A Quick User's Guide for IDEAS v1.1: a Parameter Identification Toolbox with Symbolic Analysis of Uncertainty

Copyright(c) INRA - Univ Paris-Sud - CNRS

Reference number from APP:
IDDN.FR.001.110021.000.R.C.2010.000.20700.

Authors:
Rafael Muñoz Tamayo[1,2,3], rafael.munoztamayo@jouy.inra.fr, rafaun@yahoo.com
Béatrice Laroche[3], Beatrice.Laroche@lss.supelec.fr
Marion Leclerc[2], marion.leclerc@jouy.inra.fr
Éric Walter[3], Eric.Walter@lss.supelec.fr
(1) UR341 INRA Jouy-en-Josas, Unité de Mathématiques et Informatique Appliquées
(2) UR910 INRA Jouy-en-Josas, Unité d'Ecologie et Physiologie du Système Digestif
(3) UMR8506 Univ Paris Sud-CNRS-SUPÉLEC, Laboratoire des Signaux et Systèmes

Contents

Chapter 1

What is IDEAS?

IDEAS is the acronym for IDEntification and Analysis of Sensitivity. This software is a Matlab® toolbox for parameter identification of ordinary differential equation (ODE) models. The parameter estimation is performed in the maximum-likelihood (ML) sense. The current version v1.1 tackles the estimation problem for the case of synchronous observations.
All the functions generated are accessible and can be utilized in other user-defined routines, and modified if needed.
IDEAS is a free toolbox, developed at the teams Mathématiques et Informatique Appliquées and Unité d'Ecologie et Physiologie du Système Digestif from the Institut National de la Recherche Agronomique (INRA) at Jouy en Josas and the Laboratoire des Signaux et Systèmes (L2S), a public research facility common to CNRS, SUPELEC and Univ Paris-Sud from France.

IDEAS was presented in the 15th IFAC Symposium on System Identification, SYSID 2009. If you publish results using this toolbox, please cite the respective reference as appears in (2).

Chapter 2

Background

The theory of parameter identification and uncertainty analysis that are used in the software was published elsewhere (2). In this part is recalled to favor the understanding of the toolbox.

2.1 Parameter estimation

The parameter estimation is performed in the ML sense.
Consider the state-space model:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \boldsymbol{\theta}, t),\ \mathbf{x}(0) = \mathbf{x}_0(\boldsymbol{\theta}), \tag{2.1}$$

where $\mathbf{x}(t, \boldsymbol{\theta})$ is the state vector ($\mathbf{x} : \mathbb{R}^+ \times \mathbb{R}^{n_p} \to \mathbb{R}^{n_x}$), $\boldsymbol{\theta}$ is the parameter vector ($\boldsymbol{\theta} \in \mathbb{R}^{n_p}$), and $\mathbf{f}$ is a C^1 (continuous with continuous first-order partial derivatives) vector-valued function of the state and parameters ($\mathbf{f} : (\mathbb{R}^{n_x} \times \mathbb{R}^{n_p} \times \mathbb{R}^+) \to \mathbb{R}^{n_x}$).
In the special case of synchronous observations the model output, represented as the $\mathbb{R}^{n_y}$ vector $\mathbf{y}_\mathrm{m}$, satisfies

$$\mathbf{y}_\mathrm{m}(t, \boldsymbol{\theta}) = \mathbf{h}(\mathbf{x}, \boldsymbol{\theta}, t) \tag{2.2}$$

The vector of data collected at time t_i is modelled as:

$$\mathbf{y}(t_i) = \mathbf{y}_\mathrm{m}(t_i, \boldsymbol{\theta}^*) + \boldsymbol{\varepsilon}_i,\ i = 1, ..., n_\mathrm{t}, \tag{2.3}$$

with n_t the number of observation times, $\boldsymbol{\theta}^*$ the true value of the parameter vector. The measurement errors $\boldsymbol{\varepsilon}_i (i = 1, ..., n_\mathrm{t})$ are assumed to be independent, homoscedastic, zero mean and Gaussian, which means that $\boldsymbol{\varepsilon}_i \sim \mathbf{N}(\mathbf{0}, \boldsymbol{\Sigma})$. Under these hypotheses, the ML estimator is:

$$(\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\Sigma}}) = \arg \min_{\boldsymbol{\theta}, \boldsymbol{\Sigma}} L(\boldsymbol{\theta}, \boldsymbol{\Sigma}), \tag{2.4}$$

where

$$\begin{aligned} L(\boldsymbol{\theta}, \boldsymbol{\Sigma}) = \tfrac{n_{\mathrm{t}}}{2} \ln \det \boldsymbol{\Sigma} + \\ \tfrac{1}{2} \textstyle\sum_{i=1}^{n_{\mathrm{t}}} [\mathbf{y}(t_i) - \mathbf{y}_{\mathrm{m}}(t_i, \boldsymbol{\theta})]^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} [\mathbf{y}(t_i) - \mathbf{y}_{\mathrm{m}}(t_i, \boldsymbol{\theta})]. \end{aligned} \tag{2.5}$$

