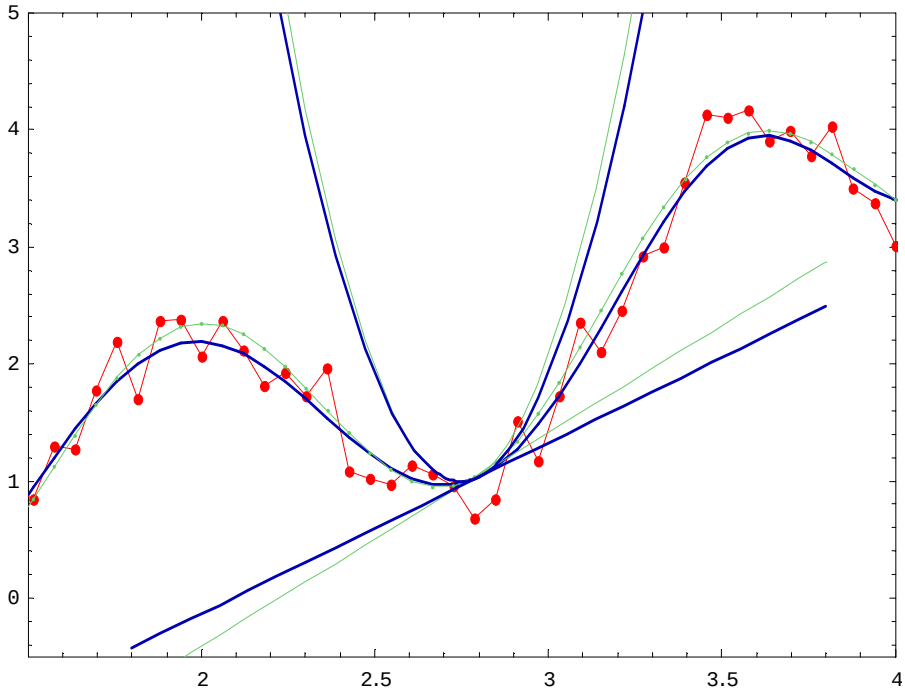
Linear Approximation with Regularization and Moving Least Squares

Igor Grešovnik
May 2007

Revision 4.6
(Revision 1: March 2004).



Contents:

Change of notation:

$m \to N_v$ - number of points where approximated function is evaluated

$m \to N_g$ - number of points where gradient of the approximated function is evaluated

$n \to N_b$ - number of basis functions

Use of indices:

k – index of sampling points

i, *j* – indices of components of approximation coefficients, components of right-hand side vector and components of the system matrix in systems of equations

l, *m* - components of co-ordinate derivatives

t – components of gradients of the sampled (approximated) function.

1 LINEAR FITTING

1.1 Weighted Least Squares in Function Approximation

We have values of some function $f(\mathbf{x})$, $\mathbf{x} \in \mathrm{R}^N$, in N_v points:

$$f(\mathbf{x}_k) = y_k \ , \ k = 1, ..., N_v \, . \tag{1}$$

We would like to evaluate coefficients of linear combination of N_b functions $f_1(\mathbf{x})$, …, $f_n(\mathbf{x})$

$$\tilde{f}(\mathbf{x}; \mathbf{a}) = a_1 f_1(\mathbf{x}) + a_2 f_2(\mathbf{x}) + ... + a_n f_n(\mathbf{x}) = \sum_{j=1}^{N_b} a_j f_j(\mathbf{x}), \tag{2}$$

such that

$$\tilde{f}(\mathbf{x}_k) \approx f(\mathbf{x}_k) = y_k \quad \forall k = 1, ..., N_v \, , \tag{3}$$

i.e. we want that the linear approximation (or approximation) agrees as much as possible with values of $f(\mathbf{x})$ in all points $\mathbf{x}_k$. We look for the best agreement in the weighted least squares sense, i.e. we minimize the function

$$\chi^2(\mathbf{a}) = \phi(\mathbf{a}) = \sum_{k=1}^{Nv} w_k^2 \left(y(\mathbf{x}_k) - y_k \right)^2 = \sum_{k=1}^{Nv} w_k^2 \left(\left(\sum_{j=1}^{Nb} a_j f_j(\mathbf{x}_k) \right) - y_k \right)^2 . \tag{4}$$

with respect to parameters of approximation a_i. $\mathbf{w}$ is the *m*-dimensional vector of weights, which weight significance of points $\mathbf{x}_i$. Minimum is the stationary point of $\phi(\mathbf{a})$ where

$$\frac{d\,\phi(\mathbf{a})}{d\,a_i} = 0 \quad \forall\, i = 1, \ldots, N_b \,. \tag{5}$$

