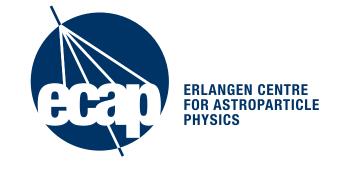
# Levenberg-Marquardt Minimization

Jörn Wilms Remeis-Sternwarte & ECAP Universität Erlangen-Nürnberg

http://pulsar.sternwarte.uni-erlangen.de/wilms/ joern.wilms@sternwarte.uni-erlangen.de





### Introduction

#### The fitting problem:

Given the linear model

$$N_{\text{ph, predicted}}(c; \rho) = \Delta T \cdot \sum_{i=0}^{n_{\text{ch}}} A(E_i) \cdot R(c, i) \cdot M(E_i; \rho) \cdot \Delta E_i + N_{\text{background}}(c)$$

$$\forall c \in \{1, 2, \dots, n_{\text{en}}\} \ (1)$$

where  $M(E_i; \rho)$  is the spectral model, which depends on parameters  $\mathbf{x}$ , and

• given a fit statistics

$$S^{2}(\mathbf{x}) = f(N_{\text{ph, measured}}, N_{\text{ph, predicted}}(\mathbf{x}))$$
 (2)

what is the **x** of the "most likely" mode.

→ Minimization problem

#### Minimization Methods

Take the residual vector at a value x:

$$\mathbf{R} = \frac{\text{computed} - \text{observed}}{\sigma} \tag{3}$$

Change as one varies **x**:

$$\boldsymbol{\beta} = \mathbf{J}^{\mathsf{T}} \mathbf{R} \quad \text{where} \quad \mathbf{J} = \frac{\mathrm{d}\mathbf{R}}{\mathrm{d}\mathbf{x}}$$
 (4)

J: Jacobi matrix.

Change parameters by

$$\Delta \mathbf{x} = -\alpha^{-1} \mathbf{\beta} \tag{5}$$

where the curvature matrix is

$$\alpha = \mathbf{J}^{\mathsf{T}}\mathbf{J} \tag{6}$$

Then iterate until convergence.

# Levenberg-Marquardt

Levenberg-Marquardt-method (LM-Method): Numerical method to minimize fit statistics

Levenberg, 1944, Q. Appl. Math 2, 164, Marquardt, 1963, SIAM J. Appl. Math. 11, 431

Modify simple gradient minimization by damping: Replace  $\alpha$  with  $\alpha'$  where

$$\alpha'_{jj} = \begin{cases} \alpha_{jj}(1+\lambda) & \text{multiplicative damping} \\ \alpha_{jj} + \lambda & \text{additive damping} \end{cases}$$
(7)

i.e., the damped curvature matrix is (add. damping):

$$\alpha + \lambda \mathbb{1}$$
 (8)

LM: adjust  $\lambda$ : Initially: want  $\alpha \sim$  gradient, to lie in direction of steepest descent.

for additive damping: shrinks  $\Delta p \Longrightarrow$  stablizing iteration by preconditioning the matrix  $\alpha$ ; in contrast, multiplicative damping will help the method against badly scaled problems

# Levenberg-Marquardt

LM algorithm: decide how to change  $\lambda$  between steps:

Depending on how sum of squares (SOS) behaves at  $\mathbf{x} + \Delta \mathbf{x}$ :

- SOS decreased:  $\lambda' = \lambda \cdot DROP$
- SOS increased: drop p', set  $\lambda' = \lambda \cdot BOOST$

#### Note:

1. Assumes sum of squares-like likelihood ML won't work well on C-stat!!

2. Efficiency depends on values of BOOST and DROP.

Lampton (1997, Computers in Physics 11, 110): additive, DROP = 0.1, BOOST = 1.5 is well behaved, but not always

XSPEC does not allow changing these parameters, ISIS does