Uncertainty analysis in Larch (or IFEFFIT? haven't checked.)

Implementation used in Larch

- Larch uses lmfit package for the Levenberg-Marquardt algorithm.
- lmfit calls Scipy.optimize.leastsq() to call MINPACK's lmdif() and lmder() functions.
- The hessian matrix return by leastsq() is calculated using an approximate hessian matrix calculated by

$$\mathbf{H} = \mathbf{J}(\mathbf{x})^T \mathbf{J}(\mathbf{x})$$

- The covariance matrix is calculated by

$$\Sigma = \left(\mathbf{J}^T \mathbf{J}\right)^{-1} = \mathbf{P}\left(\mathbf{R}^T \mathbf{R}\right)^{-1} \mathbf{P}$$

where $\mathbf{P}$ is the permutation matrix and $\mathbf{R}$ is the upper triangular matrix obtained from the QR decomposition of $\mathbf{J}$.

Maximum likelihood estimation

- If we assume that we have a distribution function of $\mathbf{y}$ by a model $\mathbf{F}(\mathbf{x})$ with parameters θ as f, then the likelihood function is given by

$$L(\theta) = \prod_{i=0}^{n} f_i(\mathbf{y}|\theta)$$

the logaritmic likelihood function is given by

$$l(\theta) = \sum_{i=0}^{n} \log f_i(\mathbf{y}|\theta)$$

The first derivative is the score function.

$$\mathbf{s}(\theta) = \begin{pmatrix} \frac{\partial l(\theta)}{\partial \theta_1} \\ \frac{\partial l(\theta)}{\partial \theta_2} \\ \vdots \\ \frac{\partial l(\theta)}{\partial \theta_n} \end{pmatrix}$$

maximization of the likelihood function is equivalent to calculating a zero of the score function.

$$\mathbf{s}(\theta) = \mathbf{0}$$

For a nessesary and sufficient condition for a maximum, the Hessian matrix $\mathbf{H}$ of $l(\theta)$ must be negative definite.

if we take the second derivative of $l(\theta)$ with respect to θ, we get the Hessian matrix $\mathbf{H}$ of $l(\theta)$.

$$\mathbf{H} = \begin{pmatrix} \frac{\partial^2 l(\theta)}{\partial \theta_1^2} & \frac{\partial^2 l(\theta)}{\partial \theta_1 \partial \theta_2} & \cdots & \frac{\partial^2 l(\theta)}{\partial \theta_1 \partial \theta_n} \\ \frac{\partial^2 l(\theta)}{\partial \theta_2 \partial \theta_1} & \frac{\partial^2 l(\theta)}{\partial \theta_2^2} & \cdots & \frac{\partial^2 l(\theta)}{\partial \theta_2 \partial \theta_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 l(\theta)}{\partial \theta_n \partial \theta_1} & \frac{\partial^2 l(\theta)}{\partial \theta_n \partial \theta_2} & \cdots & \frac{\partial^2 l(\theta)}{\partial \theta_n^2} \end{pmatrix}$$