Derivatives of $\phi(\mathbf{a})$ are

$$\frac{d\,\phi(\mathbf{a})}{d\,a_i} = 2\sum_{k=1}^{Nv}\left(w_k^2 \left(\sum_{j=1}^{Nb} a_j\, f_j(\mathbf{x}_k) - y_k \right) f_i(\mathbf{x}_k) \right) \tag{6}$$

Equation (5) therefore gives the following system of equations for unknown coefficients a_j:

$$\sum_{j=1}^{Nb} a_j \sum_{k=1}^{Nv} \left(w_k^2\, f_j(\mathbf{x}_k)\, f_i(\mathbf{x}_k) \right) = \sum_{k=1}^{Nv} \left(w_k^2 y_k\ f_i(\mathbf{x}_k) \right),\ i = 1, \ldots, n \tag{7}$$

Coefficients **a** can therefore be obtained by solving the linear system of equations

$$\mathbf{C}\mathbf{a} = \mathbf{d}\,, \tag{8}$$

where

$$C_{ij} = \sum_{k=1}^{Nv} w_k{}^2\, f_i(\mathbf{x}_k)\, f_j(\mathbf{x}_k) \tag{9}$$

and

$$d_i = \sum_{k=1}^{Nv} w_k{}^2\, f_i(\mathbf{x}_k)\, y_k \tag{10}$$

The system of equations (8) for calculation of approximation coefficients is called a *normal system of equations*. It can be shown that **C** is positive-semidefinite. If **C** has a full rank *n* then it is *positive-definite*, and the system can be solved by the Cholesky factorization

$$\mathbf{C} = \mathbf{V}^T\mathbf{V}\,. \tag{11}$$

(where **V** is upper triangular) followed by the solutions of a lower triangular system,

$$\mathbf{V}^T\,\mathbf{y} = \mathbf{d}\ , \tag{12}$$

and an upper triangular system,

$$\mathbf{V}\,\mathbf{a} = \mathbf{y}\ . \tag{13}$$

1.1.1 Solution of over-determined system of equations by QR decomposition

Here we point at the relation between the least squares formulation (4), (8)and direct solution of the over-determined system of equations (3).

We can write introduce matrix **A** and vector **b** such that

$$A_{kj} = w_k f_j(\mathbf{x}_k) \tag{14}$$

and

$$\mathbf{b}_k = w_k y_k . \tag{15}$$

Then the equation

$$\mathbf{A}_{(m\times n)} \tilde{\mathbf{a}}_{(n\times 1)} = \mathbf{b}_{(m\times 1)} \tag{16}$$

reads component-wise as

$$\forall k, \sum_{j=1}^{Nb} w_k f_j(\mathbf{x}_k) \tilde{a}_j = w_k y_k$$

or

$$\forall k,\ w_k \sum_{j=1}^{Nb} \tilde{a}_j f_j(\mathbf{x}_k) = w_k y_k\ , \tag{17}$$

which is exactly (3), if we take into account (2) and denote coefficients by $\tilde{a}_j$ instead of a_j.

Equation (16) (or (17) in component-wise notation) is an over-determined system, therefore we can not divide both sides of each equation by w_k because the this would affect the significance of individual equations and therefore the solution (the system in this case does not have an exact solution and therefore the relative significance of equations is important).

Now, we can show that the system (16) is in some sense equivalent to the system (8). This is seen by observing that

$$\begin{aligned} \mathbf{C} &= \mathbf{A}^T\mathbf{A} \\ \mathbf{d} &= \mathbf{A}^T\mathbf{b} \end{aligned}\ , \tag{18}$$

i.e. the least squares system of equations (8) (also referred to as the *normal system of equations*) is obtained by left multiplying the over-determined system (16) by $\mathbf{A}^T$.

It can be shown that we can obtain the solution of the normal system (8) by performing the QR decomposition of the matrix **A** form the system (16) [1]:

$$\underline{\underline{\mathbf{A}_{(Nv\times Nb)} = \mathbf{Q}_{(Nv\times Nv)}\,\mathbf{U}_{Nv\times Nb}}} \tag{19}$$

We denote by **z** solution of the orthogonal system $\mathbf{Q}\,\mathbf{z} = \mathbf{b}$, i.e.

