JOURNAL OF CHEMOMETRICS
J. Chemometrics 2000; **14**: 667–692

Non-linear regression analysis: new approach to traditional implementations

E. V. Bystritskaya, A. L. Pomerantsev* and O. Ye. Rodionova

Institute of Chemical Physics RAS, 4 Kosygin Street, 117977, Moscow, Russia

SUMMARY

Non-linear regression (NLR) analysis in chemometric applications is the main subject of the paper. The following novel items of NLR procedure are reported. The modification of gradient method is considered. For inversion of the Fisher matrix the recurrence algorithm based on the matrix exponential is used. A new method of sequential Bayesian estimation allows processing of the data successively for every response. Each data set is fitted individually, but taking into account the information about common parameters estimated on previous data sets. A posterior Bayesian distribution is built after every set processing. A new method of confidence estimation is suggested. Unlike bootstrap, not initial data but parameter estimates are simulated. This method has the same accuracy as bootstrap but is about 1000 times faster. A new coefficient of non-linearity is introduced. It is calculated by the Monte Carlo procedure and accounts for the model structure as well as the experimental design features. All new ideas were implemented in the software FITTER, a new Excel Add-In. Its main capabilities are reported. The paper is illustrated with a number of practical examples in DSC, TMA and TGA data analysis.

KEY WORDS: non-linear regression; confidence intervals; coefficient of non-linearity; FITTER Add-In for Excel; models for dynamic methods with variable temperature

1. INTRODUCTION

Every chemometrician knows what non-linear regression (NLR) analysis is, but few apply it in practice. Just compare the numerous papers about linear regression analysis and all its variations (PCA, PLS) with the few papers about NLR in chemometrics journals [1]. NLR provides a real chance to predict the behaviour of a complex chemical system in a region of setting that is far from the observed one. In fact, it enables us to use sophisticated 'hard' models for extrapolation. What are the circumstances that hinder the practical use of NLR? Let us consider the differences between linear and non-linear regression in order to understand the problem.

A linear model is an equation

$$f = a_1x_1 + a_2x_2 + \ldots + a_px_p$$

* Correspondence to: A. L. Pomerantsev, Institute of Chemical Physics RAS, 4 Kosygin Street, 117977 Moscow, Russia. E-mail: rcs@chph.ras.ru

Received 18 August 1999
Accepted 20 March 2000

where a_i are unknown parameters and x_i are known predictors or functions of predictors. It is essential that we have linearity in the parameters but not necessarily in the predictors. Thus the model $f = a\exp(-20x)$ is linear because it is linear in the parameter a, despite the predictor x. Formally speaking, a linear model is a point in the space of available functions of p arguments. The rest of the space is occupied by non-linear models. Therefore the problem of fitting model choice appears.

Sometimes physicochemical reasons help us to understand which form of model should be used. A linear model helps us to solve a system which is too complicated for substantive analysis alone. However, the easiness of this trivial choice is an illusion, since the multicollinearity problem is waiting for us at the next step. Multicollinearity [2] means the singularity of the regression information matrix. Such a problem occurs in non-linear regression too, but its interpretation is different. It can be compared to a pessimist/optimist wrangle— is this bottle half empty or half full? Linear analysis introduces an optimistic point of view. It is supposed that the model is too full, so it is necessary to restrict the number of parameters by all means (PLS, PCA). On the other hand, there are no extra parameters in a non-linear model, since the parameters are dictated by the nature of the investigated process. That is why using NLR analysis we represent the pessimistic point of view and presume insufficiency of the experimental data set. Such a view leads to specific methods of struggle against the multicollinearity in non-linear models (Bayesian for example), though nothing hinders our use of traditional ones.

The search for parameter estimates of a non-linear model is a little more complicated than for a linear one. It is not an obstacle, because the procedure of objective function optimization has been developed in detail; a number of reliable algorithms are known. Marquardt's method [3] is the most popular, but there are some more suitable methods. The two main problems we face in NLR analysis are the following: to guess the initial parameter values and to calculate the derivatives of a model. The problem of the initial guess has no simple solution (and does not seem likely to be ever solved). Here we have to place our trust in a successful choice of an experienced investigator who understands the nature of the problem. We also rely on the stability of the minimization algorithm that is able to get off from the far points. The problem of computing derivatives is easier. We will explain it in Section 4.

To find the estimates of unknown parameters of a model is only half the job. It is also necessary to interpret the obtained results, i.e. to calculate the accuracy of these estimates (standard deviations, variance–covariance matrix), to verify the lack of fit (Fisher's test, Student's test) and to build confidence intervals. The classical theory of ordinary linear regression gives simple solutions [4] for all these tasks (see Section 7). In the case of NLR we face a dilemma: whether to use a linear approximation or to apply statistical simulation methods [5,6]. The first variant is simple in calculations but does not guarantee correct results. As practice shows, the second variant gives very exact results, though it takes a long time to execute it. We have developed a new method [7] of confidence estimation for NLR. Unlike bootstrap, we simulate parameter estimates, not initial data. In order to distribute them correctly, we apply a trick using the chi-squared distribution of an objective function. It was shown that this method is as accurate as bootstrap but about 1000 times faster. It will be explained in Section 7.

However, a typical chemometrics user does not bother about the problems we are discussing here. First of all, she/he is interested in the true final result. 'Solve all your technical problems as you like, just give me a simple tool to fit my data. And for God's sake no programming, please! I don't know MATLAB! I need something easy-to-use, like Word or Excel'. That is a common position. Researchers have perfectly satisfied this demand by providing the user with a large choice of programs for linear regression analysis [1], but not for NLR. Of course, there are many soft tools for NLR, but all the programs we know do not meet the general needs.

We have developed a new tool for NLR analysis [8] that utilizes all the ideas presented in this paper. It is named FITTER. In designing it, we followed the rule 'the easier is the better'. That is why we did not create our own interface, but instead externalized all the methods as an Add-In for the popular Microsoft Excel program. FITTER is similar to the well-known Solver Add-In. As with Solver, all data necessary for solution of an NLR task are placed on a sheet of a standard workbook and then registered by dialogues. Visual Basic is a very slow language; therefore all complicated calculating procedures were written in C++ and assembled in the separate DLL, thus acquiring a speed that satisfies users. We used FITTER to process all data sets described in this paper. Therefore all experimental data and models are presented in the form they are used by FITTER, i.e. on Excel worksheets.

Generally the most important issue is the statement of the relevant NLR objective. This determines the choice of the appropriate tool for investigation. The first goal is to predict (strictly speaking, to interpolate) the response values within the area of settings. The second goal is to forecast (to extrapolate) the model values at settings that are far from the observed area. We would like to emphasize that this is another problem. It is well known that prediction error does not depend on the form of the model; it is mostly determined by measurement error. In contrast, forecast error depends firstly on the model and secondly on measurement error. Therefore 'soft' models (mainly linear) are suitable for prediction and only 'hard' (mainly non-linear) ones are applicable for forecasting [9].

In this paper we try to demonstrate the possibilities to overcome all these obstacles and to apply NLR to practical task solution. Let us start with a simple example that shows the essence of the problem.

2. *TGA example*

Figure 1 shows experimental data obtained by the method of thermogravimetric analysis (TGA) for a sample of polymer material. The main idea of TGA is the following. A sample is heated at a constant heating rate and its mass is measured in the course of heating. Figure 1 presents plots of the relative change in mass, $y = m/m_0$ (curve 1), and of the increase in temperature T from $T_0 = 373$ K at a rate of 3 °C min^{-1} (curve 2). In this case the decrease in sample mass is connected with the loss of plasticizer.

Usually, when describing the kinetics of evaporation of non-polymer components from polymers, it is supposed [10] that the rate of evaporation is proportional to the component concentration:

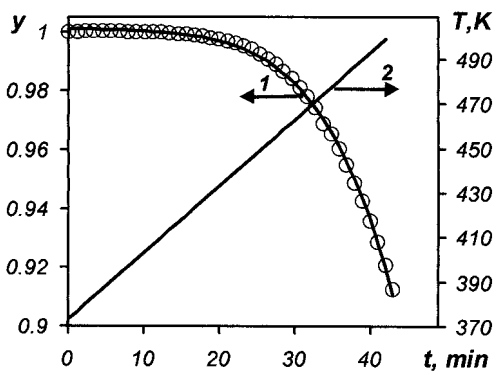


Figure 1. TGA data (○), fitting curve (1) and growth of temperature (2).

$$W_s = kC \tag{1}$$

where k is the evaporation rate constant and C is the plasticizer concentration. According to (1), the equation for the TGA response, $y = m/m_0$, has the form

$$\frac{dy}{dt} = -kC, \qquad y(0) = y_0 \tag{2}$$

With regard to shrinkage of the sample in the course of evaporation, this may be written as

$$C = 1 - \frac{1 - C_0}{y} \tag{3}$$

where C_0 is the initial concentration of plasticizer. The evaporation rate constant k is proportional to the specific surface F of the sample and depends on the temperature T according to the Arrhenius law:

$$k = k_0 \, F \, e^{-E/RT} \tag{4}$$

where R is the universal gas constant. Finally, the temperature T in the TGA experiment increases with time according to the linear equation

$$T = T_0 + vt \tag{5}$$

where v is the heating rate.

