Screening active inputs in factor-rich models by factors' main effect and interaction properties

Andrea Saltelli
Joint Research Centre, European Commission, Ispra, Italy

2007

A great deal of additional information on the European Union is available on the Internet.
It can be accessed through the Europa server
http://europa.eu.int

Printed in Italy

Abstract

We introduce a new method for screening factors in mathematical or computational models with large numbers of factors. The method proposed here represents an improvement over the best available practice for this setting, the method of elementary effects. This approach is particularly efficient for factors screening of non-linear and non-additive models.

EUROPEAN COMMISSION
DIRECTORATE-GENERAL
Joint Research Centre

The mission of the Joint Research Centre is to provide customer-driven scientific and technical support for the conception, development, implementation and monitoring of European Union policies. As a service of the European Commission, the JRC functions as a reference centre of science and technology for the Community. Close to the policy-making process, it serves the common interest of the Member States, while being independent of special interests, whether private or national.

Publications Office
Publications.eu.int

Introduction

We introduce a new method for screening factors in mathematical or computational models with large numbers of factors. In factors screening one tries to identify active factors in a factor-rich model with a minimum number of computer simulations. The methods proposed here represents an improvement over the best available practice for this setting, the method of elementary effects (Morris 1991(7), Campolongo et al. 1999(1), Campolongo et al. 2000(2), Campolongo et al. 2007(3)). Reviews of screening methods for computer experiments are in Campolongo et al. 2000(2), Saltelli et al., 2007(10), while review on sensitivity analysis in general are in Saltelli et al. 2004(8), Saltelli et al., 2005(11), Helton et al., 2006(4). Resources for sensitivity are available at (6).

The idea is to explore the space of the input factors by combining steps along the X_i axis, where X_i is one of the factors, with steps along the $\mathbf{X}_{\sim i}$, an axis where all factors but X_i change their value. We anticipate that such an approach should be particularly efficient for factors screening of non-linear and non-additive models.

The method

A generic model 0.0.1 is considered:

$$Y = f(X_1, X_2, \ldots, X_k) \qquad (0.0.1)$$

The new design is exemplified for a case where the are $k = 15$ factors and each factors is explored over $l = 5$ different values or levels. The reason for these choices of k and l will be apparent in a moment. According to standard practice(1) the levels could correspond to equally spaced percentiles of the distribution of each factors.

Table 1 shows how to arrange simulations so that steps in the $\mathbf{X}_{\sim i}$ direction are generated.

A step along the $\mathbf{X}_{\sim 1}$ axis is obtained comparing comparing run r_1 with r_2, et coetera as shown in Table 2. E.g., a step along the $\mathbf{X}_{\sim 15}$ axis is obtained from from r_1 and r_6 as all factors but X_{15} change of levels between r_1 and r_6.

In the following r will indicate number of rows in $\mathbf{B}$ and r_j a generic row of the same matrix.

The scheme of table 2 is extended to include steps along the X_i axis as shown in matrix $\mathbf{B}$ Table 3.

r_i	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	2	2	1	3	3	3	3	3	3	3	3	3	3	3	3
4	3	3	3	3	2	1	4	4	4	4	4	4	4	4	4
5	4	4	4	4	4	4	4	3	2	1	5	5	5	5	5
6	5	5	5	5	5	5	5	5	5	5	5	4	3	2	1

Table 1: Base Matrix $\mathbf{B}$. Run number r_i and level selected for each factor. The first column indicates the simulation (or "run" as a simulation is termed in computer jargon) r_i from $i = 1$ to $i = 6$, X_j denotes factor j and the tabled integer number indicates the level at which the factor is kept, e.g. factor X_1 is kept at level 3 in simulation r_4.

$\mathbf{X}_{\sim i}$	run	run
1	2	1
2	3	2
3	3	1
4	4	3
5	4	2
6	4	1
7	5	4
8	5	3
9	5	2
10	5	1
11	6	5
12	6	4
13	6	3
14	6	2
15	6	1

Table 2: Scheme for the computation of effects of the type $\mathbf{X}_{\sim i}$ from the runs of matrix $\mathbf{B}$, Table (1).