$$\underline{\underline{\mathbf{z} = \mathbf{Q}^T\,\mathbf{b}}}\ , \tag{20}$$

Matrix **U** is *upper trapezoid* (by the QR decomposition). We write **U** and **z** in block form,

$$\mathbf{U}_{(Nv\times Nb)} = \begin{bmatrix} \mathbf{V}_{(Nb\times Nb)} \\ \mathbf{0}_{((Nv-Nb)\times Nb)} \end{bmatrix} \quad \left(v_{ik} = 0, i > k\right)\ , \tag{21}$$

$$\mathbf{z}_{(Nv\times 1)} = \begin{bmatrix} \mathbf{y}_{(Nb)} \\ \mathbf{w}_{(Nv-Nb)} \end{bmatrix}\ . \tag{22}$$

Now it follows from $\mathbf{C} = \mathbf{A}^T\mathbf{A}$ (taking into account the decomposition and the block form) that

$$\mathbf{C} = \mathbf{A}^T\mathbf{A} = \mathbf{V}^T\mathbf{V}\ ,$$

i.e. <u>**V** is a Cholesky factor of the normal matrix $\mathbf{C} = \mathbf{A}^T\mathbf{A}$</u>. With QR factorization, we have avoided calculation of **C** and its Cholesky factorization. We can further verify that

$$\mathbf{d} = \mathbf{A}^T\,\mathbf{b} = \mathbf{V}^T\,\mathbf{y}\ . \tag{23}$$

This means that **y**, which is the upper part of the transformed **z**, is the solution of the lower triangular system (12).

The least squares solution is therefore obtained (according to (13)) by solving the least squares system

$$\underline{\underline{\mathbf{V}\,\tilde{\mathbf{a}} = \mathbf{y}}}\ . \tag{24}$$

Advantage of using the QR factorization is that the matrix **A** is better conditioned than $\mathbf{A}\mathbf{A}^T$. If spectral sensitivity of matrix **A** equals σ_1/σ_n, then the spectral sensitivity of $\mathbf{A}\mathbf{A}^T = \mathbf{B}$ is $\left(\sigma_1/\sigma_n\right)^2$. In this expression, σ_1^2 is the largest and σ_n^2 the smallest eigenvalue of **B**.

==========

From (2) we can see that

$$y\left(\mathbf{x}_i, \mathbf{a}\right) = \sum_{k=1}^{Nb} a_k\, f_k\left(\mathbf{x}_i\right). \tag{25}$$

and therefore

$$\frac{d\, y(\mathbf{x}_i, \mathbf{a})}{d\, a_k} = f_k(\mathbf{x}_i) = \frac{A_{ik}}{w_i} \tag{26}$$

We see that

$$\frac{A_{ik}}{w_i} = \frac{d\, y(\mathbf{x}_i, \mathbf{a})}{d\, a_k} \tag{27}$$

Sometimes we define matrix $\mathbf{X}$ so that

$$X_{ij} = \frac{d\, y(x_j, \mathbf{a})}{d\, a_i} = f_i(\mathbf{x}_j) = \sigma_j A_{ji}\,. \tag{28}$$

1.1.2 Statistical background

We have a model that predicts a set of measurements (observations) y_i, which is dependent on a set of unknown parameters a_i:

$$y_i(\mathbf{a})\,. \tag{29}$$

In function approximation, we have a model for a function of one or a set of independent variables,

$$y_i(\mathbf{a}) = y(\mathbf{x}_i; \mathbf{a}) \tag{30}$$

From the point of view of parameter estimation, this is the same as (29) because independent variables $\mathbf{x}_i$ are used just to distinguish between distinct measurements (to index the measurements, the same as index i in (29)), and actual functional relations are not actually used. In the least squares formulation, parameters $\mathbf{a}$ are estimated by minimizing the sum of squares,

$$\min_{\mathbf{a}} \chi^2(\mathbf{a}) = \sum_{i=1}^{Nv} \left(\frac{\left(y_i - y_i(\mathbf{a})\right)}{\sigma_i} \right)^2\,. \tag{31}$$

Note that for linear models,

$$y_k(\mathbf{a}) = \sum_{j=1}^{Nb} a_j\ f_{jk} \tag{32}$$

or in function approximation,

$$y(\mathbf{x};\mathbf{a}) = \sum_{j=1}^{Nb} a_j \, f_j(\mathbf{x}) \tag{33}$$

Both forms are equivalent, which can be easily seen if we writhe the second form (33) for $\mathbf{x} = \mathbf{x}_i$.

1.1.2.1 Statistical background

The statistical background described here applied for general least squares fitting (also nonlinear). For founding the least squares procedures, we must assume that measurement errors are independently random and normally distributed:

$$y_i \sim \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y_i-\mu_i}{\sigma_i}\right)^2} . \tag{34}$$

When fitting the parameters, we would like to find the parameters that are "most likely" to be correct. It is *not meaningful to ask* e.g. "What is the probability that given parameters **a** are correct"[1]. However, the intuition tells us that the parameters for which the model data doesn't look like the measured data are unlikely.

