

# Determining the form of ordinary differential equations using model inversion

Bill Whiten<sup>1</sup>

(Received 11 December 2013; revised 10 July 2014)

## Abstract

The model inversion approximation extracts parameters from within a nonlinear function so that they are exposed in a linear position convenient for further analysis. Experimental data can then be used to examine how the parameters vary with operating conditions. In particular, linear regression provides the selection and evaluation of the nonzero elements in linear relations between the parameters and the operating conditions. Where data is generated from an ordinary differential equation, this model inversion is used to investigate certain properties of the ODE equations, such as the reliance of ODE terms on external conditions, and the form of nonlinear relations of the state variables.

---

<http://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/7767> gives this article, © Austral. Mathematical Soc. 2014. Published August 2, 2014, as part of the Proceedings of the 11th Biennial Engineering Mathematics and Applications Conference. ISSN 1446-8735. (Print two pages per sheet of paper.) Copies of this article must not be made otherwise available on the internet; instead link directly to this URL for this article.

# Contents

<b>1</b>	<b>Introduction</b>	<b>C330</b>
<b>2</b>	<b>Model inversion</b>	<b>C332</b>
<b>3</b>	<b>Regression</b>	<b>C332</b>
3.1	Linear regularisation . . . . .	C333
3.2	Term selection . . . . .	C334
<b>4</b>	<b>Example: ODE rates and initial conditions</b>	<b>C335</b>
4.1	Data generation . . . . .	C336
4.2	Equation fitted . . . . .	C337
<b>5</b>	<b>Example: A nonlinear ODE</b>	<b>C338</b>
5.1	Function fitted . . . . .	C339
<b>6</b>	<b>Conclusions</b>	<b>C343</b>
<b>A</b>	<b>Appendix</b>	<b>C346</b>

## 1 Introduction

Ordinary differential equations (ODEs) are often used to describe physical systems where the coefficients in the model change with external conditions that are held constant for each test case. For instance, the growth of a plant can be described by an ordinary differential equation, and the rate coefficients in this equation vary with external conditions such as temperature, fertiliser, and atmosphere content. The model inversion approximation (Whiten, 1994, 2013) provides a method that converts these coefficient relations into a linear location so that efficient linear regression techniques can be used to analyse the form of these relations.

Grey box models, which have part of the model form specified and require data to complete the model, are usually handled using a combination of nonlinear least squares (Press et al, 2007, Sec. 15.5) with some method of evaluating alternatives such as enumeration, a genetic algorithm, or synthetic annealing (Bohlin, 2006; Ter Braak, 2006). Kojovic and Whiten (Kojovic, 1989; Kojovic & Whiten, 1994) automated techniques that fitted the parameters using individual data sets and then looked for parameter relations between the different data sets.

The model inversion technique (Whiten, 1994, 2013) provides an alternative method of efficiently developing grey box models. One potential application of this technique is using experimental data to determine unknown parts of the ordinary differential equations describing the data.

Consider an ODE of the form

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{g}[\mathbf{x}(t), t, \mathbf{p}(\mathbf{c})], \quad (1)$$

where the sizes of the vector functions  $\mathbf{x}$ ,  $\mathbf{g}$ ,  $\mathbf{p}$ , and the vector  $\mathbf{c}$  are determined by the problem being considered. The function  $\mathbf{g}$  is assumed known, and function  $\mathbf{p}(\mathbf{c})$  is unknown and assumed to be of the form  $\mathbf{A}\mathbf{c}$  with  $\mathbf{A}$  unknown, and  $\mathbf{c}$  is known operating conditions for the ODE. Data values of  $\mathbf{x}(t)$  for given values of  $\mathbf{c}$  recorded from a system modelled by equation (1) are used to determine the value of  $\mathbf{A}$ . Section 2 describes the method of converting the nonlinear problem into an approximate linear problem in the elements of  $\mathbf{A}$ , and Section 3 considers the regression methods needed to calculate and select the nonzero elements of  $\mathbf{A}$ .

Section 4 uses data consisting of values of  $\mathbf{x}(t)$  for a range of  $t$  values generated using six different values of the vector  $\mathbf{c}$ , to demonstrate determining the location and values of the nonzero elements of  $\mathbf{A}$ . This example also shows how the ODE initial conditions are calculated using parameters from  $\mathbf{p}(\mathbf{c})$ . In Section 5 values in  $\mathbf{A}$  are used to select from different proposed nonlinear terms of  $\mathbf{x}(t)$ . The data used to determine  $\mathbf{A}$  as in Section 4 is a time sequence of the two dimensional function  $\mathbf{x}(t)$ .

## 2 Model inversion

The model inversion approximation (Whiten, 1994, 2013) uses an approximate value of unknown parameters functions within a function to derive a formula that moves the parameter functions  $\mathbf{p}(\mathbf{c})$  from inside the main function  $\mathbf{f}[\mathbf{p}(\mathbf{c})]$  to a linear position:

$$\begin{aligned} \mathbf{f}(\mathbf{p}(\mathbf{c})) &\approx \left[ \mathbf{f}(\mathbf{p}_0) - \frac{\partial \mathbf{f}(\mathbf{p}_0)}{\partial \mathbf{p}_0^T} \mathbf{p}_0 \right] + \frac{\partial \mathbf{f}(\mathbf{p}_0)}{\partial \mathbf{p}_0^T} \mathbf{p}(\mathbf{c}) \\ &= \left[ \mathbf{f}(\mathbf{p}_0) - \frac{\partial \mathbf{f}(\mathbf{p}_0)}{\partial \mathbf{p}_0^T} \mathbf{p}_0 \right] + \frac{\partial \mathbf{f}(\mathbf{p}_0)}{\partial \mathbf{p}_0^T} \mathbf{A} \mathbf{c}. \end{aligned} \quad (2)$$