The system of Equations (2)–(5) represents the mathematical model that describes the TGA curves for the desorption of plasticizer. From the regression point of view the value y is a response, values t, C_0, F, v and T_0 are predictors and values y_0, k_0 and E are unknown parameters. This model has no analytical solution; it is typical for all dynamic methods with variable temperature. (As a rule, the solution may be expressed with the help of the function

$$E_n(z) = \int_z^\infty t^{-n} e^{-zt} dt \quad (n = 0, 1, 2, \ldots; \; z > 0)$$

This is an exponential integral [11].) However, it is easy to perform a regression analysis of the model given as a differential equation with the help of the software FITTER. In Figure 2 the model (2)–(5) and parameter estimates are shown.

This solution may be used to forecast the service life of products made of the polymer. It is known [12] that failure is connected with a loss of plasticizer. The loss causes bursting. The experiments show that the plasticizer concentration $C_c = 0{\cdot}12$ is critical, so the service life may be defined as the moment when the concentration falls to this value. The extrapolation is made to the following values of predictors: the time t is within the interval 0–25 years; the initial concentration of plasticizer, C_0, is equal to 0·35; the form factor F is defined by the article size (1·5 or 2·0); the heating rate is $v = 0$; and the temperature T_0 is equal to the service temperature (293 or 303 K). It is necessary to take into account the uncertainties in parameter estimates during forecasting. FITTER does this by building confidence intervals of forecast values.

The forecast is shown in Figure 3, where two curves of plasticizer concentrations and their 0·95 low confidence intervals are presented. Clearly, the first sample will serve about 20 years and the


```
'TGA Desorbtion Model
D[y]/D[t]=-k*[1-(1-CO)/y]; y(0)=y0
k=F*exp(k0-E/E/R/T)
T=T0+v*t
R=1.98717
y0=?
k0=?
E=?
```

Parameters estimation

Name	Initial	Final	Deviation
y0	1	1.000881	0.0002
k0	10	13.99632	0.240161
E	10000	18052.2	225.423

Figure 2. TGA model and parameters as presented on worksheet.

second one about 10 years. These results agree perfectly with known facts about the service life of similar products. Both methods of linearization and statistical simulation were used for confidence interval construction. The results were so close to each other that we cannot show them in the plot. This evidences that this model may be treated as a linear one when forecasting it to the above-mentioned conditions.

With the help of this example we come to the following obvious conclusion. Except for 'the terrible appearance', this model is no more complicated than a linear one. Moreover, it is even easier. To be convinced of this we will retrace the way of analysis again. The choice of model is simple and evident. Equations (1)–(5) are absolutely clear to any student chemists. Certainly, it was necessary to realize that the diffusion stage is not important. In fact, every polymer chemist shares this opinion. The process of estimation starts at the standard initial point $y_0 = 1$, $k_0 = 1$ and $E = 10\,000$

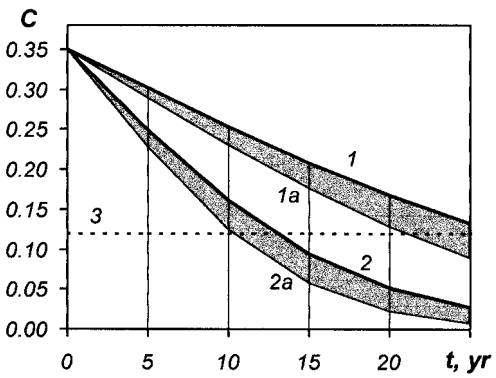


Figure 3. Forecast of service life by TGA method: 1, $T = 293$, $F = 2{\cdot}0$; 1a, 0·95 low confidence limit; 2, $T = 303$, $F = 1{\cdot}5$; 2a, 0·95 low confidence limit; 3, critical level.

(E is an activation energy so it would be unwise to set it to less then 5000 cal mol^{-1}). The calculations take only 20 s on a Pentium-100. Certainly, these data may be fitted by some linear model, e.g. $\ln(y-1) = \ln(k) + nt$, but it is impossible to use this equation for extrapolating to constant temperature. It is very difficult to create a linear model depending on all these predictors. In any case we cannot do it even if the true form of the model is known.

3. NLR BASICS

The main purpose of regression analysis is to fit some experimental data with a known curve and to predict response values. Usually a regression model contains the following components:

- data matrix $\mathbf{X} = \{x_{ij},\ i = 1,\ldots,m,\ j = 1,\ldots,N\}$ of predictor variables which are observed at N points:

$$\mathbf{x} = (x_1, \ldots, x_m)^{\mathrm{T}} \tag{6}$$

- set of measured response values **y** (observations):

$$\mathbf{y} = (y_1, \ldots, y_N)^{\mathrm{T}} \tag{7}$$

- set of weights **w** known for each observation point:

$$\mathbf{w} = (w_1, \ldots, w_N)^{\mathrm{T}} \tag{8}$$

- known function $f(\mathbf{x},\mathbf{a})$ that depends on the vector of unknown parameters:

$$\mathbf{a} = (a_1, \ldots, a_p)^{\mathrm{T}} \tag{9}$$

Here N is the number of observations (Equation (7)), m is the number of predictors **x** (Equation (6)) and p is the number of parameters **a** (Equation (9)). The response values **y** (Equation (7)) are random variables that differ from the 'true' function values f by measurement errors $\boldsymbol{\varepsilon}$. We consider two types of measurement errors: absolute error and relative error. Absolute error is added to the 'true' values:

$$y_i = f_i + \varepsilon_i, \quad i = 1, \ldots, N \tag{10}$$

and relative error is multiplied by the 'true' values:

$$y_i = f_i(1 + \varepsilon_i), \quad i = 1, \ldots, N \tag{11}$$

Regarding the random error vector

$$\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_N)^{\mathrm{T}} \tag{12}$$

the following assumptions are made:

- *unbiased*—the mean value of $\boldsymbol{\varepsilon}$ is equal to zero:

$$E(\varepsilon_i) = 0, \quad i = 1, \ldots, N \tag{13}$$

- *uncorrelated*—the covariance of different errors is equal to zero:

$$\mathrm{cov}(\varepsilon_i, \varepsilon_j) = 0, \quad i \neq j \tag{14}$$

- *homoscedastic*—the weighted variance of errors is constant:

$$w_i^2 \mathrm{cov}(\varepsilon_i, \varepsilon_i) = \sigma_\varepsilon^2 = \mathrm{const}, \quad i = 1, \ldots, N \tag{15}$$

The constant σ_ε^2 is called the weighted error variance. Usually it is unknown and should be estimated together with parameters **a** (Equation (9)). Note that the variable σ_ε^2 (Equation (15)) is not a response variance; the latter may be presented for non-zero weights only, as $D(y_i) = w_i^{-2}\sigma_\varepsilon^2$. By default, all weights are equal to one. If data do not agree with the homoscedastic assumption (Equation (15)), it is possible to set different weight values in order to achieve constant weighted variance at each observation point.

There may be prior information about unknown parameters **a** (Equation (9)) and weighted error variance σ_ε^2 (Equation (15)). It is possible to make a fitting with respect to it. Such information should be presented in Bayesian form including the following components:

- prior parameter (Equation (9)) values:

$$\mathbf{b} = (b_1, \ldots, b_p)^{\mathrm{T}} \tag{16}$$

- prior Bayesian information matrix:

$$\mathbf{H} = \{h_{ij},\ i = 1, \ldots, p,\ j = 1, \ldots, p\} \tag{17}$$

- prior value of weighted error variance (Equation (15)):

$$s_0^2 \tag{18}$$

- prior number of degrees of freedom:

$$n_0 \tag{19}$$

Prior information involving all four items may be called Bayesian information of full range. Sometimes the prior value of weighted error variance (Equation (18)) and its number of degrees of freedom (Equation (19)) are unknown. In this case we have so-called Bayesian information of short range. The Bayesian information matrix **H** (Equation (17)) is the matrix inverse of the covariance matrix **C** (Equation (34)) and is similar to the **F** matrix (Equation (39)). If the prior information involves more parameters than needed, it should be recalculated. The following algorithm performs this operation. Suppose one is interested in the first r elements of prior parameter vector **b** (Equation (16)) (they are used as prior ones) and that the last $p-r$ elements are not present in the problem and should be discarded. Then the primary Bayesian information matrix **H** (Equation (17)) is transformed to a new one by the equation

$$\tilde{\mathbf{H}} = \begin{bmatrix} \mathbf{X} - \mathbf{Z}\mathbf{Y}^{-1}\mathbf{Z}^{\mathrm{T}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{20}$$

In this formula, **X, Y** and **Z** are the block matrices composing the initial matrix **H**:

$$\mathbf{H} = \begin{bmatrix} \mathbf{X} & \mathbf{Z} \\ \mathbf{Z}^{\mathrm{T}} & \mathbf{Y} \end{bmatrix}$$

Here the $r \times r$ square matrix **X** corresponds to the first r elements of vector **b**, the $(p-r) \times (p-r)$ square matrix **Y** corresponds to the last $p-r$ elements of **b**, and **Z** is a rectangular matrix with dimension $r \times (p-r)$. If the prior information comprises fewer parameters than needed, it is completed with zero values up to the proper dimension.

