ELSEVIER

Chemometrics and Intelligent Laboratory Systems 28 (1995) 73–87

Chemometrics and intelligent laboratory systems

Robust regression and outlier detection for non-linear models using genetic algorithms

P. Vankeerberghen [a], J. Smeyers-Verbeke [a], R. Leardi [b], C.L. Karr [c], D.L. Massart [a,*]

[a] ChemoAc, Pharmaceutical Institute, Vrije Universiteit Brussel, Laarbeeklaan 103, B-1090 Brussels, Belgium
[b] Istituto di Analisi e Tecnologie Farmaceutiche ed Alimentari, Via Brigata Salerno (Ponte), I-16147 Genova, Italy
[c] US Bureau of Mines, Tuscaloosa Research Center, P.O. Box L, University of Alabama Campus, Tuscaloosa, AL 35486-9777, USA

Received 9 September 1994; accepted 20 January 1995

Abstract

Experimental data such as calibration and pharmacokinetic data can be contaminated with outliers. Robust regression based on the calculation of the least median of squared residuals (LMS) is robust to the presence of outliers and is used for outlier detection. The original LMS program only handles models which are linear in the parameters. A genetic algorithm can be used to obtain the LMS parameters for models that are non-linear in the parameters. In this work the feasibility of using genetic algorithms for LMS is demonstrated by means of curved analytical calibration and pharmacokinetic data contaminated with outliers.

1. Introduction

The least median of squares (LMS) regression method for non-linear models involves basically the search for the lowest median of squared residuals. The objective function landscape is always multimodal which requires a global search method to obtain the LMS parameters. Two global optimising methods used in chemometrics are simulated annealing and genetic algorithms. Since both require a proper configuration, we prefer to study only one method which is the most appealing to us, i.e. the genetic algorithms.

In a first theoretical part of this paper, robust regression, least median of squares regression and genetic algorithms (GAs) will be discussed in more detail. The second part deals with the application of both GAs and LMS to a curved calibration and a pharmacokinetic model.

2. Robust regression and the least median of squares method

Least sum of squares (LS) may strongly be influenced by the presence of outlier(s). Robust regression has been developed to handle contaminated data,

[*] Corresponding author.

SSDI 0169-7439(95)00010-0

i.e., data containing outliers. The difference between least squares and a robust regression method is illustrated in Fig. 1.

Many of the robust methods are based on the median statistic. Among them are the single median [1], the repeated median [2] and the least median of squares [3] methods. The first two methods were developed for straight line models, LMS for models linear in the parameters. The breakdown point as defined by Hampel [4] (i.e., the smallest fraction of contaminated data which leads to large deviating model parameters) changes from 0% for LS to 50% for LMS. Practically, this means that up to 50% of the data set may be contaminated with outliers before LMS breaks down and produces aberrant parameter estimations.

Recently, Koscielniak [5] developed robust regression procedures based on the single and repeated median for calibration models which are non-linear in the parameters. These methods are computed without explicit optimisation function and provide initially no outlier detection procedures.

In this work the least median of squares method will be used since it has an optimisation function and it allows outlier detection.

In contrast with the LS principle which minimises the sum of squared residuals, ($\Sigma_{i=1}^{n}(res)^2$), LMS minimises the median of squared residuals ($med(res)^2$). The median is defined as the $[n/2]+1$ ranked value.

Once the robust model is obtained, outlier detection can be applied. A robust estimator of the pure error is computed from the median of squared residuals and is compared with each residual to define whether the corresponding data point is outlying or not.

The robust estimator of the pure error is obtained in two steps. In the first step, an initial scale estimate is calculated by means of the following equation [3]:

$$s^0 = 1.4826\left(1 + \frac{5}{n-p}\right)\sqrt{med\left(res_i^2\right)} \quad (1)$$

where n is the number of data points, p the number of parameters, res_i the LMS residuals. In a second step, the LMS residuals of the data points are compared with this initial scale estimate. When a residual is larger than 2.5 times s^0, the data point is omitted for the computation of the final scale estimate s^*. This value is calculated by means of the following equation:

$$s^* = \sqrt{\frac{\Sigma_i^{n^*}\left(res_i^2\right)}{n^* - p}} \quad (2)$$

with n^* the number of data points retained.

Finally, when the LMS residual of a data point is

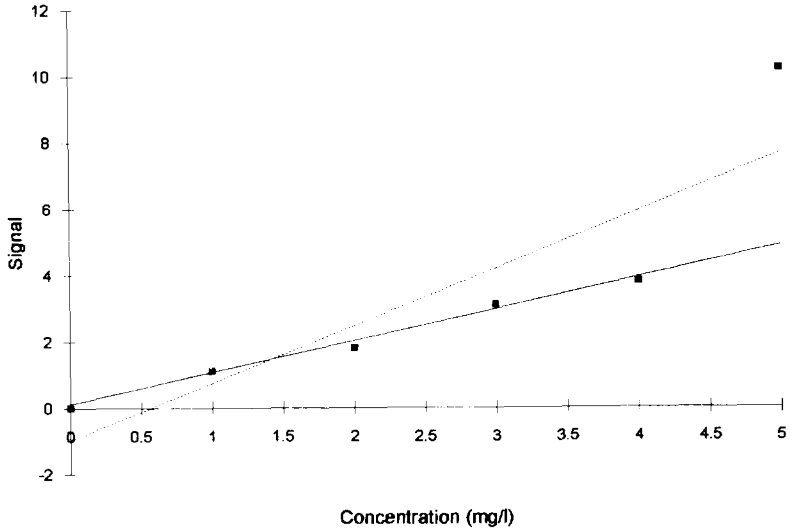

Fig. 1. Straight line with an outlier. The difference between a LS regression line (dotted line) and LMS regression line (full line) is obvious.

2.5 times larger than s^*, the point is considered to be an outlier.

3. Genetic algorithms

Genetic algorithms [6,7], abbreviated as GAs, are global search or optimization techniques. The principle will be briefly explained. The GA advances through sets of candidate solutions to the problem. Information is exchanged between the better solutions, to produce improved candidate solutions. A stochastic component in the search for the global optimum introduces single changes in the candidate solutions, with a relatively low probability. The exchange of information and the stochastic operator produce better and worse candidate solutions. The bad solutions are removed from the set and the better ones are copied to the next set. This whole process continues until the optimum is achieved. The major strengths of GAs are the following.

(1) Robustness towards local optima because many diverse points are examined at the same time [8].