This is a general relation so the lengths of the vector functions  $\mathbf{f}$  and  $\mathbf{p}$  and the vector  $\mathbf{c}$  are determined by the problem being considered. The dimensions of the matrix  $\mathbf{A}$  must be compatible with vectors  $\mathbf{p}$  and  $\mathbf{c}$ . The approximation (2) is easily derived by substituting  $\mathbf{p}(\mathbf{c}) = \mathbf{p}_0 + \delta \mathbf{p}$ , expanding as a series, eliminating  $\delta \mathbf{p}$ , replacing  $\mathbf{p}(\mathbf{c})$  by  $\mathbf{A} \mathbf{c}$ , and separating the known terms from the terms containing the unknown matrix  $\mathbf{A}$ . This creates an offset from  $\mathbf{f}(\mathbf{p}_0)$  so that the nonzero values in  $\mathbf{A}$  can be selected and determined by linear regression methods. Transformations of elements of  $\mathbf{c}$  are introduced into  $\mathbf{A} \mathbf{c}$  to provide for nonlinear relations within  $\mathbf{p}(\mathbf{c})$ . The derivative  $\partial \mathbf{f}(\mathbf{p}_0) / \partial \mathbf{p}_0^T$  is the Jacobian of  $\mathbf{f}(\mathbf{p})$  evaluated at  $\mathbf{p}_0$ . The approximation (2) is applied repeatedly, first calculating  $\mathbf{A}$  then using  $\mathbf{p} = \mathbf{A} \mathbf{c}$ , to give an improved value of  $\mathbf{p}_0$ .

## 3 Regression

The advantage of approximation (2) is, given the estimate  $\mathbf{p}_0$  and values of  $\mathbf{c}$ , it provides a linear expression for the elements of  $\mathbf{A}$ . Thus, given data sets of operating conditions  $\mathbf{c}_i^*$  and resulting measured function values  $\mathbf{f}_i^*$ , the error between model predictions and the data  $\mathbf{f}_i^*$  is a linear function of the

elements in  $\mathbf{A}$ :

$$\mathbf{f}_i(\mathbf{A}\mathbf{c}_i^*) - \mathbf{f}_i^* \approx \mathbf{b}_i - \mathbf{B}_i\mathbf{a}, \quad (3)$$

where, comparing with approximation (2),

$$\mathbf{b}_i = \mathbf{f}_i(\mathbf{p}_0) - \frac{\partial \mathbf{f}_i(\mathbf{p}_0)}{\partial \mathbf{p}_0^\top} \mathbf{p}_0, \quad \mathbf{B}_i = \text{kron} \left( \mathbf{c}_i^{*\top}, \frac{\partial \mathbf{f}_i(\mathbf{p}_0)}{\partial \mathbf{p}_0^\top} \right), \quad \mathbf{a} = \text{vec}(\mathbf{A}), \quad (4)$$

where  $\text{kron}(\cdot, \cdot)$  is the Kronecker matrix product (Johnson & Horn, 1991, chap. 4), and  $\text{vec}(\cdot)$  puts columns of  $\mathbf{A}$  into a single column vector. Whiten (1994, 2013) gives expanded versions of this conversion to a linear form. After an initial iteration the values of  $\mathbf{p}_0$  will depend on the data set  $\mathbf{i}$ , estimated from the product of values of  $\mathbf{A}$  from a previous iteration and  $\mathbf{c}_i^*$ . Note that it is not necessary that the function  $\mathbf{f}_i(\mathbf{A}\mathbf{c}_i^*)$  be the same for each data set, hence the subscript on  $\mathbf{f}$ .

Efficient linear regression techniques are available to locate the zero values in  $\mathbf{a}$  and evaluate the remaining values. As we may be investigating a large number of possible elements in  $\mathbf{a}$ , this regression may be singular or near singular and thus a regularised minimising of the linear approximation is required.

### 3.1 Linear regularisation

Regularisation of the minimisation of  $\|\mathbf{b}_i - \mathbf{B}_i\mathbf{a}\|_2$  is done by biasing the solution to be close to a vector  $\mathbf{a}_0$  by minimising the sum of  $\|\lambda(\mathbf{a} - \mathbf{a}_0)\|_2$  and  $\|\mathbf{b}_i - \mathbf{B}_i\mathbf{a}\|_2$ . After each iteration the value of  $\mathbf{a}_0$  is updated to the just calculated value of  $\mathbf{a}$ , and, as shown in Appendix A, descent directions are available, so  $\mathbf{a}_0$  moves progressively closer to a minimum of  $\|\mathbf{b}_i - \mathbf{B}_i\mathbf{a}\|_2$ . This gives the following terms to be minimised as a sum of squares

$$\begin{bmatrix} \mathbf{B} \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{a} - \begin{bmatrix} \mathbf{b} \\ \lambda \mathbf{a}_0 \end{bmatrix}, \quad (5)$$

where  $\mathbf{B} = [\mathbf{B}_i]$ ,  $\mathbf{b} = [\mathbf{b}_i]$  and  $\lambda$  is chosen to approximately minimise  $\|[\mathbf{f}_i(\mathbf{A}\mathbf{c}_i^*) - \mathbf{f}_i^*]\|_2$ . Similar to the Levenberg–Marquardt algorithm (Press et al, 2007, p. 801), for the first iteration  $\lambda$  is set to 0.001 and then  $\lambda$  is progressively increased by factors of ten until the sum of squares  $\|[\mathbf{f}_i(\mathbf{A}\mathbf{c}_i^*) - \mathbf{f}_i^*]\|_2$  is reduced and then starts to increase. At the start of each following iteration with the updated value for  $\mathbf{a}_0$ ,  $\lambda$  is decreased by a factor of one hundred.