The first column in Table 3 indicates the simulation/run number. If we call N the total number of runs we moved form $r = 6$ in $\mathbf{B}$ to $N = 36$ runs in $\mathbf{M}$. The original runs 1 to 6 have become in the order runs $1, 2, 5, 10, 17, 26$. The additional runs have been generated as follows:

Run 3 is a step in the X_1 direction taken from from run 1.
Run 4 is a step in the X_1 direction taken from from run 2.
Run 5 is the previous run 3 while:
Run 6 is a step in the X_2 direction taken from from run 2.
Run 7 is a step in the X_2 direction taken from from run 5.
and Run 8 is a step in the X_3 direction taken from from run 1.
Run 9 is a step in the X_3 direction taken from from run 5.

r_i	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	type	for factor
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	b	1,2,6,10,15
2	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	b	1,3,5,9,14
3	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	c1(1)	1
4	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	c2(1)	1
5	2	2	1	3	3	3	3	3	3	3	3	3	3	3	3	b	2,3,4,8,13
6	1	3	2	2	2	2	2	2	2	2	2	2	2	2	2	c2(2)	2
7	2	3	1	3	3	3	3	3	3	3	3	3	3	3	3	c5(2)	2
8	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	c1(3)	3
9	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	c5(3)	3
10	3	3	3	3	2	1	4	4	4	4	4	4	4	4	4	b	4,5,6,7,12
11	2	2	1	4	3	3	3	3	3	3	3	3	3	3	3	c5(4)	4
12	3	3	3	4	2	1	4	4	4	4	4	4	4	4	4	c10(4)	4
13	1	2	2	2	3	2	2	2	2	2	2	2	2	2	2	c2(5)	5
14	3	3	3	3	3	1	4	4	4	4	4	4	4	4	4	c10(5)	5
15	1	1	1	1	1	2	1	1	1	1	1	1	1	1	1	c1(6)	6
16	3	3	3	3	2	2	4	4	4	4	4	4	4	4	4	c10(6)	6
17	4	4	4	4	4	4	4	3	2	1	5	5	5	5	5	b	7,8,9,10,11
18	3	3	3	3	2	1	5	4	4	4	4	4	4	4	4	c10(7)	7
19	4	4	4	4	4	4	5	3	2	1	5	5	5	5	5	c17(7)	7
20	2	2	1	3	3	3	3	4	3	3	3	3	3	3	3	c5(8)	8
21	4	4	4	4	4	4	4	4	2	1	5	5	5	5	5	c17(8)	8
22	1	2	2	2	2	2	2	2	3	2	2	2	2	2	2	c2(9)	9
23	4	4	4	4	4	4	4	3	3	1	5	5	5	5	5	c17(9)	9
24	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	c1(10)	10
25	4	4	4	4	4	4	4	3	2	2	5	5	5	5	5	c17(10)	10
26	5	5	5	5	5	5	5	5	5	5	5	4	3	2	1	b	11,12,13,14,15
27	4	4	4	4	4	4	4	3	2	1	*4	5	5	5	5	c17(11)	11
28	5	5	5	5	5	5	5	5	5	5	*4	4	3	2	1	c26(11)	11
29	3	3	3	3	2	1	4	4	4	4	4	5	4	4	4	c10(12)	12
30	5	5	5	5	5	5	5	5	5	5	5	5	3	2	1	c26(12)	12
31	2	2	1	3	3	3	3	3	3	3	3	3	4	3	3	c5(13)	13
32	5	5	5	5	5	5	5	5	5	5	5	4	4	2	1	c26(13)	13
33	1	2	2	2	2	2	2	2	2	2	2	2	2	3	2	c2(14)	14
34	5	5	5	5	5	5	5	5	5	5	5	4	3	3	1	c26(14)	14
35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	c1(15)	15
36	5	5	5	5	5	5	5	5	5	5	5	4	3	2	2	c26(15)	15