The equation 2.5 is optimized depending on the hypothesis on the covariance matrix $\boldsymbol{\Sigma}$ (see (1) and (3)). IDEAS offers to the user four alternatives:

Unweighted Least Squares. If $\boldsymbol{\Sigma}$ is assumed to be proportional to the identity matrix, the ML estimator for $\boldsymbol{\theta}$ is the unweighted least-squares estimator, which minimizes the cost function

$$J_1(\boldsymbol{\theta}) = \sum_{i=1}^{n_{\mathrm{t}}} [\mathbf{y}(t_i) - \mathbf{y}_{\mathrm{m}}(t_i, \boldsymbol{\theta})]^{\mathrm{T}} [\mathbf{y}(t_i) - \mathbf{y}_{\mathrm{m}}(t_i, \boldsymbol{\theta})], \tag{2.6}$$

and the ML estimate of the covariance matrix is

$$\widehat{\boldsymbol{\Sigma}} = \frac{J_2(\widehat{\boldsymbol{\theta}})}{n_{\mathrm{t}}} I. \tag{2.7}$$

Maximum likelihood with unknown $\boldsymbol{\Sigma}$ and diagonal. If $\boldsymbol{\Sigma}$ is unknown and diagonal, the ML estimator for $\boldsymbol{\theta}$ minimizes the cost function

$$J_2(\boldsymbol{\theta}) = \sum_{k=1}^{n_{\mathrm{y}}} \frac{n_{\mathrm{t}}}{2} \ln \left[\sum_{\mathrm{i}=1}^{n_{\mathrm{t}}} [y_k(t_i) - y_{\mathrm{m}_k}(t_i, \boldsymbol{\theta})]^2 \right], \tag{2.8}$$

and the ML estimate of the covariance matrix is

$$\widehat{\boldsymbol{\Sigma}} = \mathrm{diag}(\widehat{\sigma}_1^2, \cdots, \widehat{\sigma}_{n_{\mathrm{y}}}^2), \tag{2.9}$$

with

$$\widehat{\sigma}_k^2 = \frac{1}{n_{\mathrm{t}}} \sum_{i=1}^{n_{\mathrm{t}}} [y_k(t_i) - y_{\mathrm{m}_k}(t_i, \widehat{\boldsymbol{\theta}})]^2. \tag{2.10}$$

Maximum likelihood with unknown $\boldsymbol{\Sigma}$. If the covariance matrix is unknown, the ML estimator for $\boldsymbol{\theta}$ minimizes the cost function:

$$J_3(\boldsymbol{\theta}) = \ln(\det \sum_{i=1}^{n_{\mathrm{t}}} [\mathbf{y}(t_i) - \mathbf{y}_{\mathrm{m}}(t_i, \boldsymbol{\theta})][\mathbf{y}(t_i) - \mathbf{y}_{\mathrm{m}}(t_i, \boldsymbol{\theta})]^{\mathrm{T}}), \tag{2.11}$$

and the ML estimate of the covariance matrix is

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{n_\mathrm{t}} \sum_{i=1}^{n_\mathrm{t}} [\mathbf{y}(t_i) - \mathbf{y}_\mathrm{m}(t_i, \widehat{\boldsymbol{\theta}})][\mathbf{y}(t_i) - \mathbf{y}_\mathrm{m}(t_i, \widehat{\boldsymbol{\theta}})]^\mathrm{T}. \tag{2.12}$$

Maximum likelihood with known $\boldsymbol{\Sigma}$. If $\boldsymbol{\Sigma}$ is known (provided by the user), the estimator ML corresponds to the Gauss-Markov estimator, which minimizes the cost function

$$J_4(\boldsymbol{\theta}) = \sum_{i=1}^{n_\mathrm{t}} [\mathbf{y}(t_i) - \mathbf{y}_\mathrm{m}(t_i, \boldsymbol{\theta})]^\mathrm{T} \boldsymbol{\Sigma}^{-1} [\mathbf{y}(t_i) - \mathbf{y}_\mathrm{m}(t_i, \boldsymbol{\theta})]. \tag{2.13}$$

2.2 Confidence intervals of the estimates

The covariance matrix $\mathbf{P}$ of the ML estimator is approximated by the inverse of the FIM ($\mathbf{F}$) computed at $\widehat{\boldsymbol{\theta}}$. The estimate of $\mathbf{P}$ is taken as

$$\widehat{\mathbf{P}} = \mathbf{F}^{-1}(\widehat{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_0), \tag{2.14}$$

where $\boldsymbol{\Sigma}_0$ is a nominal value for the noise covariance. For the cases when $\boldsymbol{\Sigma}$ is unknown, it is approximated to its estimate $\boldsymbol{\Sigma}_0 = \widehat{\boldsymbol{\Sigma}}$ and the FIM is written as

$$\mathbf{F}(\widehat{\boldsymbol{\theta}}) = \sum_{i=1}^{n_\mathrm{t}} \left[\frac{\partial \mathbf{y}_\mathrm{m}}{\partial \boldsymbol{\theta}}\right]^\mathrm{T}_{(t_i, \widehat{\boldsymbol{\theta}})} \boldsymbol{\Sigma}_0^{-1} \left[\frac{\partial \mathbf{y}_\mathrm{m}}{\partial \boldsymbol{\theta}}\right]_{(t_i, \widehat{\boldsymbol{\theta}})}. \tag{2.15}$$