Differentiating the sum of squares of the terms (5) with respect to  $\mathbf{a}$  and setting to zero to get the condition for the minimum gives

$$(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})\mathbf{a} = \mathbf{B}^T\mathbf{b} + \lambda^2\mathbf{a}_0. \quad (6)$$

Then, using the singular value decomposition  $\mathbf{B} = \mathbf{U}\mathbf{D}\mathbf{V}^T$  where  $\mathbf{U}^T\mathbf{U} = \mathbf{I}$ ,  $\mathbf{D}$  is diagonal and  $\mathbf{V}^T\mathbf{V} = \mathbf{V}\mathbf{V}^T = \mathbf{I}$ , gives

$$\mathbf{V}(\mathbf{D}^2 + \lambda^2\mathbf{I})\mathbf{V}^T\mathbf{a} = \mathbf{V}(\mathbf{D}\mathbf{U}^T\mathbf{b} + \lambda^2\mathbf{V}^T\mathbf{a}_0). \quad (7)$$

Solving for  $\mathbf{a}$  gives

$$\mathbf{a} = \mathbf{V}(\mathbf{D}^2 + \lambda^2\mathbf{I})^{-1}(\mathbf{D}\mathbf{U}^T\mathbf{b} + \lambda^2\mathbf{V}^T\mathbf{a}_0), \quad (8)$$

and as  $\mathbf{D}$  is diagonal, values of  $\mathbf{a}$  are easily obtained for different values of  $\lambda$ , and these are all descent directions from  $\mathbf{a}_0$ . Also, even in the nonlinear case a sufficiently large value of  $\lambda$  will reduce the value of the sum of squares unless it is already at the minimum (see Appendix A). Thus, unless we are at the solution, the sum of squared errors can be reduced and usually convergence is not dependent on the initial value of  $\mathbf{a}_0$ . After each iteration  $\mathbf{a}_0$  is updated to the calculated value of  $\mathbf{a}$  and new values of  $\mathbf{B}$  and  $\mathbf{b}$  are calculated, so that each step is towards a minimum of  $\|[\mathbf{f}_i(\mathbf{A}\mathbf{c}_i^*) - \mathbf{f}_i^*]\|_2$ .

## 3.2 Term selection

Given  $\mathbf{a} + \delta\mathbf{a} = \mathbf{V}\mathbf{D}^{-1}\mathbf{U}^T(\mathbf{b} + \delta\mathbf{b})$  where  $\delta\mathbf{a}$  is the random variation in  $\mathbf{a}$  and  $\delta\mathbf{b}$  the random variation in  $\mathbf{b}$ , from equation (8), after setting  $\lambda$  to zero as at

the solution after iterating  $\mathbf{a}_0 = \mathbf{a}$ , we obtain

$$\delta \mathbf{a} = \mathbf{V} \mathbf{D}^{-1} \mathbf{U}^T \delta \mathbf{b} = \mathbf{M} \delta \mathbf{b}. \quad (9)$$

The covariance of  $\delta \mathbf{a}$  is

$$\text{covar}(\delta \mathbf{a}) = \text{var}_m(\delta \mathbf{a} \delta \mathbf{a}^T) = \mathbf{M} \text{var}_m(\delta \mathbf{b} \delta \mathbf{b}^T) \mathbf{M}^T = \mathbf{M} \mathbf{M}^T \text{var}(\delta \mathbf{b}) \quad (10)$$

where  $\text{var}(\cdot)$  is the variance,  $\text{var}_m(\mathbf{X})$  is the matrix of the variances of the elements of matrix  $\mathbf{X}$ , and it is assumed that the elements of  $\delta \mathbf{b}$  have equal variances and are uncorrelated. Then the standard error of  $\mathbf{a}$  is

$$\text{se}(\mathbf{a}) = \sqrt{\text{diag}(\mathbf{M} \mathbf{M}')} \text{sd}(\delta \mathbf{b}) = \sqrt{\text{diag}(\mathbf{M} \mathbf{M}')} \text{sd}(\mathbf{b} - \mathbf{B} \mathbf{a}), \quad (11)$$

where  $\text{sd}(\cdot)$  is the standard deviation. Elements of  $\mathbf{a}$  less than a small multiple of their standard error are progressively be set to zero, thus removing insignificant terms from the matrix  $\mathbf{A}$ .

## 4 Example: ODE rates and initial conditions

This example demonstrates how unknown rate parameters and initial conditions relations within an ODE are determined from experimental data. For this example the function  $\mathbf{f}(\mathbf{p}) = \mathbf{f}(\mathbf{A} \mathbf{c})$  is a vector of  $x(t)$  and  $y(t)$  functions of  $\mathbf{p}$  at selected times, defined by the two dimensional set of ODEs given in Section 4.2, while the data sets  $\mathbf{f}_i^*$  are generated using the same selected times and six different values of  $\mathbf{c}_i$  from the differential equations given in Section 4.1.