(2) The power to search a large parameter space efficiently. A relatively small number of function evaluations of a large search space suffices to locate the near-global optimum [8].

(3) To our own experience, for the same type of problem, a GA requires no assumptions to be made about the geometry of the search space.

The disadvantages of GAs include:

(1) Configuration problems [8]: due to the lack of a master plan on how to construct a GA, one only has a few guidelines available on how to build and configure a GA. The main reason for this is that issues such as for example the type of crossover operator and the type of parameter coding are problem related. Configurations published in the literature are only indicative for the same type of problem, as is explained further.

(2) Premature convergence [9]: a few very fit, but non-optimal candidate solutions dominate the new set (with many copies) in such a way that diversity is lost. As a result, the GA is incapable to evolve from the local optimum.

(3) Slow finishing [9]: from most performance curves where the fitness is plotted as a function of the number of generations, one notices that a GA is good to obtain a reasonable solution in a very short time, but it takes more time to obtain the global optimum.

4. Description of the two test models

4.1. Non-linear calibration model for flame atomic absorption spectrometry (FAAS)

Atomic absorption calibration lines can be curved, especially when they are applied over an extended working range.

A hyperbolic model [5]

$$y=\frac{a_0+a_1x}{1+a_2x} \tag{3}$$

where a_0, a_1 and a_2 are the parameters to be defined can then be applied. Robust regression can, in principle, be used to locate outliers, but the original LMS program *Progress* [10] can only handle linear models. GAs are applied here to find the three parameters according to the LMS criterion. The search for the LMS parameters is then followed by the LMS outlier detection procedure.

4.2. The pharmacokinetic model

Consider the extra-vascular administration of a drug. The concentration in the compartment C_p at any time is given by the equation of Bateman:

$$C_p=\frac{FA_0k_a}{V_d(k_a-k_e)}(e^{-k_et}-e^{-k_at}) \tag{4}$$

where A_0 represents the dose of the drug, F the fraction absorbed or the bioavailability, V_d the distribution volume, t the time, k_a and k_e the absorption and elimination constants.

In a pharmacokinetic study, a patient is given a known amount of drug and the concentration levels in the blood are measured over a period of time. Curve-peeling is used to obtain the three parameters F/V_d, k_a and k_e. This method is based on a log transformation of the exponential model to a straight line model and on classical LS regression. Since the measurements are subject to random or to gross errors this procedure may lead to erroneous results when outliers are not detected and removed. Here we will apply LMS. The LMS parameters are estimated by a GA.

5. Experimental

The GA was written in the C language using the Borland C+ + compiler version 3.1 and runs on a 486 DX2 based PC. Data sets for both models were simulated by (i) adding homoscedastic normal distributed noise and (ii) manually introducing outliers. The noise is generated by the transformation of uniformly distributed pseudo-random numbers to normal distributed values as described by Box and Muller [11] and as implemented in Ref. [12]. The uniform pseudo-random numbers were generated with the Borland library function based on a multiplicative congruential generator with a period of 2^{32}.

One additional real data set was obtained for the pharmacokinetic model. SPSS for Windows version 6.0 was used for LS non-linear regression.

6. Results

6.1. Construction and configuration of the GA

As explained before, the bottleneck of applying GAs is its configuration [8]. One of the solutions is to look at the GA configurations in a field close to LMS. In the past GAs have been used for least squares linear and non-linear regression [13–16].

The following items will be discussed: the type of coding to use, the objective function, the selection operator, preservation of the best candidate solution, mating, the crossover operator and probability of mutation.

6.1.1. Coding of the parameters

Basically, two types of coding can be used: binary and floating point coding. Here, a variant of binary coding, namely Gray coding, is used.

In Gray coding one starts from 000 for the integer value 0 and flips successively the right-most bit. The integer 1 is coded as 001 (flip the right-most bit). The integer 2 is coded as 011 (the most right bit should not be flipped as it would result in an existing pattern, thus flip the second right bit). The value 3 is consequently coded as 010 (flip the most right bit because it produces a new pattern). The integer series from 0 to 7 is then coded as 000, 001, 011, 010, 110, 111, 101, 100. In contrast with binary coding, neighbouring integer values are represented in Gray coding by a bit pattern which differs by *one bit only*. The practical consequences of Gray coding were pointed out by Lucasius [8]. A smooth landscape with decimal parameters appears more irregular when it is coded in the binary mode than it would be when Gray coding is applied. Furthermore, on the average, a single mutation in a Gray coded candidate solution results in a smaller perturbation of the parameter and thus of the fitness value than occurs with binary coding [8].

Common for binary and Gray coding is the need for scaling. Consider a parameter coded as 11 bits which allows a range from 0 to $2^{11} - 1 = 2047$. Suppose the real domain range of the parameter is from 0 to 0.200, then dividing the integer by 1×10^4 yields the required range. Consequently, the resolution of the parameter is then 1×10^{-4}.

The second type of coding is floating point coding, used by Hibbert [14] and Michalewicz [17]. In favour of it are a much larger domain and a larger precision, removing the need to scale the parameter. A negative aspect is that the number of bits for each parameter is fixed, 32 for single and 64 bits for double precision.

From experience or from literature, the range of each parameter is known which makes the use of Gray coding possible. With 11 bits for each parameter and taking into account the range for each parameter, at least three significant digits are available which is sufficient for our problem. As there are three parameters to optimise for each model, the string length (33 bits) is almost three times smaller than with single precision floating point coding (string length: 96 bits). Single precision floating point coding was implemented but performed in comparison with Gray coding, as expected, worse. Larger strings are obviously much harder to optimise.

Gray coding was applied due to (i) an improved search performance, (ii) a smaller disturbing effect of a single mutation and (iii) an adjustable number of bits for each parameter, despite the need for scaling.

6.1.2. Objective function

As the objective of a GA is to maximise (higher fitness values correspond with fitter candidate solutions) and that of LMS is to minimise, we take the reciprocal of the median of squared residuals. Candidate solutions resulting in lower median of squares residuals obtain a higher fitness value. Care has been taken to avoid dividing by zero for the pharmacokinetic problem which occurs when k_a equals k_e. Therefore, the following reasoning is applied. First, k_a is compared with k_e. In case of no equality, the fitness function is computed. When the two parameters are equal, the fitness function cannot be computed. Instead, the value zero is assigned to the fitness value. Due to this low fitness value, this candidate solution (with k_a equals k_e) does not appear anymore in the next generation. A similar transformation is applied for the denominator of the calibration model.