Table 3: Matrix **M** and usage of runs to compute factors effect. Run type b=base run. Run type ci(j)=copy of run r_i with the level of factor X_j increased of one step. There are r=6 base runs, so that the number of factors is $k = 6(5/2) = 15$. The total number of runs is 6 base runs plus $2 * k$ additional runs for a total of $r^2 = 36$ simulations. Each base run is cloned exactly five times. Each input factor is explored over five levels. The exception is marked with asterisk - the 4's in runs 27 and 28 should have been 6's. This type of exception occurs exactly twice regardless of number of factors k. Each base run is used for 5 different factors. **Generalising:** a number r of base runs can be used for $k = \frac{r(r-1)}{2}$ factors explored over $l = (r-1)$ levels at a cost of $N = r^2$ simulations. Each base run is used for exactly $(r-1)$ factors while each non-base (clone) run is used only for one factor.

Run 10 is the previous run 4.

Analogously:

Run 11 is a step in the X_4 direction taken from from run 5.

Run 12 is a step in the X_4 direction taken from from run 10.

Run 13 is a step in the X_5 direction taken from from run 2.

Run 14 is a step in the X_5 direction taken from from run 10.

Run 15 is a step in the X_6 direction taken from from run 1.

Run 16 is a step in the X_6 direction taken from from run 10.

It should be clear how we have proceeded for the other runs. Note that run 17 is the previous run 5 and run 26 the previous run 6.

Let us now discuss briefly the base matrix $\mathbf{B}$ and why we have chosen to exemplify the approach with $k = 15$ factors.

In $\mathbf{B}$ for a generic row or run number r_j, we can estimate $r_j - 1$ steps of the $\mathbf{X}_{\sim i}$ type, the simulation number $r_j - 1$ can give $r_j - 1 - 1$ steps and so on, e.g. we compute the five steps $\mathbf{X}_{\sim 11}$ to $\mathbf{X}_{\sim 15}$ from row 6, four steps $\mathbf{X}_{\sim 7}$ to $\mathbf{X}_{\sim 10}$ from row 5 and so on).

Thus the total number k of steps of the $\mathbf{X}_{\sim i}$ type which one can estimate with r simulations is simply

$$k = r - 1 + r - 2 + ...1 = \frac{r(r-1)}{2} \tag{0.0.2}$$

Hence for a total run number of $r = 3, 4, 5, 6, 7, 8, 9 \ldots$ simulations we can compute steps of the $\mathbf{X}_{\sim i}$ type for $k = 3, 6, 10, 15, 21, 28, 36 \ldots$ factors respectively in matrix $\mathbf{B}$.

In other words, the number of matrix $\mathbf{B}$ simulations for a given number of factors k is the solution of the quadratic equation above 0.0.2, e.g.

$$r = \frac{1 + (1 + 8k)^{\frac{1}{2}}}{2} \tag{0.0.3}$$

that has meaningful solutions for $k = 3, 6, 10, 15, 21, 28, 36 \ldots$ factors corresponding to $r = 3, 4, 5, 6, 7, 8, 9 \ldots$ simulations in $\mathbf{B}$ as just mentioned. In the matrix $\mathbf{B}$ given as example, $k = 15$ and $r = 6$, which the right number to explore 5 different levels for each factor. It is easy to see that the number of level explored is in general $l = r - 1$.

Note that the designs for $k = 3, 6, 10$ are embedded into $\mathbf{B}$ or $\mathbf{M}$. One just has to ignore the unneeded right-most columns. With $k = 6$, $r = 4$ we would have explored conveniently 3 levels and with $k = 10$, $r = 5$ we would have explored 4 levels. Given that the present is a screening method, we would anticipate that it will applied to a k higher rather than lower than 15.

Note that in runs 27 and 28 we have moved levels from 5 to 4 instead than from 5 to 6, in order not to have isolated levels 6 explored. This technicality has to be kept in mind with different values of k as well to ensure that there will not be a level explored only twice in $\mathbf{M}$. This exception will systematically occur twice in $\mathbf{M}$ irrespective of k.