The goal of regression analysis is to determine the values of parameters **a** (Equation (9)) so that a function $f(\mathbf{x},\mathbf{a})$ fits a set of observed response values of **y** in the best way. The maximum likelihood method (MLM) [13] determines that these values minimize the objective function $Q(\mathbf{a})$, i.e.

$$\hat{\mathbf{a}} = \arg\min Q(\mathbf{a}) \tag{21}$$

The objective function consists of two elements: a sum of squares $S(\mathbf{a})$ and a Bayesian term $B(\mathbf{a})$. The sum of squares $S(\mathbf{a})$ is calculated as

$$S(\mathbf{a}) = \sum_{i=1}^{N} w_i^2 (y_i - f_i)^2 g_i^2 \tag{22}$$

where $f_i = f(\mathbf{x}_i,\mathbf{a})$. For absolute measurement error (Equation (10)), $g_i = 1$. For relative measurement error (Equation (11)), $g_i = 1/y_i$. The Bayesian term $B(\mathbf{a})$ is calculated depending on the type of Bayesian information (Equations (16)–(19)). For Bayesian information of full range,

$$B(\mathbf{a}) = s_0^2 (n_0 + R(\mathbf{a})) \tag{23}$$

For Bayesian information of short range,

$$B(\mathbf{a}) = \exp\left(\frac{R(\mathbf{a})}{N_w}\right) \tag{24}$$

Here $R(\mathbf{a})$ is the quadratic form

$$R(\mathbf{a}) = (\mathbf{a} - \mathbf{b})^{\mathrm{T}} \mathbf{H} (\mathbf{a} - \mathbf{b}) \tag{25}$$

The vector **b** (Equation (16)), the matrix **H** (Equation (17)) and the values s_0^2 (Equation (18)) and n_0 (Equation (19)) are the elements of prior information in Bayesian form, while N_w is the number of observations (Equation (7)) with non-zero weights (Equation (8)). The objective function $Q(\mathbf{a})$ is comprised of $S(\mathbf{a})$ and $B(\mathbf{a})$ as follows. If Bayesian information is absent,

$$Q(\mathbf{a}) = S(\mathbf{a}) \tag{26}$$

For Bayesian information of full range,

$$Q(\mathbf{a}) = S(\mathbf{a}) + B(\mathbf{a}) \tag{27}$$

For Bayesian information of short range,

$$Q(\mathbf{a}) = S(\mathbf{a}) B(\mathbf{a}) \tag{28}$$

The number of degrees of freedom by fit, N_f, is the important value used in NLR analysis. If Bayesian information is absent or of short range, N_f is calculated as

$$N_f = N_w - p \tag{29}$$

For Bayesian information of full range it is calculated as

$$N_f = N_w - n_0 \tag{30}$$

Here p is the number of unknown parameters (Equation (9)) and N_w is the number of responses (Equation (7)) with non-zero weights. The estimate of weighted error variance (Equation (15)) may be found by the fitting curve. It is calculated as

$$s^2 = \mathrm{SS}/N_f \tag{31}$$

where N_f is the number of degrees of freedom by fit (Equations (29) and (30)) and SS is calculated as follows. If Bayesian information is absent or of short range, then

$$\mathrm{SS} = \min S(\mathbf{a}) = \mathrm{S}(\hat{\mathbf{a}}) \tag{32}$$

For Bayesian information of full range,

$$\mathrm{SS} = \min Q(\mathbf{a}) = S(\hat{\mathbf{a}}) + s_0^2(R(\hat{\mathbf{a}}) + n_0) \tag{33}$$

Here $S(\mathbf{a})$ is the sum of squares (Equation (21)) and the values s_0^2 (Equation (18)) and n_0 (Equation (19)) are the elements of prior information in Bayesian form.

Parameter estimates calculated by MLM are random variables which vary with the experimental data. Their covariance matrix $\mathbf{C} = \mathrm{cov}(\hat{\mathbf{a}},\hat{\mathbf{a}})$ is calculated by the formula

$$\mathbf{C} = s^2\mathbf{A}^{-1} \tag{34}$$

where $\mathbf{A}$ is the Hessian matrix (Equations (35)–(37)) and s^2 is the weighted variance estimated (Equation (31)) by the fitting curve. The Hessian matrix is used in the search procedure and in calculation of the covariance matrix $\mathbf{C}$. This matrix is calculated depending on the type of Bayesian information. If Bayesian information is absent,

$$\mathbf{A} = \mathbf{V}^{\mathrm{T}}\mathbf{V} \tag{35}$$

For Bayesian information of full range,

$$\mathbf{A} = \mathbf{V}^{\mathrm{T}}\mathbf{V} + s_0^2\mathbf{H} \tag{36}$$

For Bayesian information of short range,

$$\mathbf{A} = \exp\left(\frac{R(\hat{\mathbf{a}})}{N_w}\right)\left(\mathbf{V}^{\mathrm{T}}\mathbf{V} + \frac{S(\hat{\mathbf{a}})}{N_w}\mathbf{H}\right) \tag{37}$$

Here $\mathbf{V}$ is p × N matrix whose elements are weighted derivatives of the fitting function,

$$V_{ij} = w_j\frac{\partial f(\mathbf{x}_j, \hat{\mathbf{a}})}{\partial a_i}, \quad i = 1, \ldots, p, \quad j = 1, \ldots, N \tag{38}$$

the values $R(\mathbf{a})$ (Equation (25)) and $S(\mathbf{a})$ (Equation (22)) are components of the objective function, the value s_0^2 (Equation (18)) and the matrix $\mathbf{H}$ (Equation (17)) are the elements of prior information in Bayesian form, the value N is the number of observations (Equation (7)) and the vector $\hat{\mathbf{a}}$ (Equation (21)) is the MLM estimator of unknown parameters $\mathbf{a}$ (Equation (9)). The $\mathbf{F}$ matrix is the matrix inverse of the covariance matrix $\mathbf{C}$ (Equation (34)), i.e.

$$\mathbf{F} = s^{-2}\mathbf{A} \tag{39}$$

and is similar to the Bayesian information matrix $\mathbf{H}$ (Equation (17)).

4. CALCULATION OF DERIVATIVES AND PRECISION

As we have already mentioned, the analysis of non-linear models requires high accuracy of calculations. The accuracy in itself is not important (practical computations do not require more than four or five significant digits), but it is essential as a weapon in the struggle against multicollinearity. For example, for a matrix inversion (a routine operation in regression analysis) with a spread in eigenvalues (or conditional number) equal to eight decimal orders (from 10^{-4} to 10^4), it is necessary to carry out intermediate calculations with a precision of about 10^{-10}. When it is necessary to solve an implicit or differential equation for a model evaluation, intermediate computations have to be done even more precisely to reduce error accumulation.

First of all, accuracy of estimation depends on the method applied for calculation of derivatives (Equation (38)) of a model with respect to unknown parameters. There are three practical solutions. Powell and Fletcher [14] methods do not require calculation of derivatives at all. Another possibility is to use numerical calculation of derivatives, applying difference formulae such as

$$\frac{\partial f}{\partial a} = \frac{f(a+h) - f(a)}{h}$$

The last solution is to apply direct analytical calculations. The first solution is rather weak and unreliable [13] and so cannot be applied to complicated models. The second one is the most popular. This approach reduces the accuracy of computations greatly but, as a rule, does not give more than six or seven true digits in the result. Thus it becomes impossible to use differential equations or

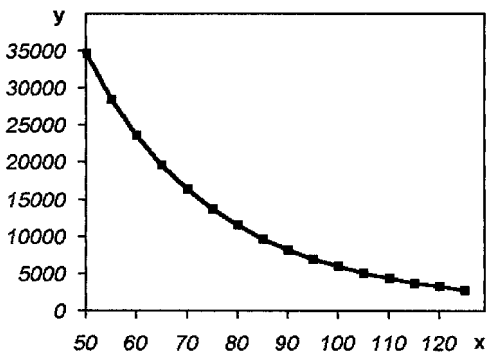
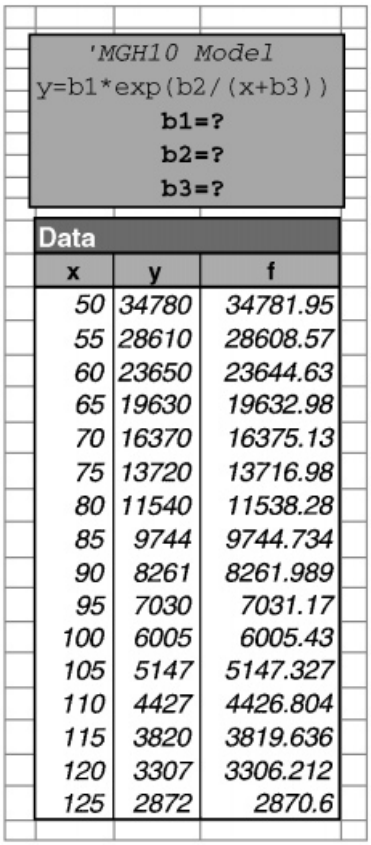


Figure 4. MGH10 data (■) and fitting curve.