6.1.3. Selection of the good candidate solutions

The selection of the candidate solutions is proportional to their fitness value. A candidate solution whose fitness value is below average is not copied in the next set, each candidate as good as the average fitness is copied in the next set, each candidate solution three times as good as average, is copied three times. Because of truncation errors, one can only copy an integer number of candidate solutions so that the population size decreases. To retain a constant population size, the following mechanism was devised: after the process of proportional copying, the other candidate solutions are selected at random until the predefined population size is obtained.

Two other options have been tried, known to prevent premature convergence. The first one is called rank-based selection. Here, the fittest solutions receive the most copies, followed by the second fittest and so on. This process discards the actual differences in fitness between the candidate solutions, e.g., it may be that the best is 10 times fitter than the second best and 100 times as fit as the average fitness: the best one will never obtain 100 copies, but only a small predefined number. This selection scheme was evaluated but no significant improvements could be found. The second one is binary tournament selection. This scheme selects the best of two randomly chosen candidate solutions and iterates until the new population is full. This scheme also performed worse.

6.1.4. Preservation of the best candidate solution

To prevent the loss of the best candidate solution during crossover or mutation, one proceeds as follows: at the stage of the selection of the good candidate solutions where the fittest candidate solution received the largest number of copies, one copy is set aside and placed back in the population after crossover and mutation occurred for the $n-1$ number of remaining candidate solutions. As a result the information of the fittest candidate solution is also processed (crossover and mutation) and the highest fitness value at each generation cannot decrease in function of the number of generations.

6.1.5. Mating

Selection of the parents which are going to mate is at random, but with the pre-requisite that the parameter values are different. A crossover between almost two identical candidate solutions is not productive since the resulting candidate solutions would not be distinct from the original solutions. Only 90% of the candidate solutions in the set mate.

6.1.6. Crossover operators

The following problem in configuring the GA is the choice of crossover operator. Basically, two types can be applied for numerical parameter optimisation; the multiple-point (B_MX) and the uniform crossover operator (B_UX). The simplest version of the B_MX series is the single-point crossover operator, B_1X, used for educational purposes. A slightly more complicated version is the two-point crossover operator B_2X.

The second type of operator is the B_UX uniform crossover operator. Here, each bit has a certain probability to be swapped which results in a larger information exchange than the B_MX operator.

Lucasius [8] proposes to apply the uniform crossover operator, B_UX, for problems where the parameters to be optimised are strongly correlated. Experiments carried out by us confirmed the superior quality of the uniform crossover operator (with bit swap rate 0.30) over the single- and multi-point crossover operators, B_1X and B_2X.

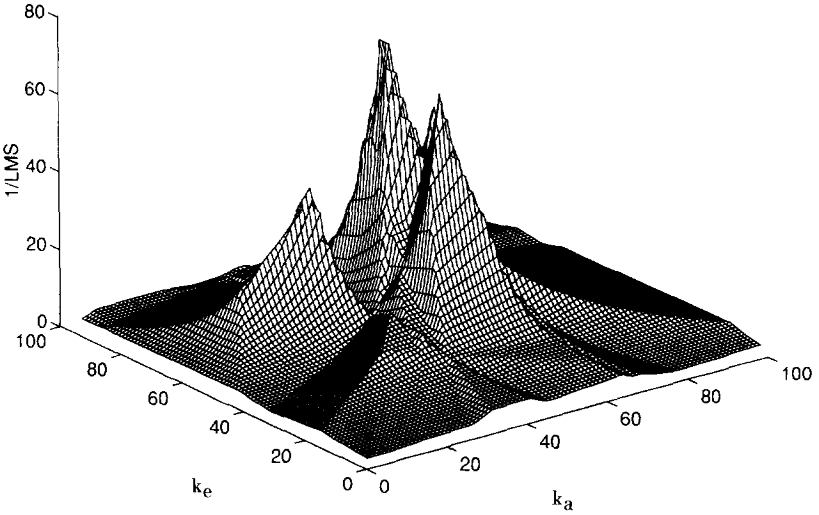

Fig. 2. LMS fitness landscape for the pharmacokinetic model with one outlier at the end. The reciprocal of the LMS value was plotted as a function of the absorption and elimination coefficients.

6.1.7. Probability of mutation

The settings for the mutation operator were one of the most difficult parameters to optimise. The fitness landscape is plotted for the pharmacokinetic model in Fig. 2. The first term of Eq. (4), i.e., the ratio of the bioavailability, dosis and the extrapolated distribution volume is kept constant to allow graphical representation. The irregular surface with local optima is very different from the smooth LS surface, for the same data set as shown in Fig. 3. This indicates that the LMS problem is much harder to solve than the LS problem which is in agreement with the conclusions of Rousseeuw and Hawkins [10,18]. Where gradient based methods can be used for LS, it is obvious that

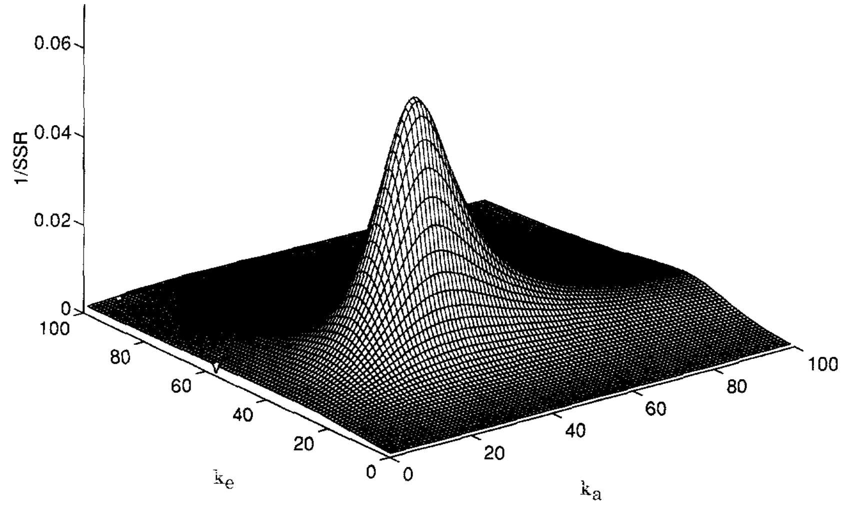

Fig. 3. LS fitness landscape for the same pharmacokinetic model as in Fig. 2 with one outlier at the end. The reciprocal of the sum of squared residuals was plotted as a function of the absorption and elimination coefficients. Notice the two loci with fitness value equal to 0.0 since k_a yields the same value as k_e.

they would fail for LMS, except when the global optimum is approximately known so that one reduces the experimental area so that only the best of the three peaks is included.