Given that our algorithm in matrix $\mathbf{M}$ also need two additional simulations for each factor to compute steps along the X_i axis, the total numbers of simulations for our design is the solution of equation 0.0.3 plus twice the number of factors k.

Hence $N = r + 2r(r-1)/2 = r^2$. In the case of matrix $\mathbf{M}$ where $r = 6$ this makes $N = 36$.

Having illustrated the sample design, we now move to discuss how effects are estimated. Let us refer to factor X_1 in matrix $\mathbf{M}$ to exemplify our approach.

Let us call Y_i the model value computed on the $i-th$ row of $\mathbf{M}$. We start with run r_3 which is simply a copy of run r_1 were factor X_1 has been moved from level (1) to level (2), and run $r = 4$ which is a copy of run r_2 were again factor X_1 has been moved from level (1) to level (2). Thus both

$$EE_a(1) = \frac{|Y_1 - Y_3|}{|Y_1|}$$

and

$$EE_b(1) = \frac{|Y_2 - Y_4|}{|Y_2|}$$

Can be taken as estimate of the so-called main effect of X_1 (see Figure 1).

Thus the elementary or first order effect $EE(1)$ of X_1 can be estimated via

$$EE(1) = \frac{1}{2}\left(|EE_a(1) + EE_b(1)|\right) \tag{0.0.4}$$

It is also clear that the both the couple of runs $1, 2$ and $3, 4$ provide an effect of moving all factors but X_1. We hence pose that an estimate of the non-linear and interaction effects of factor X_1 can be gauged by

$$EI_a(1) = ||Y_1 - Y_2| - |Y_3 - Y_4|| \tag{0.0.5}$$

Equation 0.0.5 says that if the effect of $\mathbf{X}_{\sim 1}$ varies between two different levels of X_1 then there must be a non-linear or interaction effect involving factor X_1.

On the same ground if the effect of X_1 computed at different points of $\mathbf{X}_{\sim 1}$ varies, this should as well provide information on non-linear and interaction effects of X_1, hence

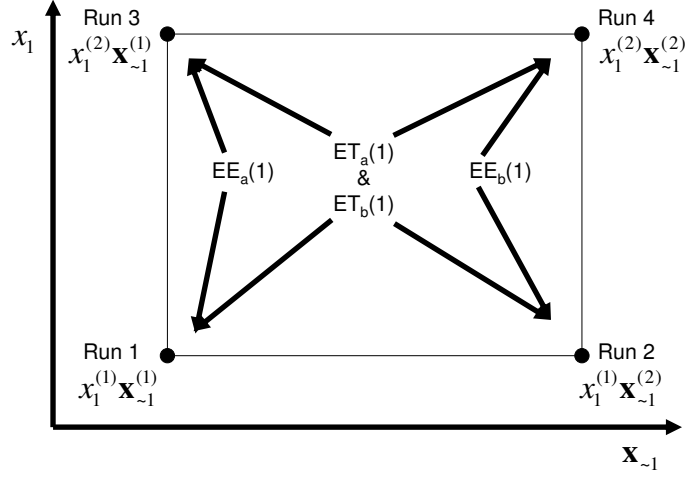$$EI_b(1) = ||Y_1 - Y_3| - |Y_2 - Y_4|| \tag{0.0.6}$$

Figure 1: Computational scheme for the effects of factor X_1 in matrix $\mathbf{M}$, Table 3. Note that effects can be obtained by comparing points both horizontally and vertically.

In the end an overall effect on non-linear and interaction terms in X_1 can be estimated by:

$$EI(1) = \frac{1}{2} |EI_a(1) + EI_b(1)| \tag{0.0.7}$$

EE and EI terms for the other factors can be similarly explored.

Note that the EI measure just described excludes the first order effect of a factor and is hence not a proxy for the total sensitivity index as described by Homma and Saltelli, 1996(5).