When $\boldsymbol{\Sigma}$ is unknown, a widely used approach is to approximate it by its ML estimate $\widehat{\boldsymbol{\Sigma}}$. The FIM is then computed taking $\boldsymbol{\Sigma}_0 = \widehat{\boldsymbol{\Sigma}}$ in (2.15). The square root η_j of the jth diagonal element of $\widehat{\mathbf{P}}$ is an estimate of the standard deviation of $\widehat{\theta}_j$, which is used to obtain an approximate 95% confidence interval for θ_j as: $[\widehat{\theta}_j \pm 2\eta_j]$.
To evaluate (2.15), it is required to compute the sensitivity matrix of the output. Let $\mathbf{s}_j = \frac{\partial \mathbf{x}}{\partial \theta_j}$ denote the sensitivity of the state to the parameter θ_j. The vector $\mathbf{s}_j$ is the solution of

$$\dot{\mathbf{s}}_j = \left[\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right]_{(\mathbf{x}, \boldsymbol{\theta}, t)} \mathbf{s}_j + \left[\frac{\partial \mathbf{f}}{\partial \theta_j}\right]_{(\mathbf{x}, \boldsymbol{\theta}, t)}, \mathbf{s}_j(0) = \mathbf{0} \tag{2.16}$$

The sensitivity of the output to the parameter θ_j ($\frac{\partial \mathbf{y}_\mathrm{m}}{\partial \theta_j}$) is evaluated as

$$\frac{\partial \mathbf{y}_\mathrm{m}}{\partial \theta_j} = \left[\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right]_{(\mathbf{x}, \boldsymbol{\theta}, t)} \mathbf{s}_j + \left[\frac{\partial \mathbf{h}}{\partial \theta_j}\right]_{(\mathbf{x}, \boldsymbol{\theta}, t)}, \frac{\partial \mathbf{y}}{\partial \theta_j}(0) = \mathbf{0}. \tag{2.17}$$

The FIM is therefore computed after the solution of the augmented system given by (2.1,2.16) at $\widehat{\boldsymbol{\theta}}$, and making the substititions on (2.17). The right-hand sides of (2.16-2.17) are evaluated using the Symbolic Toolbox of Matlab.

Chapter 3

Using Ideas

IDEAS consists of five files. The user has to create a folder and copy the source files. The toolbox runs in Matlab v7.0 or latter versions. It requires the optimization and symbolic toolbox to be executed.

IDEAS is operated through a graphical interface and dialog boxes in the Command

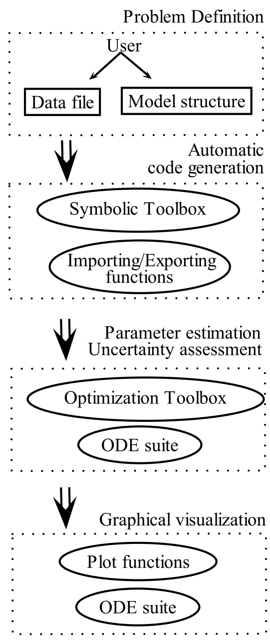


Figure 3.1: Steps followed by IDEAS

Window. It is composed by three panels corresponding to the steps in the problem

resolution. The panels are called Root, Optimization and Visualization. A dialog box tells to the user what is the current step.
The Fig. 3.1 shows the integration of the steps followed in a typical execution. IDEAS generates automatically m-functions that are called in the different stages of the execution. All these functions share the name defined by the user(problem name). The function without any suffix is the ODE model. Table 3.1 shows the meaning of the suffixes of the routines. These functions are open to the user and can be modified or used in other user-defined routines.

Suffix	**Meaning**
cost	Evaluates the optimization criterion
load	Loads the data file
optim	Performs the optimization step
ploty	Plots the data fit
plotys	Plots the sensitivity trajectories
se	Augmented model state + state sensitivity equations
sy	Calculates the sensitivities of the outputs
out	Calculates the model outputs
uncert	Calculates the FIM

Table 3.1: Explanation of the suffixes of the functions generated by IDEAS

To illustrate the use of the toolbox, consider the next example.

3.1 Example

Consider the mathematical model

$$\dot{s} = -\frac{1}{Y} + D(s_i - s)\rho, \ \ s(0) = 30, \tag{3.1}$$

$$\dot{x} = \rho - \rho_d, \ \ x(0) = 20, \tag{3.2}$$

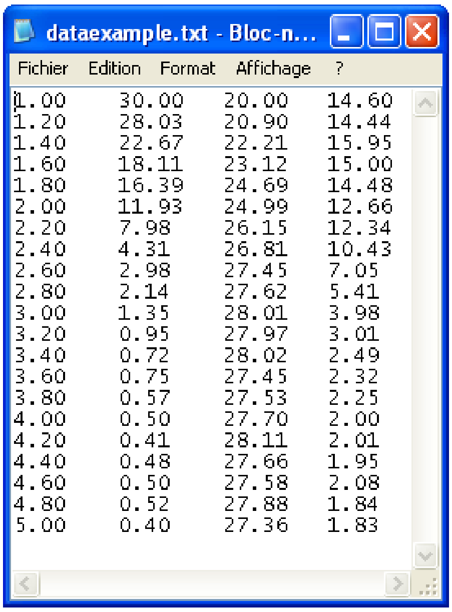
and the output vector is

$$\mathbf{y}_{\mathrm{m}} = (s, x, \frac{1-Y}{Y}\rho + \rho_d)^{\mathrm{T}}, \tag{3.3}$$