LMS tries to fit the majority of data points as well as possible. Multiple subsets of data points are possible which lead to a multimodal landscape. Only one subset has the least median of squared residuals which is then the global optimum. A careful investigation of the fitness surface for the two-parameter problem reveals that local optima are present in the neighbourhood of the global optima. For higher dimensions, one may imagine that the global optimum is surrounded by local optima which is known as a deceptive problem [7].

To obtain the plot of Figs. 2 and 3, a full grid search with a rough resolution to save computing time has been carried out. This yields the global optimum with the LMS parameters which will be used to check the optimum provided by the GA. A GA equipped with the operators as explained above including a population size of 200 candidate solutions, 15 bits/parameter and a mutation rate of 1/string length = 1/30 (2 parameters of 15 bits each), finds the LMS optimum without difficulties. After introduction of the third parameter which had been kept constant, the GA fails to produce the optimum.

Deceptive problems are difficult to optimise, even with a GA. The reason for this might be the following. Consider the fitness surface for the *two*-parameter problem as illustrated in Fig. 4. The range of both parameters measures only a few steps in the resolution of the GA. The starting point is a local optimum where the population has almost converged. When changing parameter k_e in the direction of the dotted arrow, towards the global optima, the fitness value decreases below average and the candidate solution is removed from the population. The same occurs when adjusting parameter k_a from the local towards the global optimum (full arrow). The global optimum can only be obtained when *both* parameters are changed, at the same time (dashed line). A higher mutation rate is needed, not to introduce more diversity but to move both parameters simultaneously in the direction of the global optimum. When we applied a mutation rate of two mutations per candidate solution, good results were obtained. Traditionally, large mutation rates, expressed as the number of mutations for each candidate solution, destroy almost good solutions, but in our case, it is necessary to move both parameters to the global optimum.

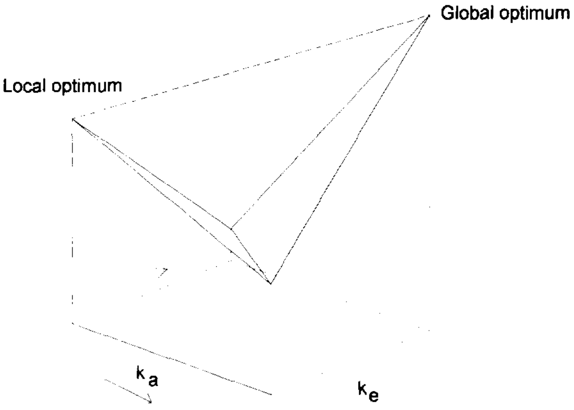

Fig. 4. Illustration of the deceptive problem. The global optimum can only be reached from the local one when *both* parameters are changed at the same time, in the direction of the global optimum.

Let us look again at our three-parameter model. The initial GA configuration for the two-parameter problem was adequate to deal with the multimodal but non-deceptive LMS landscape. On the other hand, when the third parameter was introduced, the GA, even with the larger mutation rate, converged sometimes around a rather large local optimum. When we compared the solution (i.e., the three model parameters) with that of the global optimum, we noticed a difference in all three parameters, which confirms our initial conclusion of a deception problem. When the mutation rate is enlarged, the three parameters can be changed in one candidate solution at the same time and the population migrates from the local optimum to the global solution. The deceptive topology remains but the GA with the larger mutation rate is now capable to escape from local optima.

A final note should be made. The investigation of the fitness landscape is only partially relevant to understand the difficulties which the GA has to deal with. Indeed, the GA works on the Gray and not on the decimal representation of the problem, but in this case the investigation is valid since both are closely related due to the adjacent property of Gray coding.

6.1.8. Population size

A larger population size allows more diversity than a small one and processes more information in order

Table 1
Details of the GA

Objective function	Maximise $1/med(res^2)$
Coding	Gray
Parameters	3
Bits/parameter	11
P_{mating}	0.90
Crossover operator	B_UX
Bit swap rate	0.30
Mutation rate	0.06
Selection of survivors	Proportional to their fitness
Population size	500

to find the optimum after a smaller number of generations. On the other hand, a lower number of generations can be produced in the same time. A population size of 500 was found acceptable. At each generation, the GA prints the best solution.

The configuration of the GA is summarised in Table 1.

6.2. Performance of the GA

6.2.1. Comparison of LS non-linear regression using the GA and non-linear regression using classical techniques

The GA was set up for the least squares problem by (i) changing the fitness function and (ii) assigning the mutation rate to 1/string length since the search for the LS solution is much easier than for LMS. Pharmacokinetic data sets were simulated and both the GA and non-linear regression applied. As expected, no significant difference could be found. Marques [15] used GA for LS modelling of the chromatographic behaviour as a function of the pH. Their findings that a GA does not need very good initial estimates could be confirmed for most cases. Some LS problems have multiple identical optima without a global optimal solution. Lewi [19] presents a data set (Table 2) for the pharmacokinetic model where the absorption and elimination constants differ a factor two. For the example given two equivalent solutions

Table 2
Illustration of a non-linear model where the initial guesses are important. Data simulated with $k_a = 1.72$/h, $k_e = 0.900$/h and $FA_0/V_d = 26.67$ mg/l. The standard deviation of the noise generator yields 0.030. Clearly, two equivalent (in SSR) solutions have been found

Point	Time (h)	Conc.
1	0.05	2.1314
2	0.25	8.3018
3	0.50	11.9979
4	1.00	12.7163
5	1.50	10.3118
6	2.00	7.4881
7	3.00	3.4210
8	4.00	1.4681
9	6.00	0.2662
10	9.00	0.0454
11	12.00	0.0047
12	15.00	0.0125

Non-linear regression (Marquardt) yields:

	k_a (1/h)	k_e (1/h)	FA_0/V_d (mg/l)	SSR
Initial guesses	1.7	0.200	30.0	
Result	1.7091	0.90347	26.809	4.54021×10^{-3}
With other initial guesses				
Initial guesses	0.2	1.7	30	
	0.90347	1.7091	50.715	4.54021×10^{-3}

The scaling for the three model parameters of the GA was as follows: k_a: from 0.0 to 2.048 (1/h); k_e: from 0.0 to 2.048 (1/h); FA_0/V_d: from 0.0 to 102.4 (mg/l). Over different runs, the GA finds both solutions: SSR: 4.547×10^{-3}.