'MGH10 Model
y=b1*exp(b2/(x+b3))
b1=?
b2=?
b3=?

Data

x	y	f
50	34780	34781.95
55	28610	28608.57
60	23650	23644.63
65	19630	19632.98
70	16370	16375.13
75	13720	13716.98
80	11540	11538.28
85	9744	9744.734
90	8261	8261.989
95	7030	7031.17
100	6005	6005.43
105	5147	5147.327
110	4427	4426.804
115	3820	3819.636
120	3307	3306.212
125	2872	2870.6

Figure 5. MGH10 data and model as presented on worksheet.

implicit functions in NLR analysis owing to the total loss of precision. Several tools advise users to find derivatives themselves and input the analytical formulae into the program. Oh no, thank you! Here is a computer and that is its drudgery. That is why we have instructed the computer to find analytical derivatives of any equation. A special algorithm is applied to do this. Thus, when a user inputs a formula, e.g. $y = \exp(-at)$, the program detects its derivatives, e.g. finds $dy/da = -t\exp(-at)$. In combination with the compiler that creates objective code for computation of model values by its analytical formula, it allows us to carry out calculations with high accuracy even for systems of differential equations.

We have used Statistical Reference Datasets [15] developed by the National Institute of Standards and Technology (USA) to demonstrate the robustness and reliability of FITTER. This is a collection of data sets with associated certified values of model-dependent parameters. The collection includes both generated and 'real-world' non-linear least squares problems of varying levels of difficulty. There are 27 problems. The values certified by NIST are the 'best available' solutions, obtained using 128 bit precision and confirmed by at least two different algorithms and software packages using analytical derivatives. The certified results (parameters, parameter standard deviations, residual sum of squares, residual standard deviation) are reported to 11 decimal places for each data

Table I. NIST and FITTER values for MGH10 model

	NIST-certified values		FITTER-estimated values	
	Parameter	Standard deviation	Parameter	Standard deviation
b_1	$5{\cdot}6096364710 \times 10^{-3}$	$1{\cdot}5687892471 \times 10^{-4}$	$5{\cdot}6096364711 \times 10^{-3}$	$1{\cdot}5687892471 \times 10^{-4}$
b_2	$6{\cdot}1813463463 \times 10^{3}$	$2{\cdot}3309021107 \times 10^{1}$	$6{\cdot}1813463463 \times 10^{3}$	$2{\cdot}3309021107 \times 10^{1}$
b_3	$3{\cdot}4522363462 \times 10^{2}$	$7{\cdot}8486103508 \times 10^{1}$	$3{\cdot}4522363462 \times 10^{2}$	$7{\cdot}8486103507 \times 10^{-1}$
s^2	$8{\cdot}7945855171 \times 10^{1}$		$8{\cdot}7945855171 \times 10^{1}$	

set. FITTER gives 10 or 11 correct decimal places for most of the examples. We consider these results extremely good, as calculations were done in double precision on a 32 bit machine.

One of the NIST examples is presented in Figure 4. These data were simulated and are described in detail in Reference [16]. The response variable is y and the predictor variable is t. The numerical values of data and the model are shown in Figure 5. In Table I the results of fitting are presented. One can see that they are every close to the NIST-certified values.

5. PARAMETER ESTIMATION

In accordance with the maximum likelihood method, the estimation of regression parameters is made by minimization of the objective function (Equations (26)–(28)). In gradient search methods the kernel of the minimization algorithm is the inversion of the Hessian matrix $\mathbf{A}$ (Equation (35)–(37)). This procedure is repeated at every iteration step [13]. As a rule, the search starts at a point where this matrix does not have full rank. That is why it cannot be fully inverted and some special regularizing tricks are applied. The Marquardt method [3] is the most popular. The matrix $\mathbf{A}$ is replaced with the matrix $\mathbf{A}^* = \mathbf{A} + \alpha\mathbf{I}$, where α is a small number and $\mathbf{I}$ is a unit matrix. Matrix $\mathbf{A}^*$ always has full rank and may be inverted. By the way, our experience shows that even if $\mathbf{A}$ is a full rank matrix, the pseudoinversion method should be used for quick and reliable convergence of the search.

For the same purposes we use the recurrence algorithm [17] based on the exponential matrix. It provides high stability; besides, it allows calculations of the spread in eigenvalues and completeness of search at every iteration. The main idea of this approach is the following. For inversion of the Hessian matrix $\mathbf{A}$ at every step of minimization, the recurrence algorithm is used:

$$\mathbf{B}(2t) = \mathbf{B}(t)(2\mathbf{I} - \mathbf{AB}(t)) \tag{40}$$

It is easy to see that matrix $\mathbf{B}(t)$ satisfies the matrix differential equation

$$\frac{\mathrm{d}\mathbf{B}(t)}{\mathrm{d}t} = \mathbf{I} - \mathbf{AB}(t), \qquad \mathbf{B}(0) = \mathbf{0}$$

with solution

$$\mathbf{B}(t) = \int_0^t \exp(-\mathbf{A}s)\mathrm{d}s \tag{41}$$

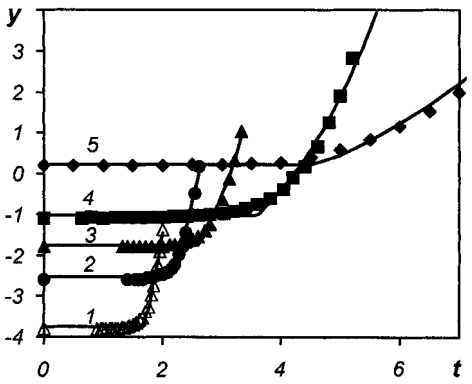


Figure 6. DSC data and fitting curves for various heating rates V (°C min^{-1}): 1, 20; 2, 15; 3, 10; 4, 5, 2.

According to (41), matrix $\mathbf{B}(t) \rightarrow \mathbf{A}^{-1}$ as $t \rightarrow \infty$. If $\mathbf{A}$ is a singular matrix, then matrix $\mathbf{B}(t)$ does not lose sense and gives the pseudoinverse matrix $\mathbf{A}^{+}$. In fact, usage of this algorithm means that one has to set the number of 'doublings', L, in Equation (40) and the small initial value β for the initial matrix $\mathbf{B}_1 = \beta\mathbf{I}$, which is connected with the precision of calculation. Then, each next matrix $\mathbf{B}_{n+1}$ is calculated as $\mathbf{B}_{n+1} = \mathbf{B}_n(2\mathbf{I} - \mathbf{AB}_n)$. The number L is similar to the regularator α in the Marquardt method. The higher L, the closer $\mathbf{B}_L$ is to the inverse matrix $\mathbf{A}^{-1}$. In the course of this recurrent procedure it is easy to monitor the trace of matrix $\mathbf{I} - \mathbf{AB}_n$:

$$sp = \text{trace}(\mathbf{I} - \mathbf{AB}_n)$$

This value changes from p at the beginning of recurrence to zero at the end (p is the number of parameters, i.e. the dimension of the matrices). Such behaviour of sp is connected with the nature of the eigenvalues of matrix $\mathbf{A}$. The smart algorithm acts like a PCA method when it inverts the matrix. It begins with the largest eigenvalue, then takes into account the next largest, then the next one, and thus it proceeds until all eigenvalues are considered or the procedure is cancelled. When the search is far from the optimum, it is better to restrict the number of eigenvalues to be accounted for in the inversion procedure. Limiting the number of 'doublings' does this. If trace sp is stable during n 'doublings', it means that the ratio of two neighbouring eigenvalues is more the 2^n, so it is reasonable to limit n by some value, e.g. 20. By applying such a method, we essentially accelerate the search procedure. Besides, the stability of the recurrent procedure allows us to invert matrices with a large spread in eigenvalues.

Figure 6 shows experimental data obtained by differential scanning calorimetry (DSC) on polypropylene samples. The DSC response y is the heat flow in a sample as a function of time t or temperature T that increases at a constant heating rate V, i.e. $T = T_0 + Vt$. If there is a chemical reaction with non-zero thermal effect, then the DSC signal is proportional to the rate of the reaction. The main problem of DSC is a quantitative interpretation of the data. Non-linear regression helps to solve this problem. Polymer thermo-oxidation is a free radical chain reaction. According to this theory [18], the data were fitted by a complex model including three predictors, i.e. V (heating rate), m (sample mass) and t (time), and 12 parameters, i.e. k (pre-exponential factor), E (activation energy), five values of initial oxidation time, t_{20}, t_{15},…,t_2, and five background parameters, f_{20},

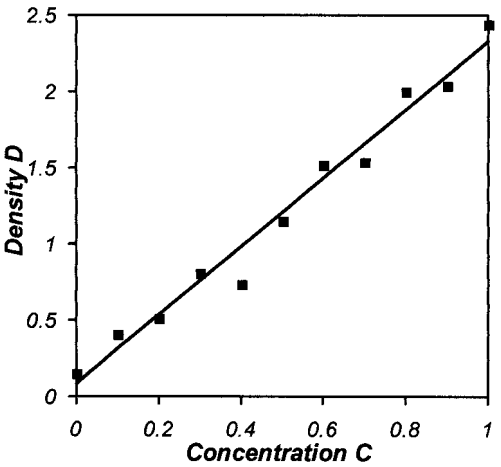


Figure 7. Calibration data and fitting line.