with

$$\rho = \mu_{max}\frac{xs}{K+s}$$
$$\rho_d = k_d x$$

The parameters to be estimated are K, μ_{max}, Y, k_d. $q = 0.041$ and $s_i = 40$.
The data file has to be provided in .txt file with the sampling time in the first column. The outputs (measurements) must follow the same order of the output vector (3.3). The data file (Fig. 3.2) has to be in the same folder with the source files. To use IDEAS, follow the next steps:


dataexample.txt - Bloc-n...

Fichier Edition Format Affichage ?

1.00	30.00	20.00	14.60
1.20	28.03	20.90	14.44
1.40	22.67	22.21	15.95
1.60	18.11	23.12	15.00
1.80	16.39	24.69	14.48
2.00	11.93	24.99	12.66
2.20	7.98	26.15	12.34
2.40	4.31	26.81	10.43
2.60	2.98	27.45	7.05
2.80	2.14	27.62	5.41
3.00	1.35	28.01	3.98
3.20	0.95	27.97	3.01
3.40	0.72	28.02	2.49
3.60	0.75	27.45	2.32
3.80	0.57	27.53	2.25
4.00	0.50	27.70	2.00
4.20	0.41	28.11	2.01
4.40	0.48	27.66	1.95
4.60	0.50	27.58	2.08
4.80	0.52	27.88	1.84
5.00	0.40	27.36	1.83

Figure 3.2: Data file

1. In Matlab, select the directory where the source files and the data are located
2. Types *ideas* in the command window. The license information of the toolbox appears. Click o, the button OK and the interface is displayed.
3. Follow the instructions given in the dialog box. For illustration we use *example* as the name of the problem and the data file is named *dataxample.txt*. Figures 3.3 - 3.10 show a typical execution.