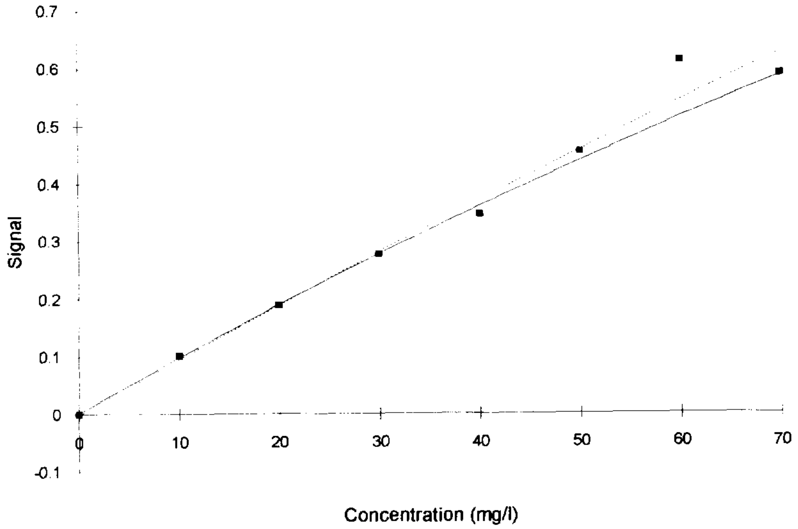

Fig. 5. Curved calibration data set contaminated with an outlier at the end. The outlier is found by LMS (full line).

have been found using non-linear regression of which only one is the real solution. No optimisation methods are available to solve these type of problems. The GA also found both solutions but the SSR (i.e., sum of squared residuals) was slightly higher than the SSR achieved by non-linear regression due to the limited resolution of the parameters. Since both optima are equivalent, also the GA cannot make a distinction between them. Therefore, domain knowledge has to be applied. Except for depot or retard drugs, the absorption rate is always larger than the elimination rate. Candidate solutions where the elimination rate exceeds the absorption rate are assigned a fitness value of zero and removed from the next generation.

6.2.2. *Comparison of GA-LMS with Progress*

To investigate the GA for its LMS performance, a curved model has been selected which can also be

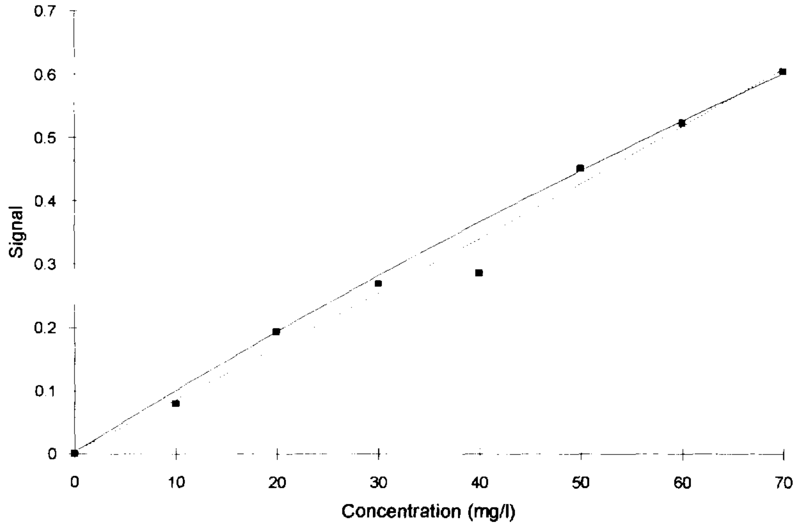

Fig. 6. Curved calibration data set contaminated with an outlier in the middle. The outlier is found by LMS (full line).

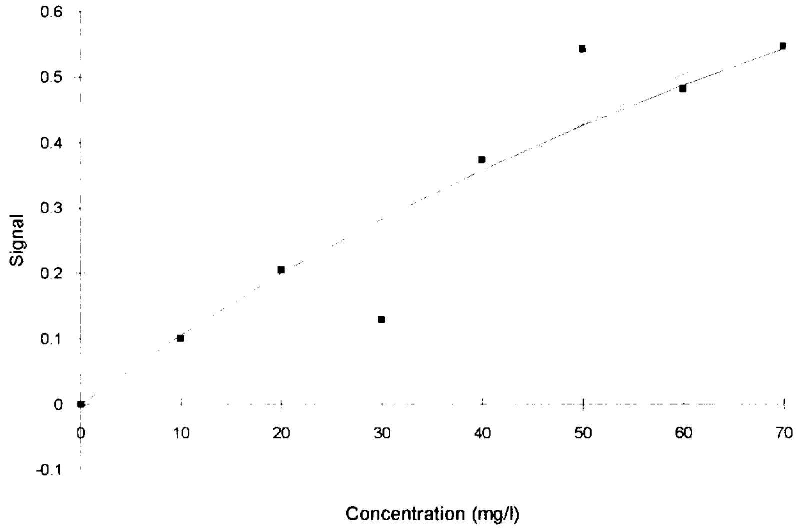

Fig. 7. Curved calibration data set contaminated with two outliers. The outliers are found by LMS (full line).

processed by the original LMS program Progress. This model consists of a second-degree polynomial for which a data set was generated with homoscedastic noise. An outlier was introduced manually in the data set. Both the GA and the original LMS program Progress processed the same data set. The GA found a lower, thus better LMS value than Progress (2.374×10^{-4} and 4.963×10^{-4}, respectively). This might be due to the fact that Progress skips an optimisation step for the estimation of the optimal intercept to save computing time.