$f_{15},\ldots,f_2$, for every heating rate V:

$$y = f_v + \begin{cases} 0, & t < t_v \\ mC\mathrm{e}^{-E/RT}, & t > t_v \end{cases}$$

$$\frac{\mathrm{d}C}{\mathrm{d}t} = k\mathrm{e}^{-E/RT}, \qquad C(t_v) = 0 \tag{42}$$

$$T = T_0 + Vt$$

This example shows that even extremely complex models can be processed with the help of this special optimization algorithm.

6. BAYESIAN ESTIMATION

Often we have to fit several data sets with individual models. Trouble arises if the models have common unknown parameters. The classical approach offers to estimate one multiresponse regression for simultaneous fitting of all data sets. It is difficult to carry out such regression owing to the large number of estimated parameters and the necessity to invert a large matrix. A new method allows processing of the data successively for every response. Each data set is fitted individually, but we take into account the information about common parameters estimated on previous data. A posterior Bayesian distribution is built after every set processing. This information is then used as prior information for processing the next data set. It is shown [19] that for linear regression such a technique gives the same estimates as the traditional multiresponse method.

The following example illustrates this method in application to Bayesian calibration. There are two data sets: calibration data and kinetic data. Calibration data contain the values of optical density D (response) measured at different concentrations C (predictor) of some product (see Figure 7). These data are used for evaluation of calibration parameters D_0 and D_1 by a simple linear model

$$D = D_0 + D_1 C \tag{43}$$

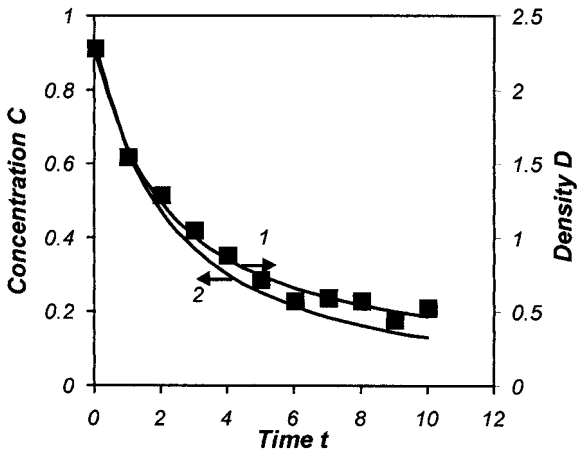


Figure 8. Kinetic data (■), fitting curve (1) and predicted concentration kinetics (2).

The second data set (see Figure 8) contains the results of a kinetic experiment where the optical density D (response) has been measured in the course of a chemical reaction of the nth order:

$$D = D_2 + D_1 C$$

$$\frac{dC}{dt} = -kC^n, \qquad C(0) = C_0 \tag{44}$$

The problem is to find the kinetic parameters C_0, k and n in the presence of waste parameters D_1 (which is the same as in the calibration data) and D_2 (a new one). The solution of the problem consists of three steps: calibration data fitting, Bayesian information building and kinetic data fitting.

At the first step the calibration parameters are estimated by the model (Equation (43)) as

$$D_0 = 0{\cdot}0882, \quad D_1 = 2{\cdot}24364$$

and the **F** matrix (Equation (38)) is obtained as

$$\mathbf{F} = \begin{bmatrix} 751{\cdot}677 & 375{\cdot}839 \\ 375{\cdot}839 & 263{\cdot}087 \end{bmatrix}$$

The error variance estimate (Equation (31)) is $s^2 = 0{\cdot}0146$ and the number of degrees of freedom by fit (Equation (29)) is $N_f = 9$. The aim of the next step is to build Bayesian information about the common parameter D_1 that should be transferred to the next kinetic data fitting step. The prior number of degrees of freedom and the prior variance value should be set equal to those values found at the first step. Thus we assume that the experiment error in the kinetic data is the same as in the calibration data and may be used for improving the error variance estimate. Since D_0 is not a common parameter, we will obtain the following results corresponding to Equations (16)–(19):

- prior parameter values (Equation (16)): $D_0 = 0$, $D_1 = 2{\cdot}24364$;

Table II. Parameters of model (44)

Parameter	Estimate	Deviation with Bayes	Deviation without Bayes
D_2	0·17929	0·59556	0·38660
D_1	2·24364	0·08380	0
C_0	0·93090	0·27718	0·17851
k	0·45852	0·11395	0·07353
n	1·72259	1·24142	0·80585
s^2	0·00773	—	—

- prior Bayesian information matrix (Equation (17)), (recalculated using Equation (20)):

$$\mathbf{H} = \begin{bmatrix} 0 & 0 \\ 0 & 75{\cdot}1677 \end{bmatrix}$$

- prior value of weighted error variance (Equation (18)): $s_0^2 = 0{\cdot}0146$;
- prior number of degrees of freedom (Equation (19)): $n_0 = 9$.

At the third step the kinetic data are fitted with the Bayesian information that was built at the second step. The results of fitting model (44) are presented in Figure 8 and Table II. Certainly, kinetic data may be fitted without Bayesian information but with fixed parameter $D_1 = 2{\cdot}24364$. The estimates of parameters are the same, but their standard deviations are different. The last column of the table contains these deviations. Hence, if one is bothered about the uncertainties in the parameter estimates, one should use the Bayesian estimation technique.

7. CONFIDENCE INTERVAL

Using estimated parameter values $\hat{\mathbf{a}}$ (Equation (21)), we can predict (forecast) the fitting model value at any desired predictor point $\mathbf{x}$:

$$\hat{f}(\mathbf{x}) = f(\mathbf{x},\hat{\mathbf{a}}) \tag{45}$$

This is the mean (to be more accurate it is the median) value of the forecast. To present the accuracy of this prediction, it is necessary to build confidence intervals. The definition of this interval is

$$\text{Prob}\{l(\mathbf{x},P) < \hat{f}(\mathbf{x}) < r(\mathbf{x},P)\} = P \tag{46}$$

Here the value P is the confidence probability and $l(\cdot)$ and $r(\cdot)$ are left and right interval limits. Applying the traditional linearization method [4], these limits may be calculated as

$$f(\mathbf{x},\hat{\mathbf{a}}) + g(P)\sqrt{\mathbf{v}^{\mathrm{T}}\mathbf{C}\mathbf{v}}$$

where $\mathbf{C}$ is the covariance matrix (Equation (34)) and

$$v_j = \frac{\partial f(\mathbf{x},\hat{\mathbf{a}})}{\partial a_j}, \quad j = 1,\ldots,p$$

For a one-sided confidence interval the function g is $g(P) = G^{-1}(P)$ and for a two-sided confidence

interval it is $g(P) = \pm G^{-1}(0{\cdot}5 + 0{\cdot}5P)$. Here the function $G^{-1}(\cdot)$ is the reciprocal of the normal distribution value (quantile).

Sometimes this method gives wrong results for non-linear models (see example below). That is why we have developed a new method [7] of confidence interval construction (Equation (46)). This method is a modification of the well-known bootstrap method [5,6]. The basic idea of traditional bootstrap is to simulate new response data

$$\mathbf{y}^* = f(\mathbf{x},\hat{\mathbf{a}}) + \varepsilon^*, \qquad \varepsilon^* \sim N(\mathbf{0},s^2\mathbf{I})$$

with the help of a pseudorandom distribution based on the estimated parameter $\hat{\mathbf{a}}$ (Equation (21)) and variance s^2 (Equation (31)) values. Such responses are used to find new parameter estimates $\mathbf{a}^*$ by MLM (Equation (21)) and to predict the model value

$$f^* = f(\mathbf{x}, \mathbf{a}^*) \tag{47}$$

By repeating such simulations M times, one can obtain a population of values (Equation (47)). Then the confidence limit is calculated as the $100P$%-percentile of this population:

$$r(\mathbf{x},P) = P\text{-percentile}\{f_1^*, \ldots, f_M^*\} \tag{48}$$

Experience shows that bootstrap gives very exact values even for non-linear models. The only drawback is that it requires too much time to perform. It is obvious that most troubles are connected with minimization of the objective function (Equations (26)–(28)). It takes 10 s to fit a complicated model with a large number of non-linear parameters. For a reliable forecast it is necessary to execute not less than $M = 1000$ recurrences, leading to more than 3 h of work.