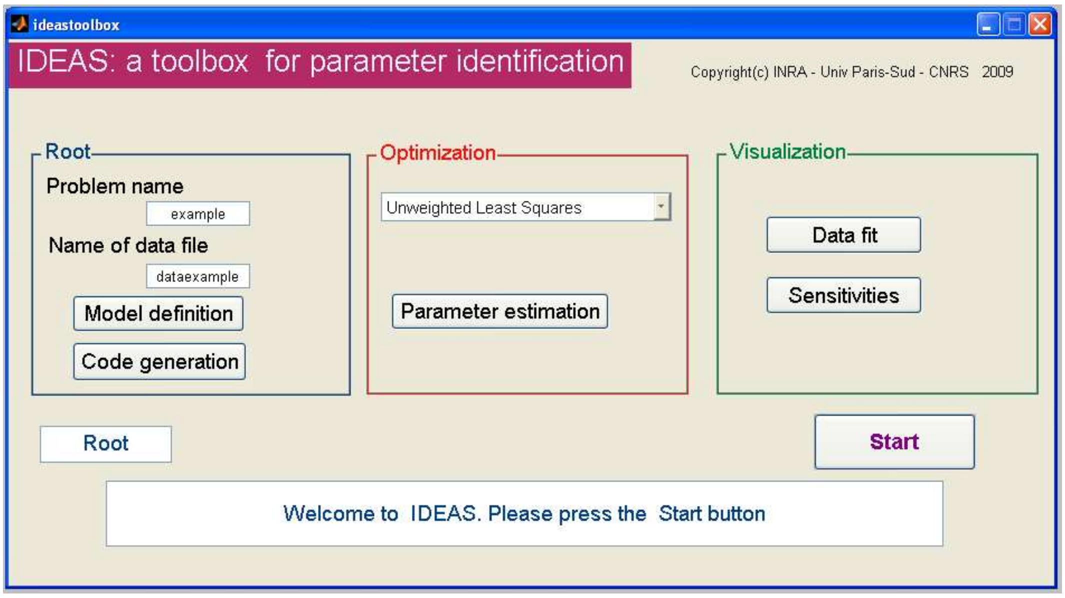


Figure 3.3: Writing the name of your problem and data file

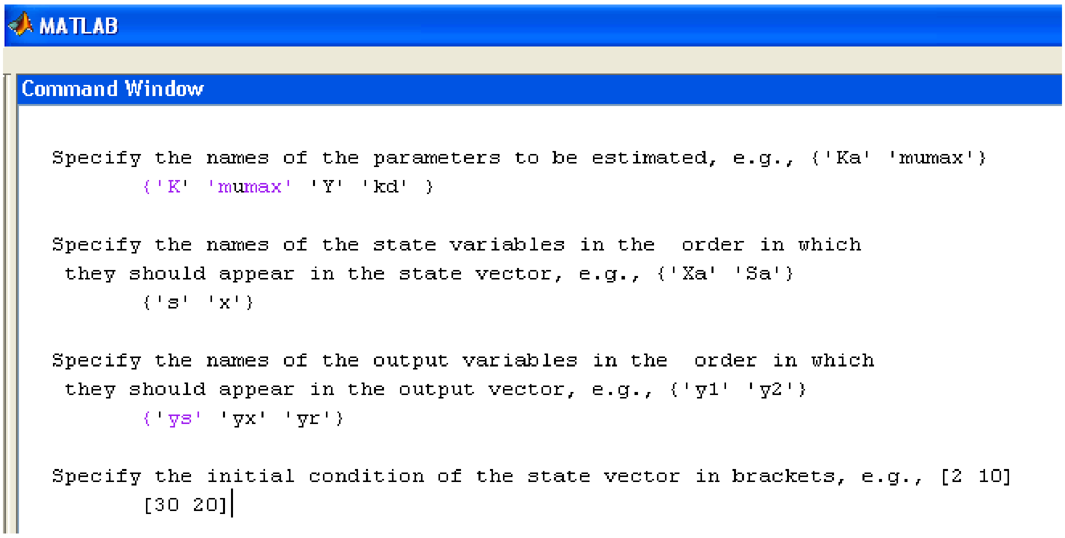


Figure 3.4: Defining model variables

Editor - D:\Rafael\ThEse\Programs_Rafael\IDEAS Toolbox_final version\IDEAS files\IDEAS_final\model_example.m

File Edit Text Cell Tools Debug Desktop Window Help

Stack: Base

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  Program to define the mathematical model
 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%   This part was automatically generated. Do not change
%   Symbolic definition of the parameters
syms K mumax Y kd
parameters = [ K mumax Y kd ];
%   Symbolic definition of the state variables
syms s x
Xx = [ s x ];
%   Symbolic definition of the outputs
syms ys yx yr
Yy = [ ys yx yr ];
% Initial conditions of the state variables
 Cinit = [  30  20  ];
 %   In this part the user defines the model. Follow the structure below
 %   Do not change the names of the items:   parameters, F, H

 % Step 1 : Write the constant parameters of the model, e.g. Kh = 10;
q= 0.042;     % 1/h, dilution rate
Si = 40;     %   Substrate concentration at the input, mg COD/l
 % Step 2 : Write the functions of the model, e.g.  ro = mumax * Xx(1)*Xx(2) / (K+Xx(1));
ro = mumax * s*x / (K+s);  % growth rate
rd = kd * x;                % decay rate
   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  The mathematical model is represented as follows :
   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
         % dx/dt = F(x,p)
         % y = H(x,p)

         % x has a dimension  n
         % F has a dimension  n
         % y has a dimension  q
         % H has a dimension  q

 % Step 3 : Write the elements of the vector F, e.g.
    % F(1) = -ro/Y  - Kh*Xx(1));
    F(1) =  -ro/Y  +q*(Si-s);
    F(2) = ro -rd;