An alternative measure which could play as a proxy of the total sensitivity index would be the following:

$$ET_a(1) = \frac{1}{2} \left[\frac{||Y_1 - Y_2| - |Y_1 - Y_4||}{|Y_1|} + \frac{||Y_2 - Y_1| - |Y_2 - Y_3||}{|Y_2|} \right] \tag{0.0.8}$$

Equation 0.0.8 says that the difference between moving all and moving all-but-X_1 must tell us something about the effect of X_1, inclusive of its interaction with non-X_1. Similarly:

$$ET_b(1) = \frac{1}{2} \left[\frac{||Y_3 - Y_4| - |Y_3 - Y_2||}{|Y_3|} + \frac{||Y_4 - Y_3| - |Y_4 - Y_1||}{|Y_4|} \right] \qquad (0.0.9)$$

And proceeding analogously to the approach taken for $EI(1)$ we can estimate the total effect of X_1 as the modulus average of the two:

$$ET(1) = \frac{1}{2} |ET_a(1) + ET_b(1)| \qquad (0.0.10)$$

As mentioned the present approach favours number of factors such as $k = 3, 6, 10, 15, 21, 28, 36 \ldots$ and the corresponding costs are in Table 4.

r	number of factors k	number of simulations N
3	3	9
4	6	16
5	10	25
6	15	36
7	21	49
8	28	64
9	36	81
10	45	100
11	55	121
12	66	144
13	78	169
14	91	196
15	105	225
16	120	256
17	136	289
18	153	324
19	171	361
20	190	400

Table 4: Number of factors k and total number of simulations N corresponding to a given column dimension of matrix $\mathbf{B}$.

These numbers have to be compared with the cost of the elementary effects methods.

When using this methods(1) a number of trajectories t is launched into the space of the input factors. In each trajectory all steps are taken along the X_i axes (no movements over the $\mathbf{X}_{\sim i}$ axes) and only elementary effects are computed using systematically adjacent steps of the trajectory. Within each trajectory, composed of $k+1$ lattice points, one factor at a time is increased till all k dimensions have been spanned. Each trajectory thus generate exactly k elementary effects, i.e. one elementary effect per factor.

The cost C of this procedure is thus the number of trajectories t times the number of factors plus one, $C = t(k+1)$. Given that t can be as high as 10 to produce good results, it is easy to say that the approach developed in the present paper can be convenient.

This intuition has its basis on the fact that in the classic elementary effects method each point is used either once (the trajectory extremes) or twice (all other points in a given trajectory t), while in the present design runs are used much more exhaustively.

In general in matrix $\mathbf{B}$ each run is used to compute effects for a total of $r-1$ factors. Thus for $k = 15$ example where $r = 6$, each run is used in the estimation of five different effects as can be easily verified (run r_1 is used to estimate factors $X_1, X_2, X_6, X_{10}, X_{15}$ (see Table 3)).

Thus even if we assume to use not just one but t' different versions of matrix $\mathbf{M}$, scrambling the assignment of factors to columns, the costs to be compared would be

$$C = t(K + 1) \tag{0.0.11}$$

for the classic elementary effect method against

$$C' = t'N = t'r^2 \tag{0.0.12}$$

Given that r^2 is more than the double of $k + 1$, with the ratio $\frac{2}{k+1}$ tending to 2 for large values of k, it will be $C' < C$ provided that for $t' < 5$ one produce results comparable in quality to the classic approach at $t = 10$.

Comparison with the a quantitative method

There are points of similarities between the approach presented in this paper and that of Saltelli, 2002(12). This is an improvement of the Sobol' method(13) for the computation of the so-called variance based measures.

A variance based first order effect for a given factor X_i can be written as

$$V_{X_i}\left(E_{\mathbf{X}_{\sim i}}\left(Y \mid X_i\right)\right) \tag{0.0.13}$$

The meaning of the inner expectation operator is that the mean of Y it is taken of all possible values on values non-X_i i.e. over all possible values of $\mathbf{X}_{\sim i}$, while keeping X_i fixed. The outer variance is taken over all possible values of X_i.