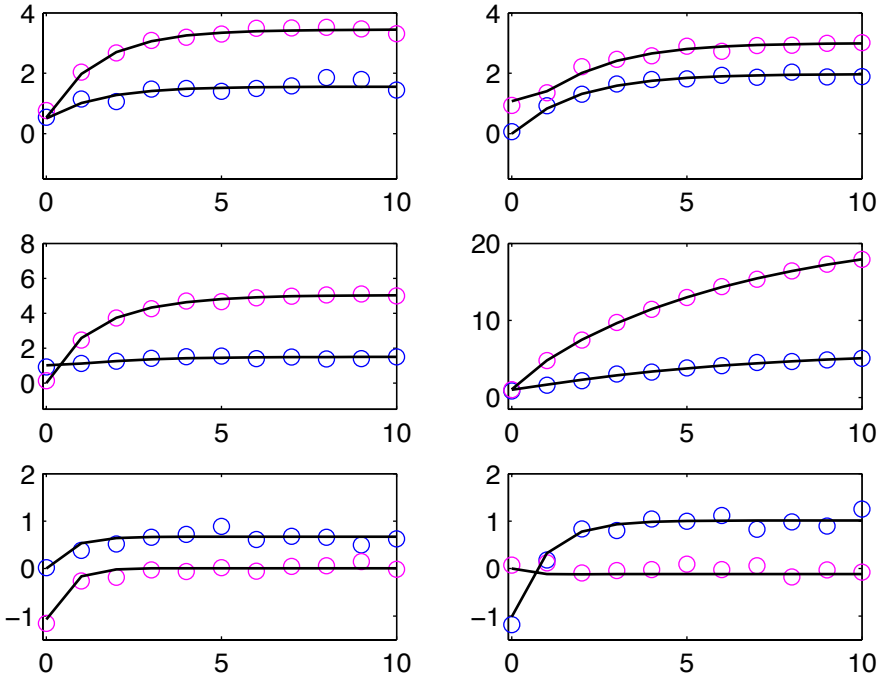


Figure 1: Simulated experimental data using equations (12) and (13) (circles), and fit to the data (lines).

### 4.1 Data generation

Sample data was generated for  $t = 0, 1, \dots, 10$  from

$$\frac{dx}{dt} = 1 + (-1 + 0.5c_2)x + 0.1c_3y, \tag{12}$$

$$\frac{dy}{dt} = (1 + c_2 + c_3) + (1 + c_2 + c_3)x + (-2 + c_3)y, \tag{13}$$



with  $x(0) = c_3$ ,  $y(0) = c_2$ , and six sets of the three element vector  $\mathbf{c}$ , where the first element is one, equal to the columns of the matrix

$$\mathbf{C} = \begin{bmatrix} 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 \\ 0.5 & 1.0 & 0.0 & 1.0 & -1.0 & 0.0 \\ 0.5 & 0.0 & 1.0 & 1.0 & 0.0 & -1.0 \end{bmatrix}. \quad (14)$$

Gaussian random values  $N(0, 0.1)$  were added to the generated values. The six sets of data are shown in Figure 1. Although these equations could be solved analytically, a numerical integrator, the fourth order Runge–Kutta RK4 (Henrici, 1964; Press et al, 2007), is used to solve them, as a numerical solution is more typical of potential applications.

## 4.2 Equation fitted

To fit the data  $x(t_i)$  and  $y(t_i)$  for  $t_i = 0, 1, \dots, 10$  generated according to Section 4.1, we use, with the given  $\mathbf{c}$  values,

$$\frac{dx}{dt} = p_1 + p_2x + p_3y, \quad x(0) = p_7, \quad \frac{dy}{dt} = p_4 + p_5x + p_6y, \quad y(0) = p_8, \quad (15)$$

where  $\mathbf{p} = \mathbf{A}\mathbf{c}$  and  $\mathbf{A}$  is an unknown eight by three matrix to be determined. The vector  $\mathbf{p} = \mathbf{A}\mathbf{c}$  takes six different values with  $\mathbf{c}$  coming from the the columns of  $\mathbf{C}$  in equation (14). Again the fourth order Runge–Kutta RK4 is used. Fixed steps are used so that reliable numerical derivatives are calculated for approximation (2). Initially  $\mathbf{A}$  is set to zero. From  $\mathbf{A} = \mathbf{0}$  it took 14 iterations for the relative change in the sum of squares of the residue to be less than  $10^{-10}$ , and another six iterations to select the nonzero terms. The standard error of the residuals is 0.101, which agrees closely with the random error with standard deviation 0.1 added during the data generation.

The original matrix  $\mathbf{A} = \mathbf{A}_{\text{Orig}}$  used in equations (12) and (13), and the calculated  $\mathbf{A} = \mathbf{A}_{\text{Calc}}$  found using (15) and the techniques in Sections 2 and 3,

are

$$A_{\text{Orig}} = \begin{bmatrix} 1.0 & 0 & 0 \\ -1.0 & 0.5 & 0 \\ 0 & 0 & 0.1 \\ 1.0 & 1.0 & 1.0 \\ 1.0 & 1.0 & 1.0 \\ -2.0 & 0 & 1.0 \\ 0 & 0 & 1.0 \\ 0 & 1.0 & 0 \end{bmatrix}, \quad A_{\text{Calc}} = \begin{bmatrix} 1.083 & 0 & 0 \\ -1.081 & 0.531 & 0 \\ 0 & 0 & 0.107 \\ 0.786 & 0.776 & 1.128 \\ 1.017 & 1.009 & 0.992 \\ -1.853 & 0 & 0.877 \\ 0 & 0 & 1.017 \\ 0 & 1.070 & 0 \end{bmatrix}. \quad (16)$$

The solid lines in Figure 1 are calculated using  $A_{\text{Calc}}$  and equations (15). A closer match to the original  $A$  values is obtained if more data is used or less random variation is introduced.