 % Step 4 : Write the elements of the vector H, e.g.
    % H(1) = ro;
H(1) = s;
H(2) = x;
H(3) = (1-Y)*ro/Y + rd;
 % Step 5 : Save the file
 % Step 6 : close the file and return to the interface
```

model_example.m

Figure 3.5: Defining the mathematical model. This file is automatically generated and the user must specify the model according to the structure describe in the template

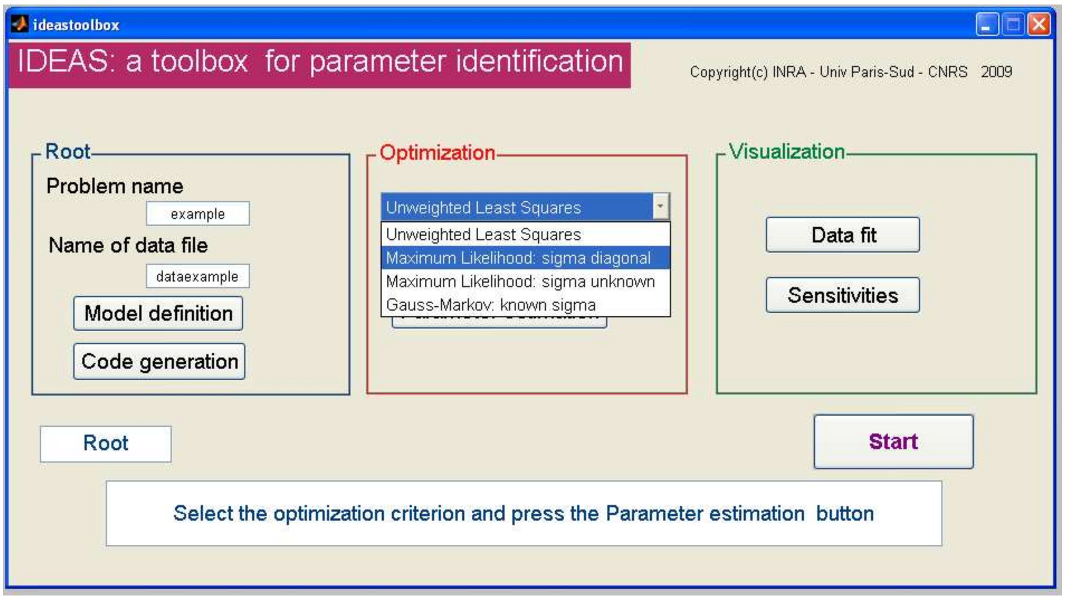


Figure 3.6: Press the Code generation button and then select the optimization criterion

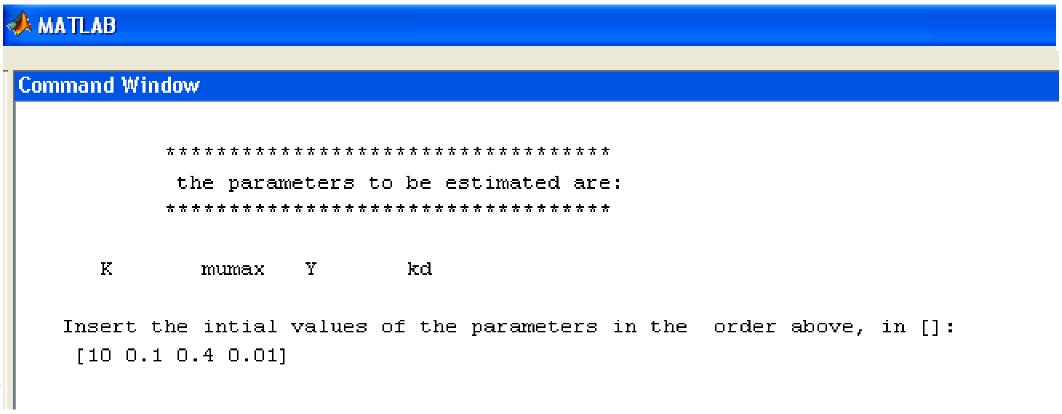
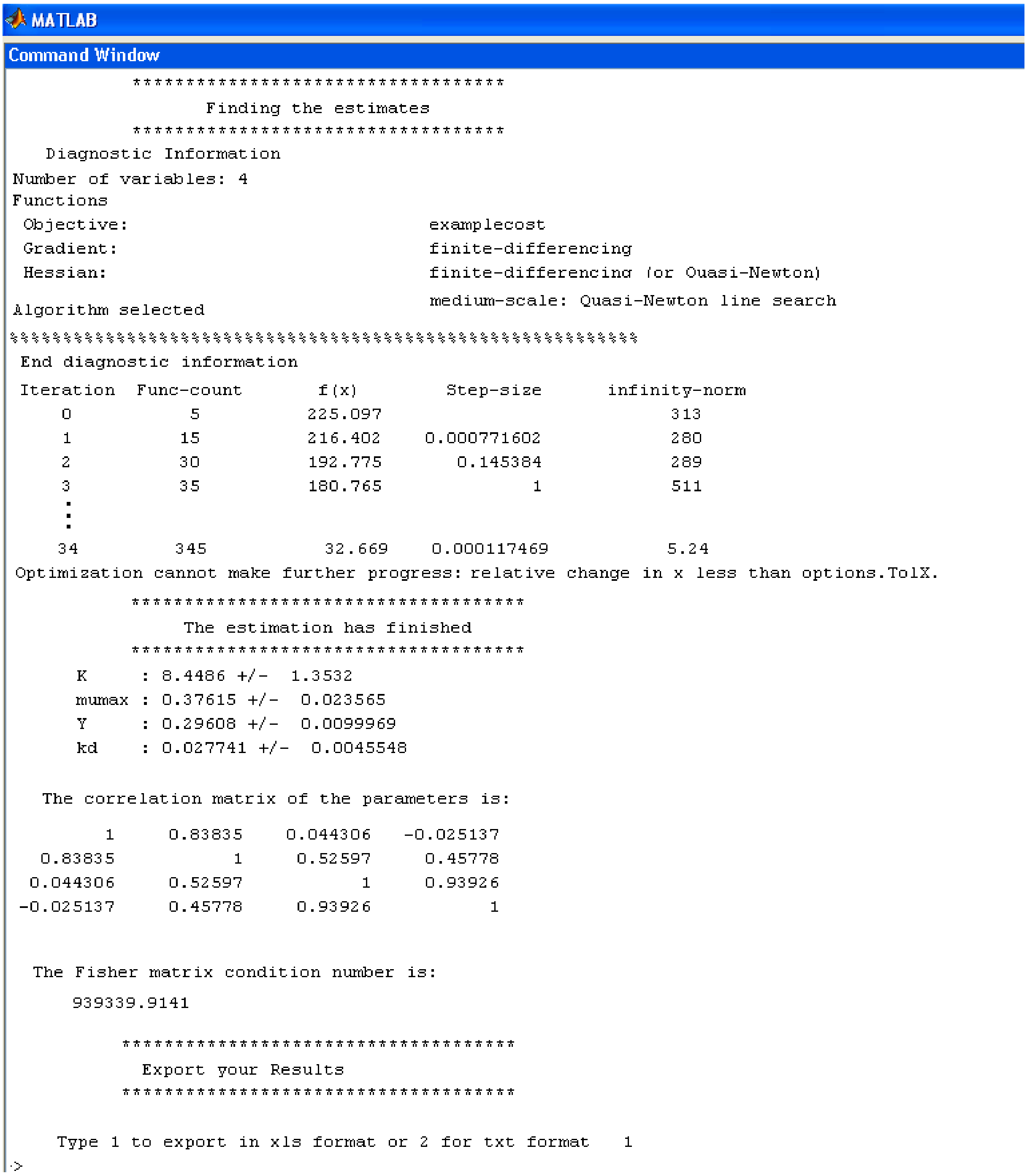


Figure 3.7: Writing initial guess of the parameter vector