We can ask the question "Given the particular set of parameters, what is the probability that the specific data set could have occurred?" If y_i take continuous values then we must say "the probability that $y_i \pm \Delta y$ occur". If this probability is very small then we conclude that the parameter under consideration are "unlikely" to be right. Conversely, the intuition tells us that the data should not be too unlikely to improbable for the correct model parameters.

In other words, we intuitively identify the probability of data given the parameters, as the likelihood of the parameters given the data. This is based on intuition and ahs no mathematical background!

We look for parameters that maximize the likelihood defined in the above way, and this form of estimation is the *maximum likelihood estimation*.

According to assumption (34), the probability of the data set is the product of probabilities for individual data points:

$$P = \prod_{i=1}^{m} \exp\left(-\frac{1}{2}\left(\frac{y_i(\mathbf{a}) - y_i}{\sigma_i} \right)^2 \right) \Delta y \tag{35}$$

Maximizing this probability is equivalent to minimizing negative of its logarithm,

$$\sum_{i=1}^{m} \frac{\left(y_i(\mathbf{a}) - y_i\right)^2}{2\sigma^2} - m \ln \Delta y \ .$$

[1] The point is that there is just one model – the correct one, and there is a statistical universe of data sets that are drawn from that model.

Since the last term is constant, minimizing the equation is equivalent to minimizing (31).

Remark: The discussion is limited to *statistical errors*, which we can average away (in a desired extent) if we take enough data. Measurements are also susceptible *systematic errors*, which can not be annihilated by any amount of averaging[2].

In equations (8)-(10) and (14)-(16), regarding statistical argumentation, we must set the weights to

$$w_i = \frac{1}{\sigma_i} \ . \tag{36}$$

Let us now estimate the uncertainties of the estimated parameters. The variance associated with a_j can be found from

$$\sigma^2\left(a_j\right) = \sum_{i=1}^{m} \sigma_i^2 \left(\frac{\partial a_j}{\partial y_i} \right)^2 \ . \tag{37}$$

from (8) we have

$$a_j = \sum_{k=1}^{n} \left[\mathbf{C}^{-1} \right]_{jk} d_k = \sum_{k=1}^{m} \left[\mathbf{C}^{-1} \right]_{jk} \left[\sum_{i=1}^{m} \frac{y_i f_k\left(\mathbf{x}_i\right)}{\sigma_i^2} \right] \tag{38}$$

Since C_{jk} is independent of y_i,

$$\frac{\partial a_j}{\partial y_i} = \sum_{k=1} \left[\mathbf{C}^{-1} \right]_{jk} \frac{f_k\left(x_i\right)}{\sigma_i^2} \ . \tag{39}$$

We write $\mathbf{C}^{-1} = \mathbf{W}$ Consequently,

$$\sigma^2\left(a_j\right) = \sum_{k=1}^{n} \sum_{l=1}^{n} W_{jk} W_{jl} \left(\sum_{i=1}^{m} \frac{f_k\left(x_i\right) f_l\left(x_i\right)}{\sigma_i^2} \right) . \tag{40}$$

The final term in brackets in the above equation is just **C**. Since this is inverse of W, the equation reduces do W_{jj}, i.e.

$$\underline{\sigma^2\left(a_j\right) = \left[\mathbf{C}^{-1} \right]_{jj}} \ . \tag{41}$$

Off diagonal elements of $\mathbf{C}^{-1}$ are *covariances* between a_j and a_k:

$$\underline{Cov\left(a_j, a_k\right) = \left[\mathbf{C}^{-1} \right]_{jk}} \tag{42}$$

[2] E.g. calibration of a measurement equipment can depend on the temperature, and if we perform all measurements at a wrong temperature then averaging will not reduce the systematic error.

1.1.2.2 Non-normal distribution of errors

In the case of non-normal errors, we often do the following things that are derived from the assumption that the error distribution is normal:

- Fit parameters by minimizing χ^2
- Use contours of constant $\Delta\chi^2$ as the boundary of the confidence region
- Use Monte Carlo simulations or analytical calculations to determine which contour of $\Delta\chi^2$ is the correct one for the desired confidence level
- Give the covariance matrix **C** as the "formal covariance matrix of the fit on the assumption of normally distributed errors"
- Interpret C_{ij} as the actual squared standard errors of the parameter estiomation

1.2 Weighted Least Squares Approximation of Function Values and Gradients

Sometimes we have gradient information beside the values of a function in a given set of points, and we want to construct an approximation that best fits the specified values and the gradients. In this section equations are derived for approximations that consider both value and gradient data.