In carrying out our research, we repeatedly observed that, in spite of the non-linearity of the model, the distribution of random variables

$$c(\mathbf{a}) = s^{-2}(Q(\mathbf{a}) - Q(\hat{\mathbf{a}})) \tag{49}$$

is very close to the χ^2 law with p degrees of freedom. In Equation (49), Q is an objective function (Equations (26)–(28)) and s^2 is a weighted variance estimate (Equation (31)). For a linear normal regression the statement is exactly true, because $c(\mathbf{a}) = (\mathbf{a}-\hat{\mathbf{a}})^{\mathrm{T}}\mathbf{F}^{-1}(\mathbf{a}-\hat{\mathbf{a}})$, where $\mathbf{F}$ is the $\mathbf{F}$ matrix (Equation (39)). In the non-linear case the validity of it may be proved by the central limit theorem [20]. Also, Equation (49) determines 'the area of indifference' $c(\mathbf{a}) < C$ of parameter estimates [13] for the likelihood function

$$L(\mathbf{y}, \boldsymbol{\alpha}) = \sigma^{-n}\exp(-Q(\mathbf{y}, \boldsymbol{\alpha})/2\sigma^2)$$

(see Figure 11). We suppose that for $C(P) = s^2\chi_p^2(P)$, where $\chi_p^2(P)$ is the P-quantile of the chi-squared distribution with p degrees of freedom, the statement

$$\mathrm{Prob}\{\mathbf{a}: c(\mathbf{a}) < C(P)\} \geq P \tag{50}$$

is fulfilled with sufficient accuracy for non-linear models even for a small number of degrees of freedom.

Below we would like to explain our algorithm that is based on these considerations and improves the traditional bootstrap method. It consists of the following steps repeated several times. At the first

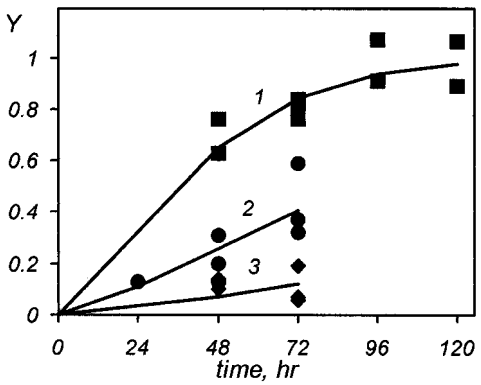


Figure 9. Data and fitting curves: 1 (■), T = 383 K; 2 (●), T = 368 K; 3 (◆), T = 353 K.

step we generate pseudorandom variable $\mathbf{a}^*$ distributed in accordance with the p-dimensional normal law:

$$\mathbf{a}^* \sim N(\hat{\mathbf{a}}, \mathbf{C}) \tag{51}$$

where $\hat{\mathbf{a}}$ is the vector of parameter estimates (Equation (21)) obtained from the real data (Equation (8)) and $\mathbf{C}$ is the parameter covariance matrix (Equation (34)). At the next step the value $c(\hat{\mathbf{a}}^*)$ (Equation (49)) is calculated and checked with the help of the chi-squared distribution. The details of this test will be introduced in the next section. If such a test is accepted, the predicted model value f^* (Equation (47)) is calculated, otherwise we return to the first step. The same sequence of steps is repeated until the number of simulated values is equal to the predefined number of simulations, M. Thereupon the confidence limits are calculated as the P-percentile of this population by Equation (48). It was shown [7] that this method (named associated simulation) gives rather satisfactory intervals and takes 1000 times less to perform than the bootstrap method.

This example shows the difference between prediction methods (see Figures 9 and 10). The data (Figure 9) are the results of generated accelerated aging tests [9] performed at temperatures T = 383, 368 and 353 K. The response value (Y) is calculated by the kinetic model

$$Y = 1 - \exp[-(kt)^n] \tag{52}$$

where the reaction rate constant $k = \exp(k_0 - E/RT)$ depends on temperature by the Arrhenius law. For simulation we use the following values of parameters ('true' values):

$$n = 1{\cdot}5, \quad k_0 = 17, \quad E = 16000, \quad \sigma^2 = 0{\cdot}005 \tag{53}$$

Their estimates (Equations (21) and (31)) obtained by FITTER are

$$\hat{n} = 1{\cdot}3586, \quad \hat{k}_0 = 17{\cdot}2382, \quad \hat{E} = 16019{\cdot}5, \quad s^2 = 0{\cdot}0053 \tag{54}$$

The prediction is performed for the value Y extrapolated to the normal temperature T = 293K (20 °C) and time $t <$ 8640 h (1 year). Both linearization and associated simulation methods of forecasting are

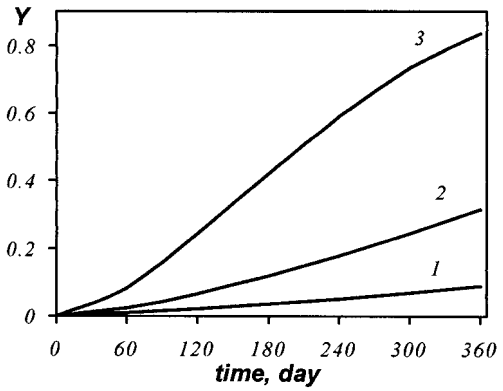


Figure 10. Forecasted values (1) and 0·95 upper confidence limits by linearization method (2) and associated simulation method (3).

tested. The results are presented in Figure 10. This example shows the great difference in forecasted values depending on the applied method of prediction.

8. COEFFICIENT OF NON-LINEARITY

What is the difference between linear and non-linear regression? Why is the complicated model (2)–(5) more similar to a linear model than the simple model (52)? How may non-linearity be evaluated? To answer such questions, we suggest a new coefficient of non-linearity. It is calculated by the Monte Carlo procedure and accounts for the model structure as well as the experimental design features.

To explain this coefficient, we have to return to the procedure of associated simulation (AS). At the first step the pseudorandom variable $\mathbf{a}^*$ (Equation (51)) was simulated. Its distribution was selected in order to obtain the right properties of the population for the linear model—mean value equal to $\hat{\mathbf{a}}$ (Equation (21)) and covariance matrix equal to $\mathbf{C}$ (Equation (34)). In other words, one may expect to find the P-share of this population in the ellipse

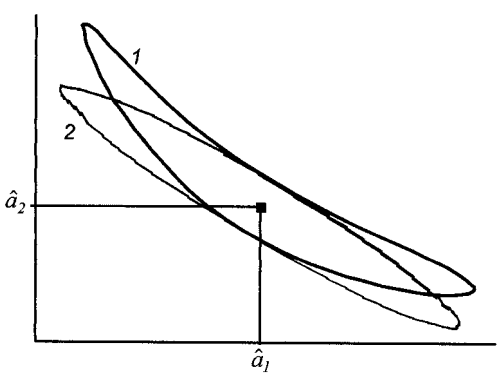


Figure 11. Objective function contour (1) and its approximating ellipse (2).

$$(\mathbf{a}-\hat{\mathbf{a}})^{\mathrm{T}}\mathbf{A}(\mathbf{a}-\hat{\mathbf{a}}) < s^2\chi_p^2(P). \tag{55}$$

where $\chi_p^2(P)$ is the P-quantile of the chi-squared distribution with p degrees of freedom and $\mathbf{A}$ is the Hessian matrix (Equations (35)–(37)); (see curve 2 in Figure 11). This ellipse approximates the contour of the objective function

$$Q(\mathbf{a}) - Q(\hat{\mathbf{a}}) < s^2\chi_p^2(P) \tag{56}$$

(see curve 1 in Figure 11). If the regression model is linear, the curves coincide. In the AS method we have to obtain the parameter population distributed according to objective function contours. This means that for all values of probability P ($0 < P < 1$) the share of simulated values $\mathbf{a}^*$ that lie in the area (56) should be equal to P. To acquire it, we choose some values of probabilities $0 < P_1 < P_2 < \ldots < P_k < 1$ which divide the parameter space into $k+1$ separate areas and calculate the expected numbers of area hits, $m_0 + m_1 + \ldots + m_k = M$. At each step of the simulation we determine what area (56) a current realization of $\mathbf{a}^*$ is in and check whether the current number of hits of this area is more than the expected number. In this case the realization is rejected. It is clear that the more a model differs from the linear one, the more realizations are rejected. Thus we obtain some criterion that gives a chance to estimate the non-linearity of the model. The criterion can measure the distance between the approximating ellipse (55) and the contour curve (56). There have been a lot of studies on the coefficient of non-linearity [21–24], beginning with the famous paper by Beale [21]. All of them deal with geometric properties of the regression curve and calculate the coefficient of non-linearity as a curative measure by applying second derivatives of the model. Thus it is a determined value. We prefer to use a random value. This may be obtained with the standard chi-squared test.