```
MATLAB
Command Window
              *************************************
                     Finding the estimates
              *************************************
    Diagnostic Information
Number of variables: 4
Functions
 Objective:                              examplecost
 Gradient:                               finite-differencing
 Hessian:                                finite-differencing (or Quasi-Newton)
Algorithm selected                       medium-scale: Quasi-Newton line search
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
 End diagnostic information
 Iteration  Func-count       f(x)        Step-size      infinity-norm
     0           5          225.097                          313
     1          15          216.402     0.000771602          280
     2          30          192.775        0.145384          289
     3          35          180.765               1          511
     :
    34         345           32.669     0.000117469          5.24
Optimization cannot make further progress: relative change in x less than options.TolX.
              ***************************************
                   The estimation has finished
              ***************************************
      K     : 8.4486 +/-  1.3532
      mumax : 0.37615 +/-  0.023565
      Y     : 0.29608 +/-  0.0099969
      kd    : 0.027741 +/-  0.0045548

   The correlation matrix of the parameters is:

         1      0.83835     0.044306    -0.025137
   0.83835            1      0.52597      0.45778
  0.044306      0.52597            1      0.93926
 -0.025137      0.45778      0.93926            1


  The Fisher matrix condition number is:
      939339.9141

             ***************************************
               Export your Results
             ***************************************

    Type 1 to export in xls format or 2 for txt format    1
>
```

Figure 3.8: Estimation process. The user can choose to export the results in .xls or .txt format. The file with the resuls is named with the suffix _results

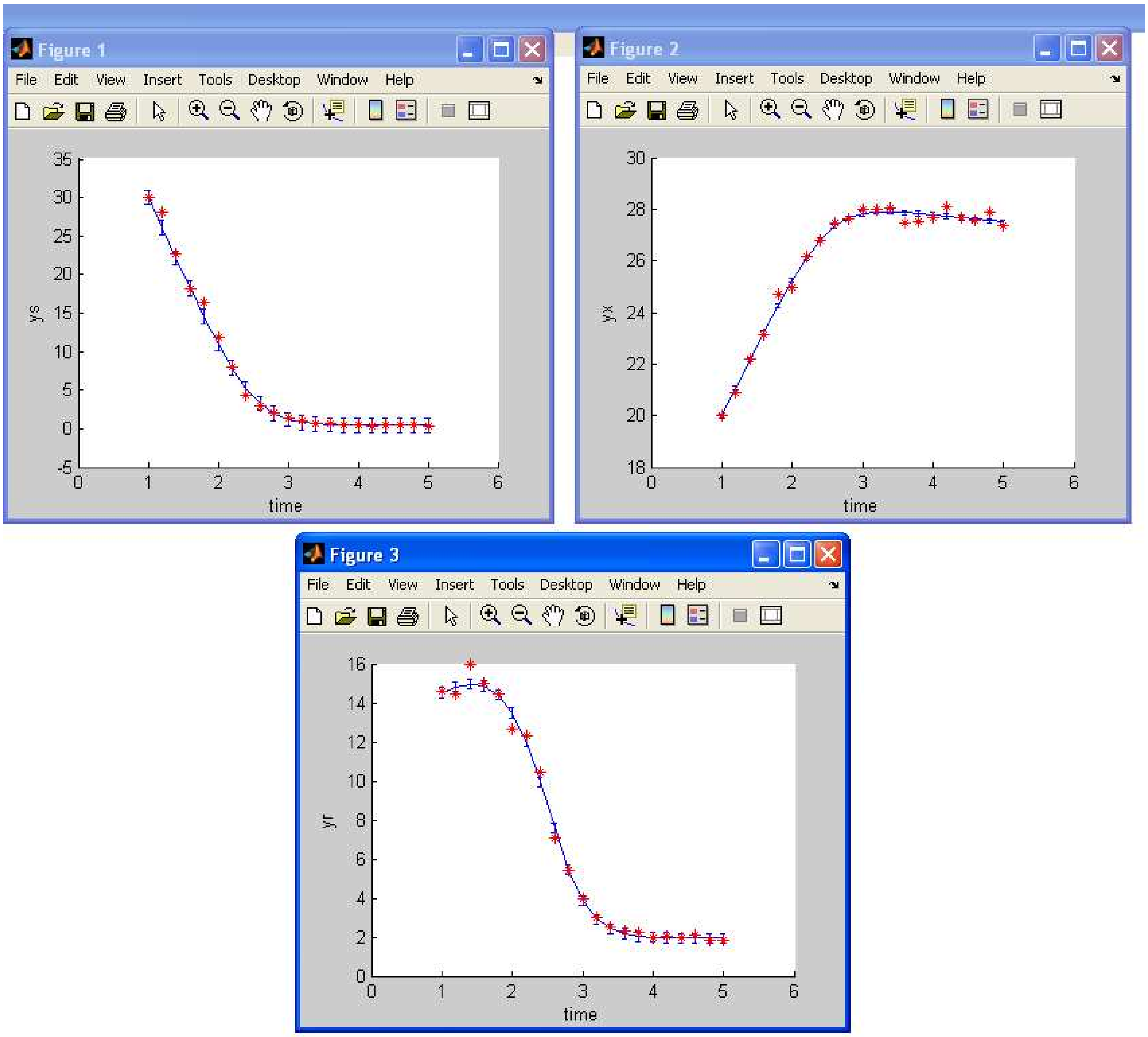


Figure 3.9: Press the Data fit button to display the match of the model against the data

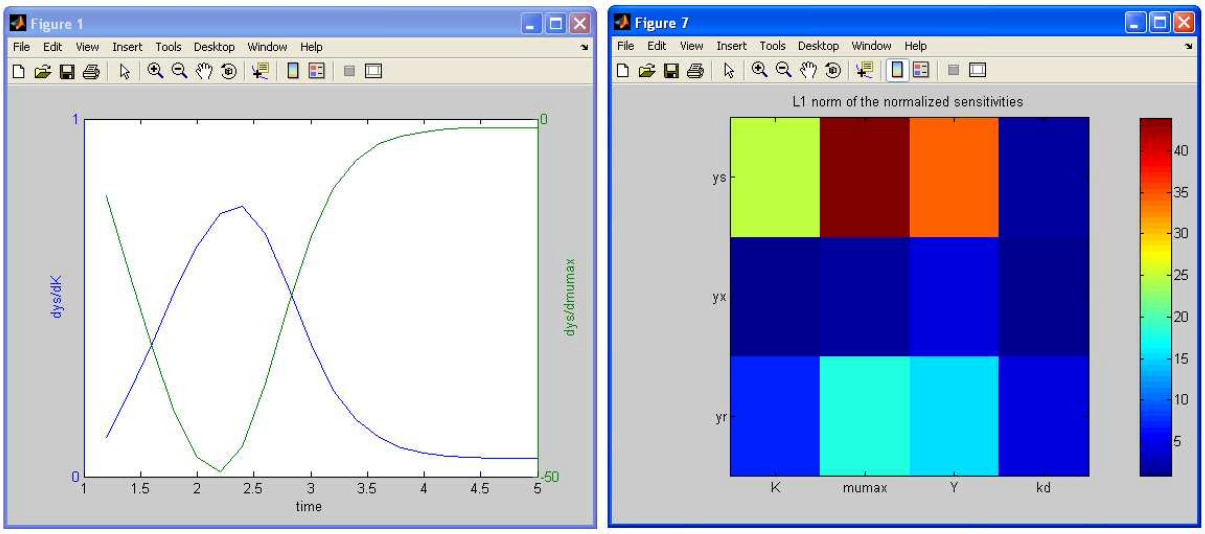


Figure 3.10: Press the Sensitivities button to display the trajectory of the sensitivities of the output. A final figure is a matrix representation of the L1 norm of the normalized sensitivities. The element k, j is computed as $\sum_{i=1}^{n_{\mathrm{t}}}\left|\frac{\widehat{\theta}_j}{y_{\mathrm{m}_k}(t_i,\widehat{\boldsymbol{\theta}})}\left[\frac{\partial y_{\mathrm{m}_k}}{\partial \theta_j}\right]_{(t_i,\widehat{\boldsymbol{\theta}})}\right|$

3.2 Some hints

The optimization step can be sensitive to numerical problems. If your estimation problem cannot be solved and you have as return warning messages, it can be due to the initialization of the parameter vector or errors in the integration of the ODE equations. You can try to make a reparametrization in order to work in log-base. You can enter to the functions with suffixes optim and cost to uncomment the instructions to work with this base. Advantage of this option is that the positivity of the parameters is guaranteed. Keep in mind that optimization is local, thus you should test the estimation with different parameter values for the initialization.

References

[1] Goodwin, G., and Payne, R. *Dynamic System Identification: Experiment Design and Data Analysis.* Academic Press, New York, 1977.

[2] Muñoz-Tamayo, R., Laroche, B., Leclerc, M., and Walter, E. IDEAS: a parameter identification toolbox with symbolic analysis of uncertainty and its application to biological modelling. In *15th Symposium on System Identification* (2009).

[3] Walter, E., and Pronzato, L. *Identification of Parametric Models from Experimental Data.* Springer, London, 1997.