6.3. Results for the curved calibration model

The GA was configured as in Table 1. The scaling of the three model parameters is from -0.02 to

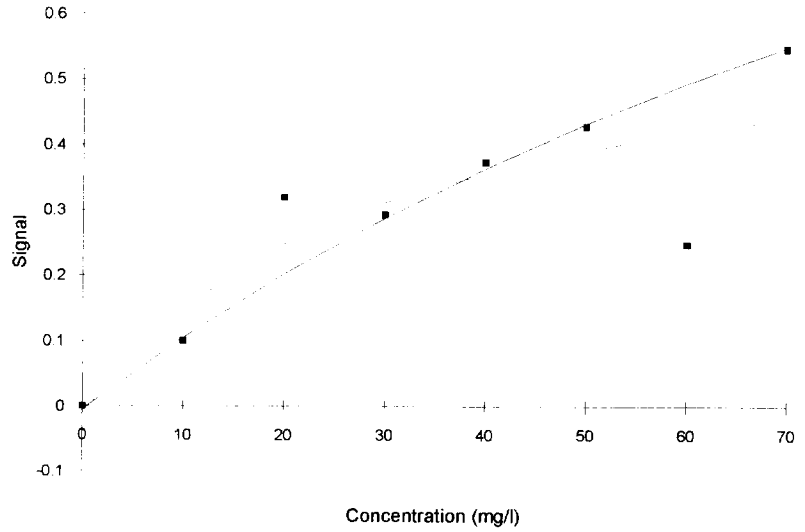

Fig. 8. Curved calibration data set contaminated with two outliers. The outliers are found by LMS (full line).

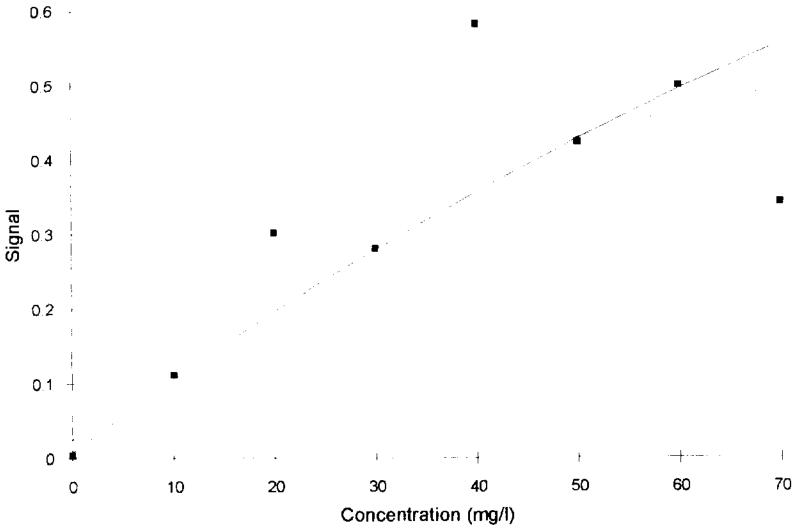

Fig. 9. Curved calibration data set contaminated with three outliers. The outliers are found by LMS (full line).

+ 0.02. The calibration line shown in Fig. 5 has been contaminated with a small outlier at the last but one position. As hoped, LMS defines the outlier. Another calibration data set (Fig. 6) has an outlier in the middle, again found by LMS. Two calibration lines, each in which two outliers were introduced, are shown in Figs. 7 and 8. The last two calibration lines have been contaminated with three outliers each, Figs. 9 and 10, respectively. The least squares regression line in Fig. 10 shows an inverted curvature due to the outlying data. For all calibration lines given, the outliers were detected.

The application of LMS to non-linear models shows no decrease in breakdown point as happens

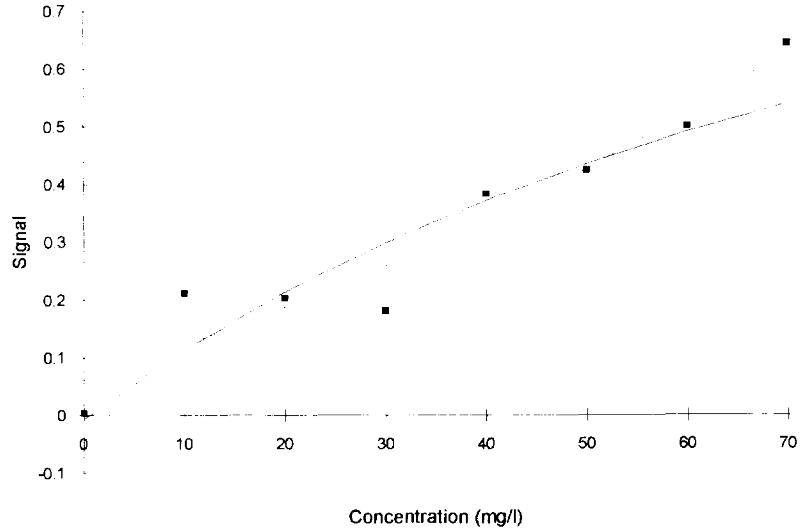

Fig. 10. Curved calibration data set contaminated with three outliers. The outliers are found by LMS (full line). The curvature of the least squares regression line is inverted by the outlying data.

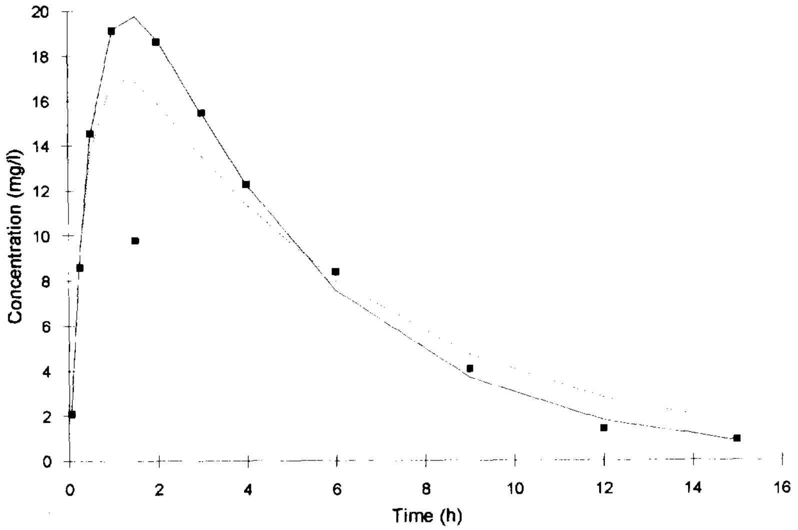

Fig. 11. Curve of pharmacokinetic data set B with an outlier, fitted by LMS (full line) and LS (dotted line).

with the repeated median. Three outliers were defined in a data set of eight points (Figs. 9 and 10), illustrating the breakdown point of 50%.

6.4. Results for the pharmacokinetic model

Two types of pharmacokinetic data sets have been analysed. The first one is the synthetic data set where outliers were introduced at different levels to allow a closer investigation. The results are presented in Table 3, together with the LS values. The scaling for the model parameters is from 0.0 to 2.048 for k_a, from 0.0 to 1.024 for k_e and from 0.0 to 51.20 for FA_0/V_d.