## 5 Example: A nonlinear ODE

This example demonstrates how unknown functions of state variables are determined. Data values for  $f_1^*$  (this example has only one data set) are generated for  $t = 0, 0.1, \dots, 4.9, 5$  from

$$\frac{dx}{dt} = 5 - g(x) + y, \quad x(0) = 0, \quad \frac{dy}{dt} = 4 + x - g(y), \quad y(0) = 0, \quad (17)$$

where  $g(x)$  is a natural cubic spline, which is cubic polynomial sections and straight line ends joined with continuous first and second derivatives (de Boor, 2001), passing through the points

$x$	0.0	1.0	2.0	3.0	4.0
$g(x)$	0.0	2.6875	5.0	6.6875	8.0
$g'(x)$	2.75	2.5625	2.0	1.4375	1.25

and is shown in Figure 2. Small Gaussian random values  $N(0, 10^{-5})$  were added to the data points generated. Figure 3 shows the points generated and used for fitting in the next section.

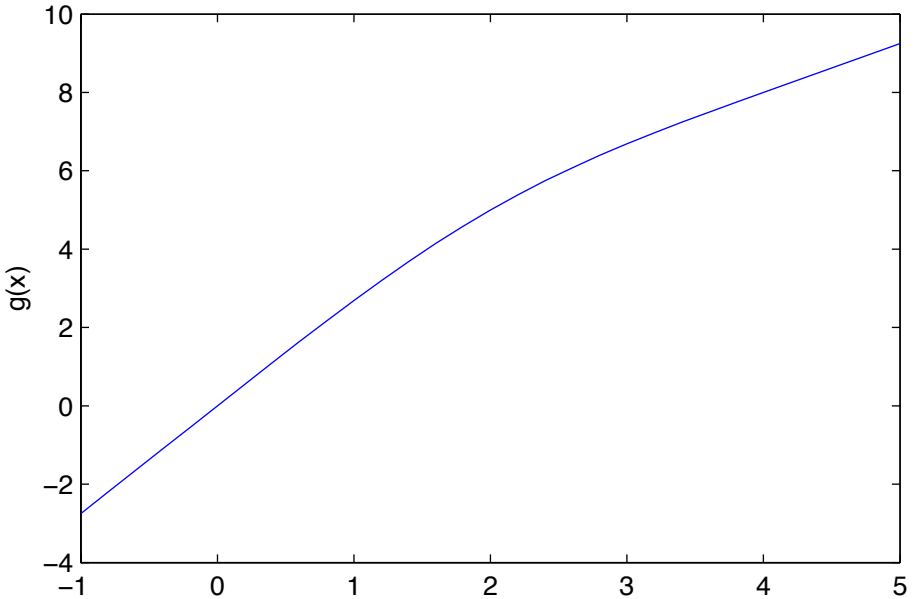


Figure 2: Relation used for  $g(x)$  in equation (17).

## 5.1 Function fitted

Similar to Section 4,  $\mathbf{f}(\mathbf{A}\mathbf{c})$  is a vector of function values consisting of  $\mathbf{x}(t)$  and  $\mathbf{y}(t)$  from equation (18) with  $t = 0, 0.1, \dots, 4.9, 5$ . Four unknown functions,  $S_{xx}(\mathbf{x})$ ,  $S_{xy}(\mathbf{y})$ ,  $S_{yx}(\mathbf{x})$  and  $S_{yy}(\mathbf{y})$ , defined using cubic spline functions (piecewise cubics with smooth joins) (de Boor, 2001) are used to define unknown coefficients in  $\mathbf{p}$  (in this case  $\mathbf{c}$  is one and  $\mathbf{p} = \mathbf{A}$  where  $\mathbf{A}$  is  $18 \times 1$ ):

$$\frac{dx}{dt} = p_1 - S_{xx}(x) + S_{xy}(y), \quad \frac{dy}{dt} = p_{10} + S_{yx}(x) - S_{yy}(y), \quad x(0) = y(0) = 0. \quad (18)$$