The associated sensitivity measure (first order sensitivity coefficient) is written as:

$$S_i = \frac{V_{X_i}\left(E_{\mathbf{X}_{\sim i}}\left(Y \mid X_i\right)\right)}{V(Y)} \tag{0.0.14}$$

Because of relation:

$$E_{X_i}\left(V_{\mathbf{X}_{\sim i}}\left(Y \mid X_i\right)\right) + V_{X_i}\left(E_{\mathbf{X}_{\sim i}}\left(Y \mid X_i\right)\right) = V(Y). \tag{0.0.15}$$

S_i will always be between zero and unit. It measures the first order (e.g. additive) effect of X_i on the model.

Another popular variance based measure is the total effect term or index(5; 12):

$$S_{Ti} = \frac{E_{\mathbf{X}_{\sim i}}\left(V_{X_i}\left(Y \mid \mathbf{X}_{\sim i}\right)\right)}{V(Y)} \tag{0.0.16}$$

S_{Ti} measures the total effect (both first order and higher - e.g. interactions - of factor X_i.

Variance based methods are quantitative so that these measure are not simply estimated over levels of the factors but over the entire factors distribution using customarily Monte Carlo methods of various sophistication.

Sampling matrices will be designed to allow the computation of measures 0.0.14 and 0.0.16.

Following Saltelli 2002(12) we imagine to have two independent sampling matrices $\mathbf{A}$ and $\mathbf{B}$. We can write A as

$$\mathbf{A} = \mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(a)} \tag{0.0.17}$$

where $\mathbf{X}_i^{(a)}$ will be a column vector of length N (the number of simulations in the matrix) and $\mathbf{X}_{\sim i}^{(a)}$ will be a matrix of dimension N times $k-1$.

Analogously

$$\mathbf{B} = \mathbf{X}_i^{(b)}, \mathbf{X}_{\sim i}^{(b)} \tag{0.0.18}$$

Two more matrices can be constructed as

$$\mathbf{A_b} = \mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(b)} \tag{0.0.19}$$

$$\mathbf{B_a} = \mathbf{X}_i^{(b)}, \mathbf{X}_{\sim i}^{(a)} \tag{0.0.20}$$

The theory of variance based measures is based on computing S_i from either the couple of matrices $\mathbf{A}, \mathbf{A_b}$ or $\mathbf{B}, \mathbf{B_a}$(13).

Analogously S_{Ti} can be computed either using the couple $\mathbf{A}, \mathbf{B_a}$ or matrices $\mathbf{B}, \mathbf{A_b}$ (see Table 5).

Saltelli 2002(12) noted that all what is needed to compute both the set of all S_i and S_{Ti} for the k factors is the triplet of matrices $\mathbf{A}, \mathbf{B}, \mathbf{B_a}$ or alternatively the triplet matrices $\mathbf{A}, \mathbf{B}, \mathbf{A_b}$. If we use e.g. the former triplet we can compute all S_i and S_{Ti} using the scheme in Table 6:

Note that $2N$ simulations are needed for computing Y corresponding to matrices $\mathbf{A}, \mathbf{B}$ while kN simulations are needed to compute Y for matrix $\mathbf{B_a}$ for all factors. As

matrix	matrix	effect
$\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(a)}$	$\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(b)}$	S_i
$\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(a)}$	$\mathbf{X}_i^{(b)}, \mathbf{X}_{\sim i}^{(a)}$	S_{Ti}
$\mathbf{X}_i^{(b)}, \mathbf{X}_{\sim i}^{(b)}$	$\mathbf{X}_i^{(b)}, \mathbf{X}_{\sim i}^{(a)}$	S_i
$\mathbf{X}_i^{(b)}, \mathbf{X}_{\sim i}^{(b)}$	$\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(b)}$	S_{Ti}

Table 5: Possible ways of computing S_i and S_{Ti} from 5

matrix	matrix	effect
$\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(a)}$	$\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(b)}$	S_i
$\mathbf{X}_i^{(b)}, \mathbf{X}_{\sim i}^{(b)}$	$\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(b)}$	S_{Ti}

Table 6: How to compute S_i and S_{Ti} according to (12)

a result the cost of this quantitative method is $N(k+2)$ with N usually a large number (1000 or higher).