For case A where there are no outliers, the LMS parameter values are not significantly different from

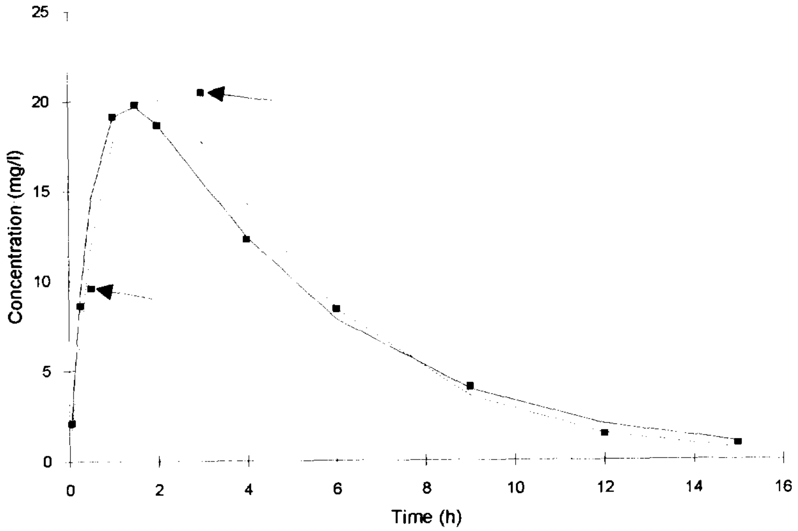

Fig. 12. Curve of pharmacokinetic data set F with two outliers, fitted by LMS (full line) and LS (dotted line). The outliers are indicated.

Table 3
Comparison of the pharmacokinetic parameters obtained by GA-LMS and classical least squares

	LMS			LS		
	FA_0/V_d (mg/l)	k_a (1/h)	k_e (1/h)	FA_0/V_d (mg/l)	k_a (1/h)	k_e (1/h)
A	27.25	1.688	0.2350	27.25	1.650	0.2320
B	27.58	1.651	0.2415	21.21	2.294	0.1761
C	27.58	1.651	0.2415	34.77	1.325	0.3012
D	27.25	1.688	0.2350	31.09	1.379	0.3154
E	27.25	1.688	0.2350	25.75	1.769	0.1896
F	26.98	1.708	0.2295	34.24	0.941	0.2921

A: No outlier. B: At the maximal concentration, but below the curve. C: At the maximal concentration, but above the curve. D: At the elimination step, below the curve. E: At the elimination step, above the curve. F: Two outliers, one at the absorption step, below the curve, and one at the beginning of the elimination step.

the LS parameters. For case B, illustrated in Fig. 11, with one negative outlier at the location corresponding with the maximal concentration, the LMS parameters remain almost unaffected, but the LS parameters are already very different. LMS defines the fifth and the ninth data point as outlying. The fifth point is certainly an outlier, but due to the fact that the majority of data points fit the LMS curve very well, the robust estimate of the pure error or the scale estimate is rather small and little deviating points are also marked as outlying.

The occurrence of outliers at other locations (C, D, E) in the curve does not alter the LMS parameters.

Fig. 12 shows the pharmacokinetic curve with two outliers. The third data point might not be noticed visually as outlying because the data point lies close to the curve in the x direction, but the residual in the y direction is very large. The seventh data point is the second outlier. LMS defines both points correctly as outlying.

The performance plot for the GA is shown in Fig. 13 where the least median of squares value for a data set with one outlier is given as a function of the number of generations. The global optimum is located after evaluating only a small fraction (5000 evaluations) of the whole search space (2^{33} or 8.59 × 10^9 possibilities).

A real pharmacokinetic data set was also obtained. To improve the quality of the solution and to increase the speed of the search for the solution, the GA works best in a well defined region in the parameter space, enclosing the global optimum. For this real data set, no information is available about this region which had to be found. The method of iterated search space contraction [8] was used to locate

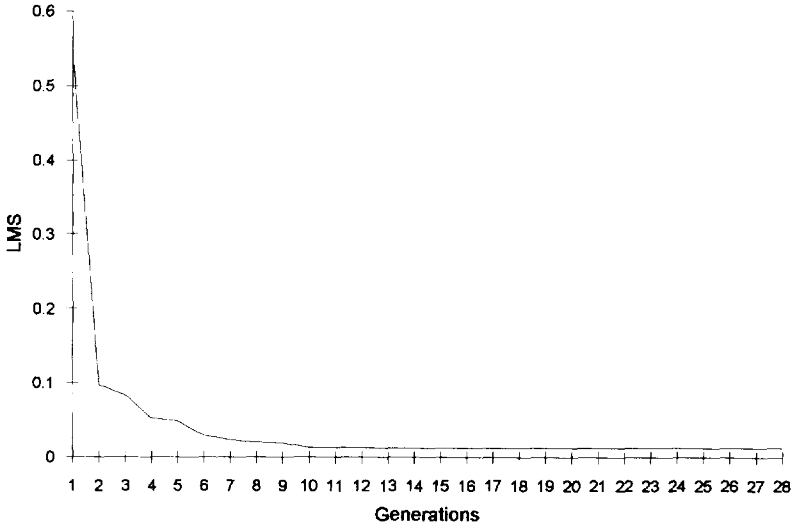

Fig. 13. Least median of squares value as a function of the number of generations. The global optimum is obtained after 10 generations (5000 function evaluations).

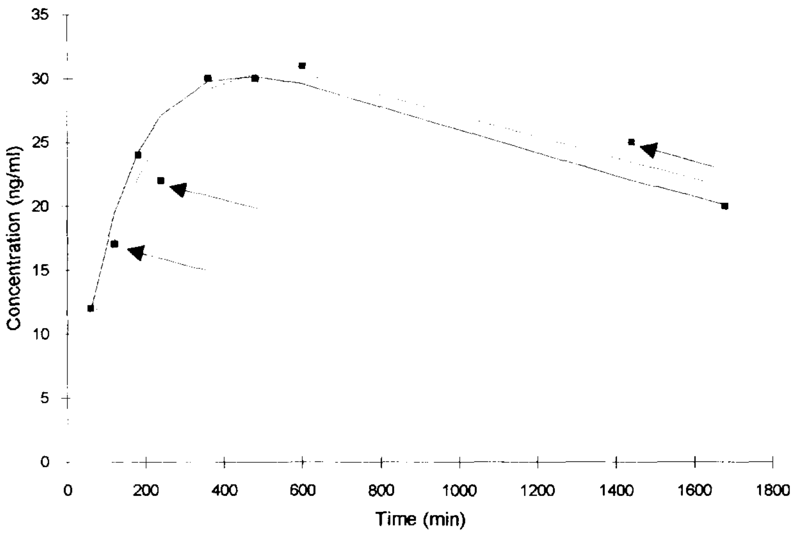

Fig. 14. Curve of a real pharmacokinetic data set, fitted by LMS (full line) and LS (dotted line). The three LMS outliers are indicated.