Let $m_0 + m_1 + \ldots + m_k = M$ be the expected values and $m'_0 + m'_1 + \ldots + m'_k = M'$ be the actual values of hits of predefined areas in the Monte Carlo procedure (Equation (51)). It is obvious that $m_i \leq m'_i$. The following value is a standard test value for grouped observations:

$$\chi^2 = \sum_{i=0}^{k} \frac{(m'_i M - m_i M')^2}{m_i M M'} \tag{57}$$

If the regression model is linear, the value (57) has a chi-squared distribution with k degrees of freedom. To make the coefficient of non-linearity more suitable in practice, it may be defined as

$$\Gamma = \frac{\chi^2}{\chi_k^2(0{\cdot}95)} \tag{58}$$

In this ratio the numerator is the test value (57) and the denominator is the 0·95-quantile of the chi-squared distribution with k degrees of freedom. For a linear (or closely linear) model the coefficient (58) may be less than one. The larger it is, the more non-linear is the model. To obtain an accurate value of coefficient (58), one has to make a lot of simulations ($M > 1000$). However, there is no sense in the precise value of the coefficient of non-linearity. It must show only tendencies of the model, so usually we quantize it in the following way. If $\Gamma < 1{\cdot}2$, then $\gamma = 1$; if $1{\cdot}2 \leq \Gamma < 1{\cdot}5$, then $\gamma = \Gamma$; if $1{\cdot}5 \leq \Gamma < 2{\cdot}5$, then $\gamma = 2$; if $2{\cdot}5 \leq \Gamma < 3{\cdot}5$, then $\gamma = 3$; if $3{\cdot}5 \leq \Gamma < 4{\cdot}5$, then $\gamma = 4$; if $\Gamma \geq 4{\cdot}5$, then $\gamma > 5$. This rough coefficient γ may be calculated with the help of a rather small number of simulations ($M \approx 1000$). The coefficient of non-linearity of model (52) for the data set presented in Figure 9 is $\gamma = 3$.

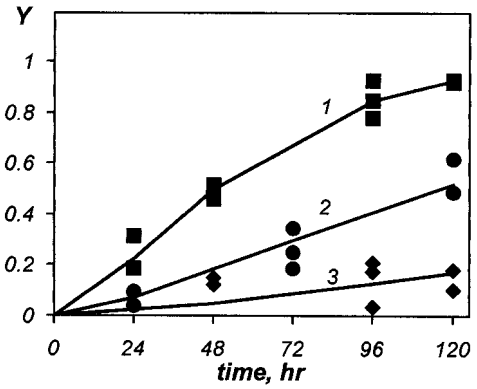


Figure 12. Data and fitting curves: 1 (■), T = 383 K; 2 (●), T = 368 K; 3 (◆), T = 353 K.

Certainly the coefficient of non-linearity depends on the form of the model, but much depends on the experimental design. The following example illustrates this point. In Figure 12 one can see data simulated by model (52) with parameters (53). They are similar to the data in Figure 9 but differ in experimental design. They were fitted, and Figure 13 shows the results of forecasting to the same conditions as in Figure 10. Comparing curves 3 in Figures 10 and 13, one can see that the non-linearity drops off. The coefficient of non-linearity calculated for this design is $\gamma = 1$. Table III gives the values of the coefficient γ for all models reported in this paper.

9. TMA EXAMPLE

This example illustrates a practical application of NLR for technological process optimization. The object under study is polyethylene (PE) fibre applied in shrinkable products. It is known that a memory effect, caused by the shrinkability of polyethylene, is achieved by cross-linking. This network is produced by application of small radiation doses. It is necessary to select the optimum doses of irradiation for different grades of PE. The composition and structure of different PE grades

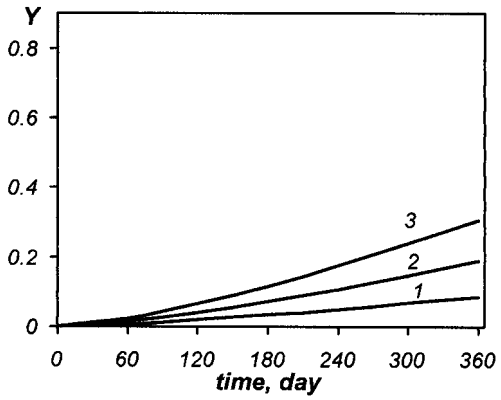


Figure 13. Forecasted values (1) and 0·95 upper confidence limits by linearization method (2) and associated simulation method (3).

Table III. Coefficients of non-linearity

Model and data	Exact coefficient of non-linearity Γ	Rough coefficient of non-linearity γ
TGA (Equations (2)–(5)	1·14	1·14
DSC (Equation (42), Figure 6)	8·43	>5
MGH10 (Figure 5)	151·0	>5
Bayes (Equation (44))	34·7	>5
Aging (Equation (52), Figure 9)	2·8	3
Aging (Equation (52), Figure 12)	0·96	1

may differ greatly. The main difficulty is that ordinary measurement methods are insufficiently sensitive for the low degree of cross-linking that is used in shrinkable PE. Therefore we use thermomechanical analysis (TMA) for density control of the network in the course of irradiation. TMA measurements are carried out on standard equipment. A specimen of fibre is heated at a constant heating rate (v) to 180 °C under constant loading (P). Then we monitor changes in deformation with time at constant temperature. The deformation grows and tends to some limiting values. These values depend on the network density. The evaluation of network density requires extrapolation of experimental data far beyond the observation area. In such a situation the application of simple 'soft' models may cause large errors. Therefore we used a 'hard' model based on the well-known equation of ideal network deformation [25]:

$$\frac{P}{S} = G\left(Y - \frac{1}{Y^2}\right) \tag{59}$$

Here P is the loading, S is the cross-sectional area of undeformed fibre, $Y = L/L_0$ is the ratio of the current sample length to the initial one and G is the elastic modulus, which is proportional to the total concentration of network knots. Further, it was assumed that the network consists of two kinds of knots: chemical knots, i.e. cross-links which were formed during radiation curing, and physical knots caused by molecular interaction [26]. Thus the joint modulus G is represented as a sum of two components: chemical G^{c} and physical G^{p}. Physical knots degrade under loading in the course of the experiment. The analysis of experimental TMA curves has shown that the kinetic model for the destruction of physical knots may be represented as the sum of fast and slow exponential terms:

$$G^{\mathrm{p}} = G^{\mathrm{p}}_{10}\exp(-k_1 t) + G^{\mathrm{p}}_{20}\exp(-k_2 t), \quad k_1 \gg k_2 \tag{60}$$

In this equation the parameter G^{p}_{10} is proportional to the concentration of weaker physical knots which degrade quickly with a large kinetic constant k_1, while the parameter G^{p}_{20} is proportional to the concentration of stronger knots with a small destruction constant k_2. Equation (60) is a typical 'soft' model, as the number of exponential terms in it is limited only by the calculation possibilities (again, multicollinearity is a companion of the 'soft' model). Thus, using the 'hard' equation (59) and the 'soft' one (60), we obtain the final model, which is an implicit function:

$$0 = \frac{S}{P}\left[G^{\mathrm{c}} + G^{\mathrm{p}}_{10}\exp(-k_1 t) + G^{\mathrm{p}}_{20}\exp(-k_2 t)\right]\left(Y - \frac{1}{Y^2}\right) \tag{61}$$

In this equation the variable Y is the response to be evaluated and the variables t, P and S are predictors. It is essentially a non-linear model with five unknown parameters. The parameter G^{c} is

```
        ' TMA Model
0=S/P*A*(y-1/y^2)-1;   1<y<1.5
   A=G+G1*exp(-k1*t)+G2*exp(-k2*t)
      S=0.8659
          G=?
          G1=?
          k1=?
          G2=?
          k2=?
```

Figure 14. TMA model as presented on worksheet.

Table IV. Estimates of chemical modulus G^c for different experiments

Series	ν (°C min^{-1})	P (g)	s^2	G^c (g mm^{-2})
1	10	10	$1{\cdot}441 \times 10^{-7}$	$10{\cdot}609 \pm 0{\cdot}008$
2	10	7	$12{\cdot}03 \times 10^{-7}$	$8{\cdot}312 \pm 0{\cdot}038$
3	5	5	$0{\cdot}075 \times 10^{-7}$	$8{\cdot}707 \pm 0{\cdot}026$
4	10	5	$0{\cdot}649 \times 10^{-7}$	$8{\cdot}708 \pm 0{\cdot}046$
5	15	5	$1{\cdot}915 \times 10^{-7}$	$8{\cdot}587 \pm 0{\cdot}066$
All		—	$7{\cdot}871 \times 10^{-7}$	$8{\cdot}328 \pm 0{\cdot}029$

the required characteristic of network density. The other parameters $(G_{10}^p, k_1, G_{20}^p, k_2)$ are outline our attention. Model (61) is shown in Figure 14 as it is presented on the worksheet.

In solving the problem of extrapolation out of the observation area, model adequacy acquires the primary importance. In our case, model verity is proved by the independence of the estimated G^c value on the conditions of measurement, i.e. loading and heating rate. Figure 15 shows TMA curves

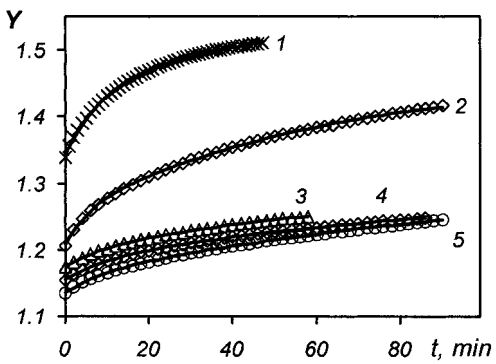


Figure 15. TMA data and fitting curves for PE fibre irradiated by 10 Mrad dose. Experimental conditions are given in Table IV.