We have values of some function $f(\mathbf{x})$ and its gradients in *m* points:

$$f(\mathbf{x}_k) = y_k \quad \nabla f\left(\mathbf{x}_{k_g}\right) = \mathbf{g}_{k_g}, \quad i = 1, ..., N_v. \tag{43}$$

We would like to evaluate coefficients of linear combination of N_b functions $f_1(\mathbf{x}), \ldots, f_{N_b}(\mathbf{x})$

$$\tilde{f}(\mathbf{x}) = a_1 f_1(\mathbf{x}) + a_2 f_2(\mathbf{x}) + ... + a_n f_n(\mathbf{x}) = \sum_{j=1}^{Nb} a_j f_j(\mathbf{x}), \tag{44}$$

such that

$$\tilde{f}(\mathbf{x}_k) \approx y_k = f(\mathbf{x}_k) \quad \forall k = 1, ..., N_v \ \wedge \ \nabla \tilde{f}\left(\mathbf{x}_{k_g}\right) \approx \mathbf{g}_{k_g} = \nabla f\left(\mathbf{x}_{k_g}\right) \quad \forall k_g = 1, ..., N_g, \tag{45}$$

In order to keep the generality of derivation, we will allow throughout the text that values and gradients of the approximated function $f(\mathbf{x})$ are evaluated in different sets of points (which may however partially or fully coincide). We will denote the *k*-th component of $\mathbf{g}_i$ by g_{ik}. The gradient of the approximation is simply

$$\nabla f(\mathbf{x}) = \sum_{j=1}^{Nb} a_j \nabla f_j(\mathbf{x}) \tag{46}$$

We want that the linear approximation (or approximation) agrees as much as possible with values of $f(\mathbf{x})$ and that its gradient agrees as much as possible with gradients of $f(\mathbf{x})$ in all points $\mathbf{x}_i$. We look for the best agreement in the weighted least squares sense, i.e. we minimize the function

$$\begin{aligned}\phi(\mathbf{a}) &= \sum_{k=1}^{Nv}\left(w_k^2\left(y(\mathbf{x}_k)-y_k\right)^2\right)+\sum_{k_g=1}^{Ng}\left(\sum_{t=1}^{N} w_{k_g t}{}^2\left(\frac{\partial y}{\partial x_t}\left(\mathbf{x}_{k_g}\right)-g_{k_g t}\right)^2\right)= \\ &\sum_{k=1}^{Nv}\left(w_k^2\left(\left(\sum_{j=1}^{Nb} a_j f_j(\mathbf{x}_k)\right)-y_k\right)^2\right)+\sum_{k_g=1}^{N_g}\left(\sum_{t=1}^{N} w_{k_g t}{}^2\left(\sum_{j=1}^{Nb}\left(a_j\frac{\partial f_j}{\partial x_t}\left(\mathbf{x}_{k_g}\right)\right)-g_{k_g t}\right)^2\right)\end{aligned} . \tag{47}$$

with respect to parameters of approximation a_i. w_k are N_v weights that weigh significance of values in points $\mathbf{x}_k$ and $w_{k_g l}$ w_k are $N_g \cdot N$ weights that weigh significance of individual gradient components in $\mathbf{x}_{k_g}$. Minimum is the stationary point of $\phi(\mathbf{a})$ where

$$\frac{d\,\phi(\mathbf{a})}{d\,a_i} = 0 \;\; \forall i = 1, \ldots, N_b . \tag{48}$$

Derivatives of $\phi(\mathbf{a})$ are

$$\begin{aligned}\frac{d\,\phi(\mathbf{a})}{d\,a_i} &= 2\sum_{k=1}^{Nv}\left(w_k^2\left(\sum_{j=1}^{Nb} a_j f_j(\mathbf{x}_k)-y_k\right) f_i(\mathbf{x}_k)\right)+ \\ &+2\sum_{k_g=1}^{N_g}\sum_{t=1}^{N}\left(w_{k_g l}{}^2\left(\sum_{j=1}^{Nb} a_j\frac{\partial f_j}{\partial x_t}\left(\mathbf{x}_{k_g}\right)-g_{k_g t}\right)\frac{\partial f_i}{\partial x_t}\left(\mathbf{x}_{k_g}\right)\right)\end{aligned} \tag{49}$$

Equation (48) therefore gives the following system of equations for unknown coefficients a_j: