IT Tools & Optimization BE 3 Report – 23/10/2019

Least Squares Problems: Identification of an Ecological System

Students: TRAN Gia Quoc Bao, AYHAN Sinan, ABDELLI Bilel

Introduction: In this experiment we seeked to solve an ecological problem using least squares method, in two cases: short-term and long-term measurements.

First, the notation of v was used to get:

$$\begin{cases} \dfrac{\dot{P}}{P} = r - P\dfrac{r}{K} - Qs = \begin{pmatrix} 1 \\ -P \\ -Q \\ 0 \\ 0 \end{pmatrix}^T \begin{pmatrix} r \\ \dfrac{r}{K} \\ s \\ \alpha \\ \beta \end{pmatrix} = y_1^T(P,Q).v \\ \dfrac{\dot{Q}}{Q} = -\alpha + P\beta = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \\ P \end{pmatrix}^T \begin{pmatrix} r \\ \dfrac{r}{K} \\ s \\ \alpha \\ \beta \end{pmatrix} = y_2^T(P,Q).v \end{cases}$$

Then, assuming the measurement period is sufficiently small, which means:

$$\frac{\dot{P}}{P} \approx \frac{P(t_{k+1}) - P(t_k)}{\tau P(t_k)};\ \frac{\dot{Q}}{Q} \approx \frac{Q(t_{k+1}) - Q(t_k)}{\tau Q(t_k)}$$

We had this approximation:

$$\tau \begin{pmatrix} P(t_k) & -P^2(t_k) & -Q(t_k)P(t_k) & 0 & 0 \\ 0 & 0 & 0 & -Q(t_k) & P(t_k)Q(t_k) \end{pmatrix} v = \begin{pmatrix} P(t_{k+1}) - P(t_k) \\ Q(t_{k+1}) - Q(t_k) \end{pmatrix} \leftrightarrow Lv = q$$

In the general case we repeated the form and found:

$$\tau \begin{pmatrix} P(t_1) & -P^2(t_1) & -Q(t_1)P(t_1) & 0 & 0 \\ 0 & 0 & 0 & -Q(t_1) & P(t_1)Q(t_1) \\ \dots & \dots & \dots & \dots & \dots \\ P(t_{N-1}) & -P^2(t_{N-1}) & -Q(t_{N-1})P(t_{N-1}) & 0 & 0 \\ 0 & 0 & 0 & -Q(t_{N-1}) & P(t_{N-1})Q(t_{N-1}) \end{pmatrix} v$$

$$= \begin{pmatrix} P(t_2) - P(t_1) \\ Q(t_2) - Q(t_1) \\ \dots \\ P(t_N) - P(t_{N-1}) \\ Q(t_N) - Q(t_{N-1}) \end{pmatrix} \leftrightarrow Av = B$$

We prepared a function to calculate the components of A and B using a loop:

```
function [A,B]=ComputeMatrices(Y,to)
    N=length(Y);
    B=zeros(2*(N-1),1);
    A=zeros(2*(N-1),5);
    for j = 1:(N-1)
        B(2*j-1,1)=  Y(j+1,1)-Y(j,1);
        B(2*j,1)=  Y(j+1,2)-Y(j,2);
        A(2*j-1,:)=to*[ Y(j,1) -Y(j,1)*Y(j,1) -Y(j,2)*Y(j,1) 0 0];
        A(2*j,:)=to*[ 0 0 0 -Y(j,2) Y(j,1)*Y(j,2)];

    end
  return
```

After testing with the given Ytest, we got the exact result as expected:

```
>> [Atest,Btest]=ComputeMatrices(Ytest,0.1)

Atest =

    0.0999   -0.0998   -0.0954         0         0
         0         0         0   -0.0955    0.0954
    0.0995   -0.0990   -0.0935         0         0
         0         0         0   -0.0939    0.0935
    0.0989   -0.0978   -0.0911         0         0
         0         0         0   -0.0921    0.0911
    0.0980   -0.0961   -0.0882         0         0
         0         0         0   -0.0900    0.0882


Btest =

   -0.0037
   -0.0160
   -0.0062
   -0.0183
   -0.0087
   -0.0206
   -0.0112
   -0.0229
```

Short measurement campaign:

We loaded the data1 and obtained the (rather long) A and B matrices with that.

Theoretically this allows us to calculate the v* using the least squares method because the size of the data given is just 50 and so this does not make the memory overflow nor contain data that are too outdated.

We solved for v* using the MATLAB command:

$$v_1{}^* = A_1 \backslash B_1 = (1.3981\ 1.1388\ 0.5392\ 0.2842\ 1.3357)^T$$

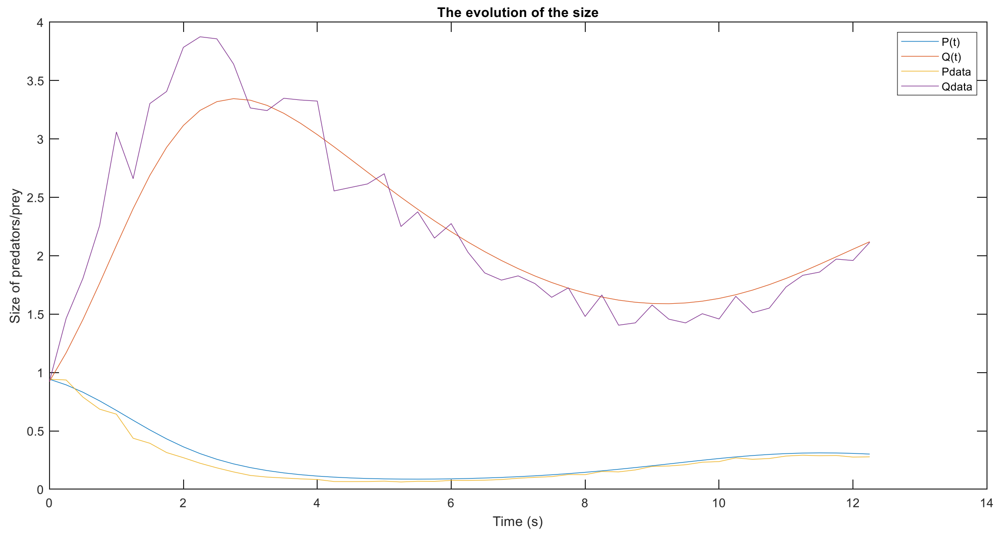
And the 5 parameters are:

$$r_1 = 1.3981; K_1 = 1.2277; s_1 = \ 0.5392\ ;\ \alpha_1 = 0.2842\ ;\ \beta_1 = 1.3357$$

And the norm of the residual associated to the solution is small:

$$||A_1 v_1{}^* - B_1|| = \sqrt{(A_1 v_1{}^* - B_1)^T (A_1 v_1{}^* - B_1)} = 1.6073$$

We solved the system of non-linear differential equations with ode45 and plot the result as well as the measured data1 on the same graph:



We see that the results obtained from solving the least squares problem is close to the results collected. That means the solution for the short-term case is correct.

Long measurement campaign:

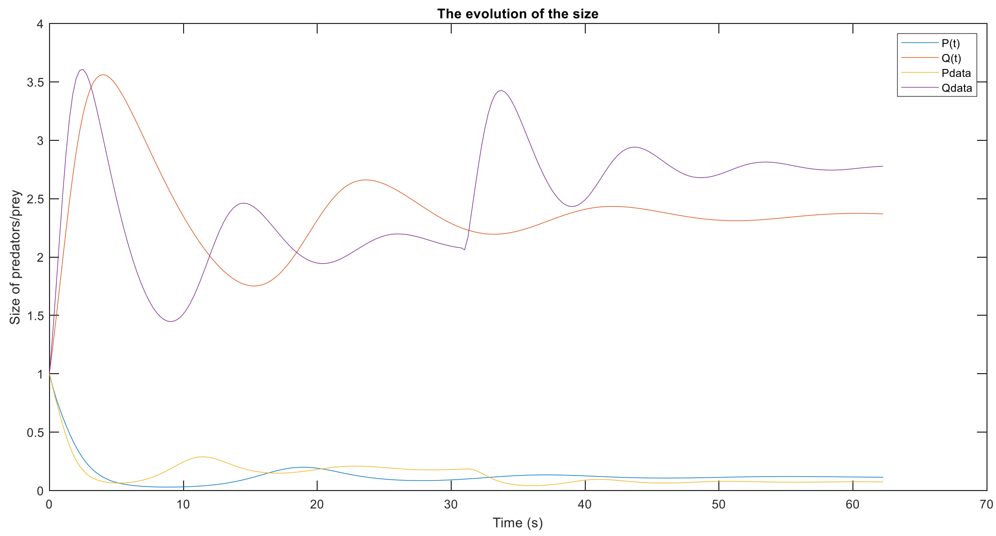
We repeated the problem with data2 and tried to compare the results:

$$v_2{}^* = A_2\backslash B_2 = (1.2125\ 1.2616\ 0.4533\ 0.1165\ 1.0060)^T$$

$$r_2 = 1.2125; K_2 = 0.9611; s_2 = 0.4533\ ;\ \alpha_2 = 0.1165;\ \beta_2 = 1.0060$$

$$\|A_2v_2{}^* - B_2\| = \sqrt{(A_2v_2{}^* - B_2)^T(A_2v_2{}^* - B_2)} = 0.7836$$

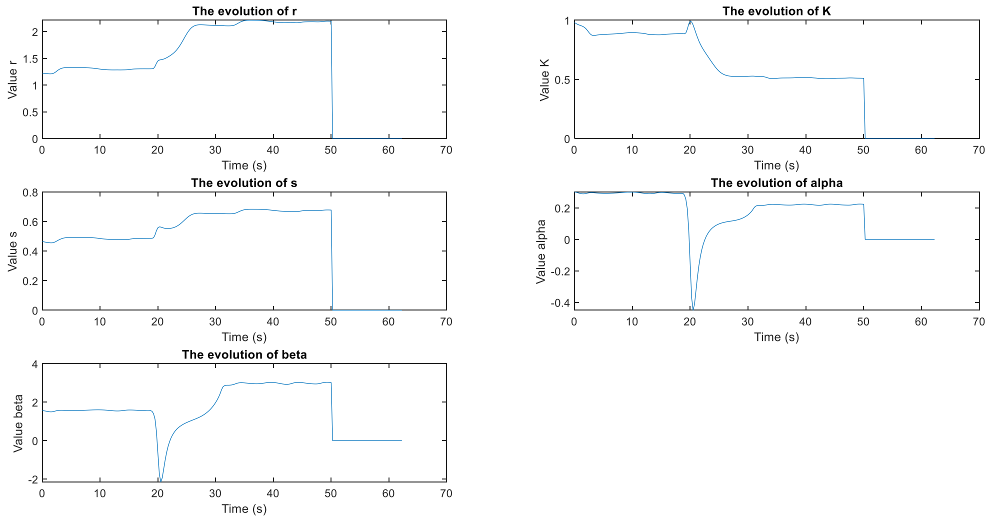
We solved the system of non-linear differential equations with ode45 and plot the result as well as the measured data1 on the same graph:



We saw that this time the lines do not fit each other very well. This is because we were paying so much attention on the old data that when there is some change in real life (the sudden change in the prey/predator population is maybe because of a disease, and this time the predator population rose suddenly at the middle of the period), our values do not converge to the real values anymore.

To avoid this we used a recursive solution with a forgetting factor $\lambda = 0.5$. We wish to plot the change in the 5 parameters with time.

First we started with the first values obtained from the short-term compaign but that caused discontinuities in our plots when there was a shift at N_p. So instead we started the recursive method from 0. This is the result where each parameter gets updated automatically.



The discrepancy between the measured and the simulated behaviours is due to various reasons:

- The fact that our measurements outnumber our parameters so we needed to use least squares to find the optimal solution.
- The fact that each time there is a discontinuity in the samples it badly affects our estimation. So we need recursive methods.
- The error when we discretized the continuous equations into the recursive ones.
- The noises that may affect our measurements.

Conclusion: We understood the use of least squares as well as the importance of using recursive methods to avoid the memory overflow and to keep our data updated.

Thank you for reading.