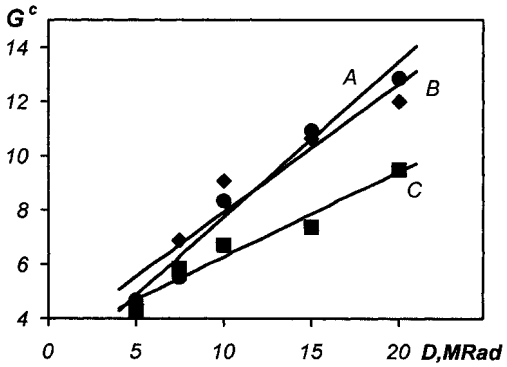


Figure 16. Dependence of chemical modulus G^c (g mm^{-2}) on irradiation dose for three different grades of PE fibre.

of the irradiated fibre measured at different conditions. The points on the plot are experimental data and the curves are due to model fitting. The results in Table IV show that the required parameter G^c really does not depend on the conditions of measurement. The final G^c value was obtained by joint processing of all five curves with application of successive Bayesian estimation. This is shown in the last row of Table IV. The dependence of the parameter G^c on the irradiation dose for three different grades of PE is shown in Figure 16. The optimum dose was determined by interpolation of these data. For this purpose we used the simple linear model

$$G^c = G_0^c + qD$$

Fibre A is used as a standard. For example, if for fibre A the standard dose is 10 Mrad, then fibre *C* should be irradiated with a dose of 18 Mrad to obtain the same network density.

This example shows the efficiency of a proper combination of 'hard' and 'soft' models in practice. Very often such a combination helps to reduce the number of experiments and to obtain much valuable information. It is important to use a reliable NLR tool to estimate the parameters of the model. The model may have a very complicated mathematical form but a clear physical basis.

10. CONCLUSIONS

We have presented our ideas and methods for non-linear regression analysis. The goal is to show that this technique lets us find solutions for many chemical problems. The main conclusions can be drawn as follows.

- The choice of model remains the main trouble in NLR application. Regarding this problem, we want to emphasize that a non-linear model may be 'soft' and a linear one may be 'hard'. Very often a model may be constructed like a nut with a hard shell and a soft kernel. The TMA example illustrates this idea.
- With respect to the extrapolation problem, non-linear regression can be considered as the primary method. Contrariwise, a linear model is more helpful in interpolation.
- We believe that there are no specific problems in estimation and interpretation of NLR models. All difficulties are similar to those in linear analysis.
- We propose a new concept of coefficient of non-linearity as a random value and show the criterion that helps to estimate the non-linearity of the model.

- The presented examples demonstrate that NLR analysis can yield interesting results in chemometric applications. Especially we recommend using it for analysis of dynamic methods with variable temperature (TGA, DSC and TMA).
- The general suggestion is that one should not hesitate to apply NLR where a non-linear model is prescribed by the nature of the investigated phenomenon. Modern NLR tools allow one to handle it as easily as a linear one.

ACKNOWLEDGEMENTS

We gratefully acknowledge the support from the Norwegian and Finnish Chemometrics Societies in granting our participation in SSC-6. The authors would also like to thank Professor Noel Overberg (NV Raychem SA, Belgium) and Dr Alexander Kruchkov (Cable Institute, Russia) for their assistance in this work.

APPENDIX

List of Symbols

$\mathbf{a} = (a_1,\ldots,a_p)^{\mathrm{T}}$	vector of unknown parameters (Equation (9))
$\hat{\mathbf{a}} = (\hat{a}_1,\ldots,\hat{a}_p)^{\mathrm{T}}$	vector of parameter estimates (Equation (21))
$\hat{\mathbf{a}}^* = (a_1^*,\ldots,a_p^*)^{\mathrm{T}}$	vector of generated parameter estimates (Equation (51))
$\mathbf{A}$	Hessian matrix ($p \times p$)(Equation (35)–(37))
$\mathbf{b} = (b_1,\ldots,b_p)^{\mathrm{T}}$	vector of prior parameters (Equation (16))
$B(\mathbf{a})$	Bayesian term (Equation (23) and (24))
$\mathbf{C}$	variance–covariance matrix ($p \times p$) (Equation (34))
$\mathbf{F}$	**F** matrix ($p \times p$) (Equation (39))
$\mathbf{H}$	prior information matrix ($p \times p$) (Equation (17))
$\mathbf{I}$	unit matrix ($p \times p$)
n_0	prior number of degrees of freedom (Equation (19))
N	number of observations
N_f	number of degrees of freedom by fit (Equations (29) and (30))
N_{w}	number of observations with non-zero weights
p	number of parameters
$Q(\mathbf{a})$	objective function (Equation (26)–(28))
s^2	estimate of weighted error variance (Equation (31))
s_0^2	prior value of weighted error variance (Equation (18))
$S(\mathbf{a})$	sum of squares (Equation (22))
$\mathbf{w} = (w_1,\ldots,w_N)^{\mathrm{T}}$	vector of weights (Equation (8))
$\mathbf{x} = (x_l,\ldots,x_m)^{\mathrm{T}}$	vector of settings (predictors) (Equation (6))
$\mathbf{y} = (y_1,\ldots,y_N)^{\mathrm{T}}$	vector of observations (response) (Equation (7))
γ	rough coefficient of non-linearity
Γ	exact coefficient of non-linearity (Equation (58))
$\boldsymbol{\varepsilon} = (\varepsilon_1,\ldots,\varepsilon_N)^{\mathrm{T}}$	vector of measurement errors (Equation (12))
σ_ϵ^2	weighted error variance (Equation (15))

REFERENCES

1. Brown SD, Sum ST, Despagne F, Lavine BK. Chemometrics. *Anal. Chem.* 1996; **68**: 21R–61R.
2. Farrar DE, Glauber RR. Multicollinearity in regression analysis. The problem revisited. *Rev. Econ. Statist.* 1967; **49**: 1.

3. Marquardt DW. An algorithm for least squares estimation of nonlinear parameters. *SIAM J.* 1963; **11**: 431–441.
4. Chambers JR. Fitting nonlinear models: numerical solutions. *Biometrika* 1973; **60**: 1–13.
5. Efron B. Bootstrap methods: another look at the jackknife. *Ann. Statist.* 1979; **7**: 1–26.
6. Efron B. Better bootstrap confidence intervals. *J. Am. Statist. Asscc.* 1987; **81**: 19–28.
7. Pomerantsev AL. Confidence intervals for nonlinear regression extrapolation. *Chemometrics Intell. Lab. Syst.* 1999; **49**: 41–48.
8. Fitter Add-In. [Online]. http://polycert.chph.ras.ru/fitter.htm [1 June 2000].
9. Bysrtitskaya EV, Pomerantsev AL, Rodionova OYe. Prediction of the aging of polymer materials. *Chemometrics Intell. Lab. Syst.* 1999; **47**: 175–178.
10. Crank J. *The Mathematics of Diffusion*. Clarendon Press: Oxford, 1975.
11. Abramowitz M, Stegun IE. *Handbook of Mathematical Functions*. National Bureau of Standards: 1964.
12. Borisov BI. Study of PVC film aging in soil. *Zh. Prikl. Khim.* 1970; **43**: 1116–1120.
13. Bard Y. *Nonlinear Parameter Estimation*. Academic Press: New York, 1974.
14. Fletcher R. Function minimization without evaluating derivatives—a review. *Comput. J.* 1965; **8**: 33–41.
15. StRD Nonlinear Least Squares Regression Datasets. [Online]. http://www.nist.gov/itl/div898/strd/nls/nls_main.shtml [1 June 2000].
16. More JJ, Garbow BS, Hillstrom KE. Testing unconstrained optimization software. *ACM Trans. Math. Softw.* 1981; **7**: 17–41.
17. Pavlov BV, Povzner AYa. A method for the numerical integration of systems of ordinary differential equations. *Zh. Vichisl. Mat. Mat. Fiz.* 1973; **13**: 1056–1059.
18. Shlyapnikov YuA. Polyolefine chain oxidation. In *Development in Polymer Stabilization.* vol. 5. Applied Science Publishers: London 1981; 1–22.
19. Maksimova GA, Pomerantsev AL. Successive Bayesian estimation of regression parameters. *Zav. Lab.* 1995; **61**: 432–435.
20. Barra JR. *Notions Fondamentales de Statistique Mathematique*. Dunod: Paris, 1971.
21. Beale EML. Confidence regions in nonlinear regression. *J. R. Statist. Soc. B* 1960; **22**: 41–75.
22. Guttman I, Meeter DA. On Beale's measures of nonlinearity. *Technometrics* 1965; **7**: 623–637.
23. Bates DM, Watts DG. Relative curvature measures of nonlinearity (with discussion. *J. R. Statist. Soc. B* 1980; **40**: 1–25.
24. Bates DM, Watts DG. Parameter transformations for improved approximate confidence regions in nonlinear least squares. *Ann. Statist.* 1981; **9**: 1152–1167.
25. Flory PJ. *Principles of Polymer Chemistry*. Cornell University Press: New York, 1953.
26. Ferry JD, Kan HC. Interpretation of deviations from neo-Hookean elasticity by a two-network model with crosslinks and trapped entanglements. *Rubber Chem. Technol.* 1978; **51**: 731–737.