the optimum as follows. The GA was set up as described earlier, with 11 bits for each parameter, but with a larger scaling of the parameters. After a few runs of the GA with this rough resolution, an optimum can be located. In the next step, the scaling of the parameters, i.e., the range, is decreased resulting in a higher resolution. Again, the GA locates an optimum. This step-wise refinement of the resolution takes place until the model parameters have the required number of significant digits. One must take care that the global optimum is not eliminated from the search space due to a relatively coarse initial grid. The results are presented in Fig. 14. LMS found three outliers. One may question the validity of those outliers taking into account the biological nature and thus variance of the samples. This can be solved with a technique developed by Vankeerberghen[20]. First one defines a quality criterion such as for example the residual variance obtained by classical non-linear regression. When this variance is higher than the pre-established acceptable value, GA-LMS is applied. When LMS defines several points as outlying, one rejects the data point corresponding with the largest outlier from the data set. Then, the quality criterion is recomputed: when it is acceptable, there is only one outlier, the one which has been rejected. When the residual variance is still too high, GA-LMS is applied again, iteratively until all outliers responsible for the unacceptable residual variance are removed.

7. Discussion

The LMS problem is much more difficult to optimise than the LS problem. We noticed several times that the GA was searching in a direction which did not lead immediately to the global solution. Suddenly, after several thousand generations, the GA switched to the direction of the global optimum. Due to the random aspect of crossover and mutation, this did not always happen in a reasonable amount of time, e.g., within 30 min. Therefore, no guarantees can be given that this happens under all circumstances. One solution to this problem is to run the GA several times with different seedings of the random number generator. All GA runs are repeated until the same solution was obtained three times.

Most of the time, for both models a very acceptable solution, very close to the global optimum, could be obtained after approximately 100 generations. Even for the same model, it occurred that some problems took much more time to solve than others. The GA performance curve shown in Fig. 13 is for one of the easiest problems.

Due to slow finishing, each GA run performed 2000 generations to ensure that the fitness did not improve and that the global optimum was obtained. As an indication of the time needed, for the pharmacokinetic model: 2000 generations took 16 min on the hardware described above. The computation of the fitness function is the most time-consuming process and thus the dominant factor.

It takes approximately 1 h of computing time for a similar pharmacokinetic problem due to the replicated runs. 'Similar' means that the optimal kinetic parameters are expected in a small, predefined range.

On the other hand, the handling of a new and unknown data set which involves an iterative search space contraction easily takes a couple of hours. The computation of 2000 generations for the calibration model took 10 min which is due to the simpler fitness function.

Depending on the scientist's experience, the configuration of a GA for an entirely *new* model may cost several days or more.

8. Conclusions

Genetic algorithms can be applied to extend the least median of squares regression method to nonlinear models. The combination of GAs with LMS is useful for (i) robust regression and (ii) the detection of outliers in pharmacokinetic and curved calibration data.

Acknowledgements

The authors wish to thank Peter Rousseeuw for his helpful advice and I. Vercruysse for supplying the pharmacokinetic data set.

References

[1] H. Theil, A rank-invariant method of linear and polynomial regression analysis (Parts 1–3), Nederlandse Akademie voor Wetenschappen, Proceedings Series A, 1950, pp. 53, 386, 521, 1397.

[2] A.F. Siegel, Robust regression using repeated medians, Biometrika, 69 (1982) 242–244.

[3] P.J. Rousseeuw, Least median of squares regression, J. Am. Stat. Assoc., 79 (1984) 871–880.

[4] F.R. Hampel, A general qualitative definition of robustness, Ann. Math. Stat., 42 (1971) 1887–1896.

[5] P. Koscielniak, Non-linear robust regression procedure for calibration in flame atomic absorption spectrometry, Anal. Chim. Acta, 278 (1993) 177–187.

[6] J.H. Holland, Genetic algorithms, Sci. Am., July (1992) 44–50.

[7] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading, MA, 1989.

[8] C.B. Lucasius, Understanding and Using Genetic Algorithms. Part 2. Representation, Configuration and Hybridisation, Ph.D. Thesis, University of Nijmegen, Nijmegen, 1993.

[9] D. Beasley, D.R. Bull and R.R. Martin, An overview of genetic algorithms: Part 1, fundamentals, Univ. Comp., 15 (1993) 58–69.

[10] P.J. Rousseeuw and A.M. Leroy, Robust Regression and Outlier Detection, Wiley, New York, 1987.

[11] G.E.P. Box and M.E. Muller, A note on the generation of random normal deviates, Ann. Math. Stat., 29 (1958) 610–611.

[12] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, Numerical recipes in C, The Art of Scientific Computing, Cambridge University Press, Cambridge, 1988, p. 217.

[13] C.L. Karr, D.A. Stanley and B.J. Scheiner, Genetic algorithm applied to least squares curve fitting, US Department of the Interior, Bureau of Mines, Report of Investigations 9339, 1991.

[14] D.B. Hibbert, A hybrid genetic algorithm for the estimation of kinetic parameters, Chemom. Intell. Lab. Syst., 19 (1993) 319–329.

[15] R.M.L. Marques, P.J. Schoenmakers, C.B. Lucasius and L. Buydens, Modelling chromatographic behaviour as a function of pH and solvent composition in RPLC, Chromatographia, 36 (1993) 83–95.

[16] A.P. De Weijer, C.B. Lucasius, L. Buydens, G. Kateman, H.M. Heuvel and H. Mannee, Curve fitting using natural computation, Anal. Chem., 64 (1994) 23–31.

[17] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Springer, Berlin, 1992.

[18] D.M. Hawkins, J.S. Simonoff and A.J. Stromberg, Distributing a computationally intensive estimator: the case of the exact LMS regression, Comp. Stat., 9 (1994) in press.

[19] P. Lewi, personal communication, 1994.

[20] P. Vankeerberghen, Run suitability checking of calibration lines, in preparation.