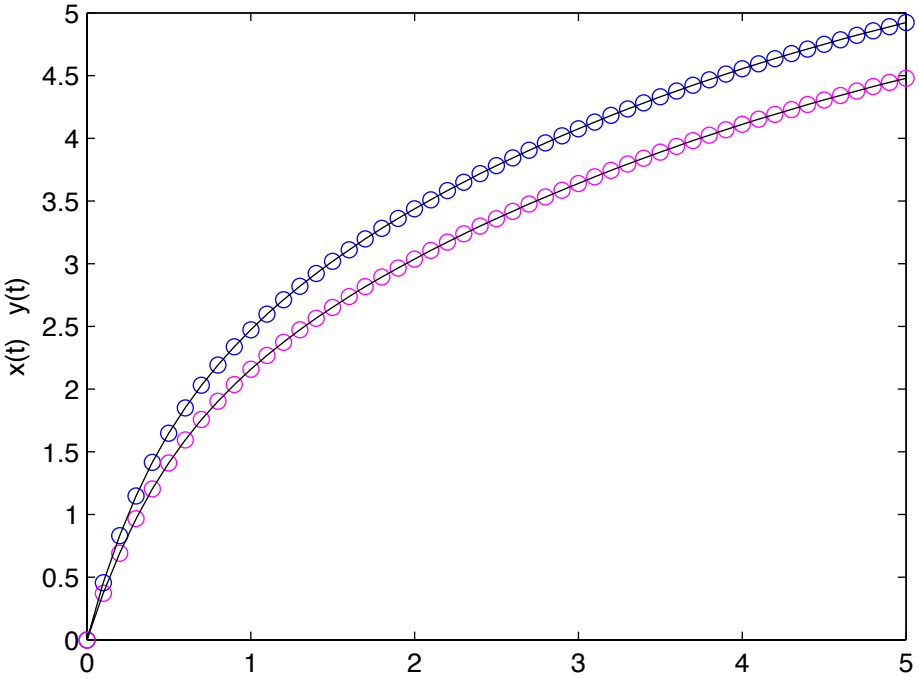


Figure 3: Data values generated by equations (17) (circles) and fitted function (lines).

The four spline functions are

$$S_{xx}(x) = p_2x + p_3s_1(x) + p_4s_2(x) + p_5s_3(x), \tag{19}$$

$$S_{xy}(y) = p_6y + p_7s_1(y) + p_8s_2(y) + p_9s_3(y), \tag{20}$$

$$S_{yx}(x) = p_{11}x + p_{12}s_1(x) + p_{13}s_2(x) + p_{14}s_3(x), \tag{21}$$

$$S_{yy}(y) = p_{15}y + p_{16}s_1(y) + p_{17}s_2(y) + p_{18}s_3(y), \tag{22}$$

which pass through  $(0,0)$  and have a linear component and three local B-spline components,  $s_1$ ,  $s_2$  and  $s_3$ , shown in Figure 4. Table 1 gives the points defining these three B-splines. The B-splines are a special case of natural cubic spline functions.

Table 1: Join points and derivatives for the B-spline components.

$x$	0.0	1.0	2.0	3.0	4.0
$s_1(x)$	0.0	1.0	0.25	0.0	0.0
$s_1'(x)$	1.5	0.0	-0.75	0.0	0.0
$s_2(x)$	0.0	0.25	1.0	0.25	0.0
$s_2'(x)$	0.0	0.75	0.0	-0.75	0.0
$s_3(x)$	0.0	0.0	0.25	1.0	0.0
$s_3'(x)$	0.0	0.0	0.75	0.0	-1.5

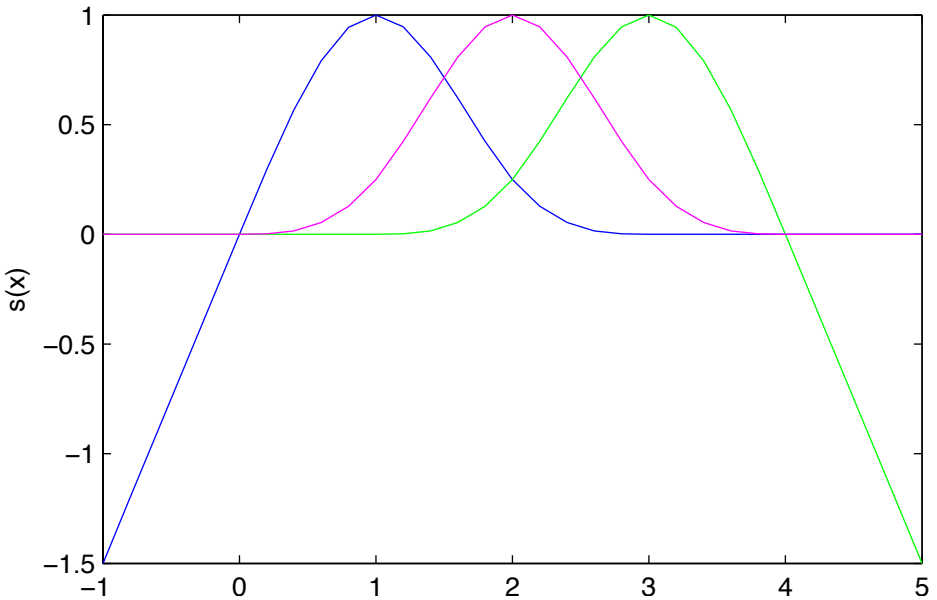


Figure 4: The three B-splines,  $s_1(x)$ ,  $s_2(x)$ ,  $s_3(x)$ , used as the basis for the nonlinear terms in equations (18).

