1}, \cdots \frac{\partial f(x)}{\partial a_N}$$

Step 1 – Make an initial guess at the coefficients.

$$[a]_0 = \begin{bmatrix} a_{0,0} & a_{1,0} & \cdots & a_{N,0} \end{bmatrix}^T$$

Make an intelligent guess! This algorithm is sensitive!

Step 2 – Evaluate the function $[f]_i$ at all measured points given the current value of the coefficients $[a]_i$.

$$[f]_i = \begin{bmatrix} f_i(x_1) & f_i(x_2) & f_i(x_3) & \cdots & f_i(x_{M-1}) & f_i(x_M) \end{bmatrix}^T$$

Step 3 – Calculate the error $[d]_i$ in the estimate of $[f]_i$.

$$[d]_i = [y] - [f]_i$$

EMPossible

12

Algorithm for Nonlinear Regression (2 of 2)

Step 4 – Construct the $[Z]_i$ matrix given the current coefficients $[a]_i$.

$$[Z]_i = \begin{bmatrix} \frac{\partial f_i(x_1)}{\partial a_{0,i}} & \frac{\partial f_i(x_1)}{\partial a_{1,i}} & \cdots & \frac{\partial f_i(x_1)}{\partial a_{N,i}} \\ \frac{\partial f_i(x_2)}{\partial a_{0,i}} & \frac{\partial f_i(x_2)}{\partial a_{1,i}} & \cdots & \frac{\partial f_i(x_2)}{\partial a_{N,i}} \\ \vdots & \vdots & \ddots & \\ \frac{\partial f_i(x_M)}{\partial a_{0,i}} & \frac{\partial f_i(x_M)}{\partial a_{1,i}} & & \frac{\partial f_i(x_M)}{\partial a_{N,i}} \end{bmatrix}$$

It is necessary to derive $N+1$ derivatives, one for each coefficient a_n.

It is necessary to evaluate each of these $N+1$ derivatives at all M points.

This is a lot of work!

Step 5 – Solve for $[\Delta a]_i$ using least squares.

$$[\Delta a]_i = \left([Z]_i^T [Z]_i\right)^{-1} \left([Z]_i^T [d]_i\right)$$

EMPossible

13

Algorithm for Nonlinear Regression (3 of 3)

Step 6 – Adjust $[a]_i$ using $[\Delta a]_i$.

$$[a]_{i+1} = [a]_i + [\Delta a]_i$$

Aside

Sometimes the standard nonlinear regression algorithm will oscillate and not converge. In many cases this can be mitigated by introducing a convergence rate parameter α. The adjustment equation becomes

$$[a]_{i+1} = [a]_i + \alpha [\Delta a]_i$$

$\alpha = 1$ is the standard nonlinear regression. Try values in the range $0.1 \leq \alpha \leq 0.5$, but always be willing to experiment with other values.

Step 7 – Go back to Step 2 until converged.

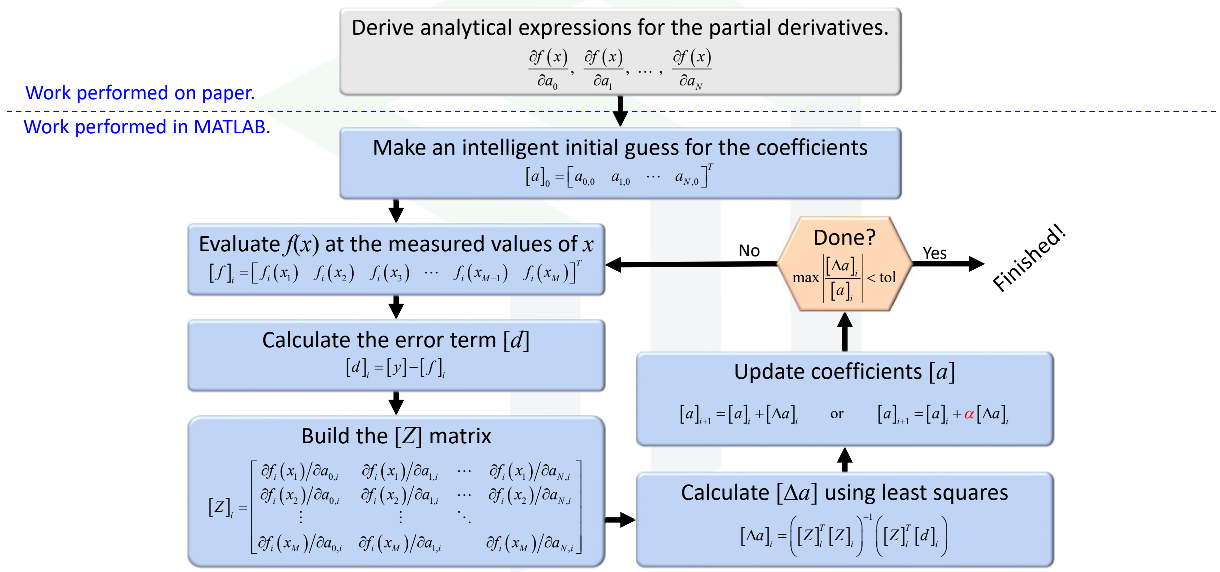
Convergence happens when the change in coefficient values $[\Delta a]_i$ falls below some tolerance.

$$\left| \frac{a_{n,i+1} - a_{n,i}}{a_{n,i+1}} \right| \cdot 100\% \leq \text{tol} \qquad \text{for all } a_n$$

EMPossible

14

Block Diagram



EMPossible

Slide 15

Notes for Nonlinear Regression

- Method can be used to fit any set of measured data to any function who's first-derivative exists.
- This method does not always converge.
  Try different initial guesses and/or introduce a convergence rate parameter α.
- Use the most intelligent initial guess as possible.
- Linear regression is a special case of nonlinear regression.

EMPossible

16