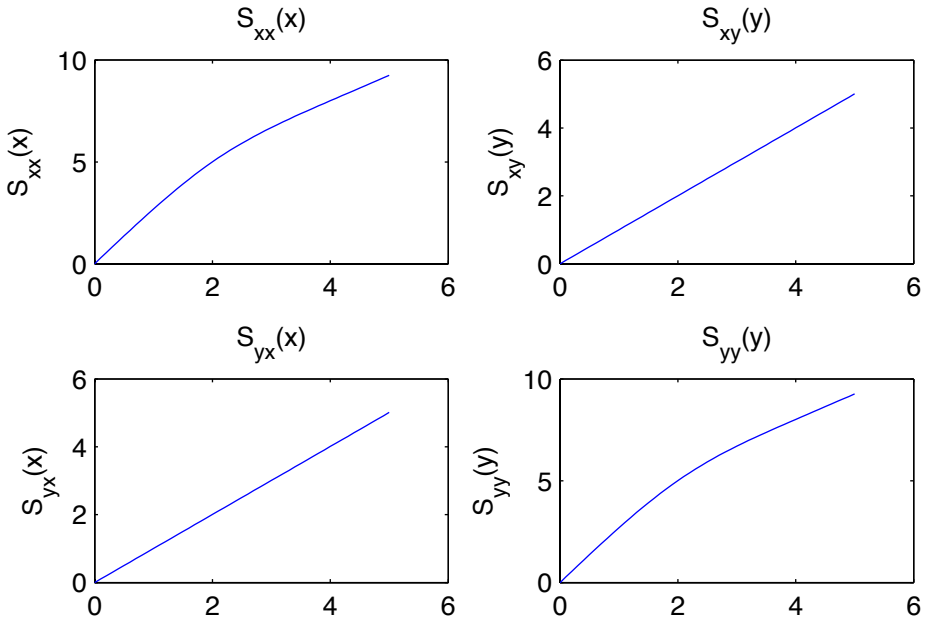


Figure 5: Fitted functions used in equation (18).

Applying the methods in Sections 2 and 3 gives the fit shown in Figure 3, using the four spline functions shown in Figure 5. The method correctly identifies the functions in equation (17) as the two nonlinear functions and the two linear components. Starting with  $\mathbf{A} = \mathbf{0}$  this case took 13 iterations to converge and another three iterations to select the nonzero terms. The standard deviation of the residuals was  $0.989 \times 10^{-5}$ , close to the  $1 \times 10^{-5}$  standard deviation of the random variation added in the generation of the data. Table 2 gives the fitted  $\mathbf{A} = \mathbf{p}$  values. These are all very close to the expected values, and round to the expected values when restricted to two decimal places, with the largest difference being less than  $3 \times 10^{-3}$ .

Table 2: Parameters  $\mathbf{p}$  which are coefficients for fitted terms in equations (18) and (19)–(22).

Variable	Constant	Slope	$s_1$	$s_2$	$s_3$	
$dx/dt$	5.0000	$S_{xx}$	2.0008	0.5000	0.7499	0.5000
		$S_{xy}$	1.0009	0	0	0
$dy/dt$	3.9999	$S_{yx}$	1.0023	0	0	0
		$S_{yy}$	2.0026	0.5000	0.7501	0.5001

## 6 Conclusions

Models, including those based on differential equations, can be inverted so that parameter relations inside the model are exposed in a linear position for efficient analysis. The parameters  $\mathbf{p}$  being investigated must have fixed values during integration, but can be related to external operating conditions  $\mathbf{c}_i$  that vary in different data sets. Multiple data sets generated with different operating condition values are combined into an analysis that minimises errors in the original measurements. Nonlinear relations can be included, as in linear regression, by including transforms of the operating conditions.

By dividing functions of state variables into a sum of basis functions, unknown functions of the state variables are determined from experimental data. Spline functions provide a convenient basis for this analysis. Given sufficient data the coefficients of the basis functions can be linear functions of operating conditions, as in the first example.

Although the minimisation is usually quite robust with zero often being an adequate start value, insufficient data or data with a too large a random component can result in multiple solutions, particularly from the selection of the zero terms in  $\mathbf{A}$ , and it may be necessary to try multiple initial approximations.

For the application to a new situation, it is not possible to make general statements recommending the amount, range or accuracy of data that would

be needed to obtain a specified model accuracy, other than the obvious, more is better. As in other cases of experimental design, a simulation of the data generation, similar to that done in Sections 4 and 5, and analysis is recommended. By testing a range of scenarios such as varying the amount of data, range of operating conditions, size of errors in the data, and model forms, it is then possible to estimate what experimental data would be needed to provide the required accuracy.

The criteria for selection or rejection of regression terms needs to be set carefully. Correlated variables can be selected if present individually, but if both are included, then they can both be rejected. Also, if there are a large number of alternatives, then a larger rejection criteria may be needed to avoid including irrelevant variables.

**Acknowledgements** The initial development of this work was done at the Julius Kruttschnitt Mineral Research Centre, The University of Queensland. Ray Phillips, Ricardo Pascual, and Jinhong Xiao worked on developing programs to implement model inversion. Jinhong Xiao (Xiao, 1998) also showed how the techniques are applied to a steady state hydrocyclone model, and Tim Tran (Tran, 2000) undertook related work on term selection in linear regression. The two referees are thanked for suggesting multiple places where this article could be expanded to make the meaning clearer.

## References

- Bohlin, T. P., *Practical grey-box process identification, Theory and applications (Advances in industrial control series)*, Springer, 2006. ISBN: 978-1-846-28402-1 <http://www.springer.com/engineering/robotics/book/978-1-84628-402-1> C331
- de Boor, C., *A Practical Guide to Splines*, Springer-Verlag, 2001. ISBN:



- 978-0-387-95366-3 <http://www.springer.com/mathematics/analysis/book/978-0-387-95366-3> C338, C339
- Henrici, P., *Elements of numerical analysis*, John Wiley, 1964. ISBN: 978-0-471-37241-7 C337
- Johnson, C. R. and Horn, R. A., *Topics in matrix analysis*, Cambridge University Press, 1991. ISBN: 978-0-521-30587-7 <http://www.cambridge.org/au/academic/subjects/mathematics/algebra/topics-matrix-analysis?format=HB> C333
- Kojovic, T., *The development and application of Model—an automated model builder for mineral processing*, PhD thesis, The University of Queensland, 1989. C331
- Kojovic, T., and Whiten W. J., *Evaluation of the quality of simulation models*, Innovations in mineral processing, (Laurentian University, Sudbury) p. 437–446, 1994. ISBN: 088667025X C331
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., *Numerical recipes*, Cambridge University Press, 2007. ISBN: 978-0-521-88068-8 <http://www.nr.com/>; <http://www.cambridge.org/us/academic/subjects/mathematics/numerical-recipes/numerical-recipes-art-scientific-computing-3rd-edition> C331, C334, C337
- Ter Braak, C. J. F., *A Markov Chain Monte Carlo version of the genetic algorithm Differential Evolution: easy Bayesian computing for real parameter spaces*, Stat. Comput. 16:239–249, 2006. doi:10.1007/s11222-006-8769-1 C331
- Tran, V., *Improved calibration techniques for nuclear analysers*, PhD thesis, The University of Queensland, 1998. C344

Whiten, W.J., Determination of parameter relations within non-linear models, *SIGNUM Newsletter*, 29(3–4):2–5, 1994. doi:[10.1145/192527.192535](https://doi.org/10.1145/192527.192535) C330, C331, C332, C333

Whiten, B., Model completion and validation using inversion of grey box models, *ANZIAM J. (CTAC)*, 54:C187–C199, 2013. <http://journal.austms.org.au/ojs/index.php/ANZIAMJ/article/view/6125> C330, C331, C332, C333

Xiao, J., *Extensions of model building techniques and their applications in mineral processing*, PhD thesis, The University of Queensland, 1998. C344

## A Appendix

To show equation (8) generates descent directions for the sum of squares  $\|\mathbf{B}\mathbf{a} - \mathbf{b}\|_2$  around the point  $\mathbf{a}_0$  from the difference  $\mathbf{a} - \mathbf{a}_0$ , use equation (6),

$$\begin{aligned} \mathbf{a} - \mathbf{a}_0 &= (\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}(\mathbf{B}^T\mathbf{b} + \lambda^2\mathbf{a}_0) - \mathbf{a}_0 \\ &= (\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T(\mathbf{b} - \mathbf{B}\mathbf{a}_0) \\ &= (\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T\mathbf{b}^*, \end{aligned} \quad (23)$$

where  $\mathbf{b}^* = \mathbf{b} - \mathbf{B}\mathbf{a}_0$  and thus  $\mathbf{b}^{*T}\mathbf{b}^*$  is the sum of squared residuals at  $\mathbf{a}_0$ . Substituting  $\mathbf{a} = \mathbf{a}_0 + \mathbf{x}$  into the sum of squares  $\|\mathbf{B}\mathbf{a} - \mathbf{b}\|_2$  and then substituting  $\mathbf{x} = \alpha(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T\mathbf{b}^*$  gives

$$\begin{aligned} \|\mathbf{B}\mathbf{a} - \mathbf{b}\|_2 &= (\mathbf{B}\mathbf{x} - \mathbf{b}^*)^T(\mathbf{B}\mathbf{x} - \mathbf{b}^*) \\ &= (\alpha\mathbf{B}(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T\mathbf{b}^* - \mathbf{b}^*)^T(\alpha\mathbf{B}(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T\mathbf{b}^* - \mathbf{b}^*) \\ &= \alpha^2\mathbf{b}^{*T}\mathbf{B}(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T\mathbf{B}(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T\mathbf{b}^* \\ &\quad - 2\alpha\mathbf{b}^{*T}\mathbf{B}(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T\mathbf{b}^* + \mathbf{b}^{*T}\mathbf{b}^*. \end{aligned} \quad (24)$$

Now, as  $(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})$  is positive definite,

$$\mathbf{b}^{*T}\mathbf{B}(\mathbf{B}^T\mathbf{B} + \lambda^2\mathbf{I})^{-1}\mathbf{B}^T\mathbf{b}^* \quad (25)$$

is positive provided

$$\mathbf{B}^T \mathbf{b}^* = \mathbf{B}^T (\mathbf{b} - \mathbf{B} \mathbf{a}_0) \quad (26)$$

is not zero. The above is only zero if  $\mathbf{a}_0$  gives the minimum of  $|\mathbf{B} \mathbf{a} - \mathbf{b}|_2$ . Provided  $\mathbf{a}_0$  is not the minimum, and  $\alpha$  is sufficiently small, the sum of squares (24) is reduced, thus equation (8) defines a descent direction from  $\mathbf{a}_0$ .

Similarly, provided the nonlinear terms  $[\mathbf{f}_i(\mathbf{A} \mathbf{c}_i^*) - \mathbf{f}_i^*]$  are smooth with most terms non-constant, locally linear and rounding errors are not significant, a sufficiently large value of  $\lambda$  will reduce the sum of squares  $||[\mathbf{f}_i(\mathbf{A} \mathbf{c}_i^*) - \mathbf{f}_i^*]|_2$ .

For the linear case, minimising  $|\mathbf{B} \mathbf{x} - \mathbf{b}|_2$  with respect to  $\mathbf{x}$  subject to the constraint  $|\mathbf{x} - \mathbf{a}^*|_2 = r^2$  using  $\lambda^2$  as the Lagrange multiplier, gives the solution  $\mathbf{x} = \mathbf{a}$  with  $\mathbf{a}$  defined by equation (8), showing this solution is the minimum on the constraint circle  $|\mathbf{x} - \mathbf{a}^*|_2 = |\mathbf{a} - \mathbf{a}^*|_2$ .

## Author address

1. **Bill Whiten**, 4 Magnet Cl, Riverhills, Brisbane, Queensland, 4072, Australia.

<mailto:billwhiten@tpg.com.au>