This has similarities with the approach taken in the present paper, where the computation of the sensitivity effects are based of 'squares' in the hyper-lattice such as Table 7.

For example, the square for factor X_1 in matrix 3 is in Figure 1

A comparison of the efficiency of the Saltelli 2002(12) method against the approach of the present paper could be based on the number of times a given point or simulation is used to compute an effect.

In Saltelli 2002(12) there are N simulation corresponding to points $\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(a)}$ and N simulations corresponding to points $\mathbf{X}_i^{(b)}, \mathbf{X}_{\sim i}^{(b)}$.

Each if these $2N$ simulations or points is used k times (see Table 6), once for each factor. There are instead kN points of the type $\mathbf{X}_i^{(a)}, \mathbf{X}_{\sim i}^{(b)}$ (N points for each factor) and each of these points is used twice, once for S_i and once for S_{Ti}. Hence a weighted average of simulation usage can be written as:

point	point
$x_i^{(j)}, \mathbf{x}_{\sim i}^{(j)}$	$x_i^{(j)}, \mathbf{x}_{\sim i}^{(j+1)}$
$x_i^{(j+1)}, \mathbf{x}_{\sim i}^{(j)}$	$x_i^{(j+1)}, \mathbf{x}_{\sim i}^{(j+1)}$

Table 7: Generic bi-dimensional square in the hyper-lattice

$$U = \frac{2kN + 2kN}{(k+2)N} = \frac{4k}{k+2} \qquad (0.0.21)$$

i.e. below and approaching 4 for increasing k.

Let us compare this with the apporach developed in the repsent paper. There are r^2 simulations or points in matrix **B**. Each of the r base runs is used for $r-1$ factors and 3 effects per factor are computed (e.g. from Y_1 one computes $EE(1)$, $ET_a(1)$ and $ET_b(1)$, if we do not count $EI_a(1)$ and $EI_b(1)$). Each of the remaining $2k$ runs are used only for one factor for the same three effects. Hence

$$W = \frac{3r(r-1) + 3(2k)}{r^2} = \frac{3r(r-1) + 6\frac{r(r-1)}{2}}{r^2} = \frac{6(r-1)}{r} \qquad (0.0.22)$$

which is below and approaching 6 for increasing r. As $W > U$ the method presented in the present work is about 50% more efficient than the one presented in Saltelli 2002(12).

Conclusion

A method has been developed to accelerate screening of factors in factor-rich models with the interesting property that both first order effects, interaction effects and total effects are gauged. Further work is needed to test the present method against available benchmarks for sensitivity analysis tools as described in (10).

Bibliography

[1] Campolongo, F., Tarantola, S. and Saltelli, A. (1999). Tackling quantitatively large dimensionality problems. Computational Physics Communication 117, 75-85.

[2] Campolongo, F., Kleijnen, J. and Andres, T., Screening Methods, in Saltelli, A., Chan, K. and Scott, M. (Eds.) (2000). Sensitivity Analysis. Wiley, New York, pp. 65-80.

[3] Campolongo, F., Cariboni, J., Saltelli, A., Simplifying a large chemical reaction model via an effective screening design, to appear on Environmental Modelling & Software, 2007.

[4] Helton, J.C., Johnson, J.D., Sallaberry, C. J., Storlie, C.B., 2006, Survey of sampling based methods for uncertainty and sensitivity analysis, SANDIA Report SAND2006-2901.

[5] Homma T., and Saltelli A., 1996 , Importance measures in global sensitivity analysis of model output, 1996, Reliability Engineering and System Safety, 52(1), 1-17.

[6] Sensitivity analysis web at JRC http://sensitivity-analysis.jrc.cec.eu.int/

[7] Morris, M. D. (1991). Factorial sampling plans for preliminary computational experiments. Technometrics 33(2),161-174.

[8] Saltelli A. Tarantola S., Campolongo, F. and Ratto, M., 2004, Sensitivity Analysis in Practice. A Guide to Assessing Scientific Models, John Wiley and Sons publishers.

[9] A. Saltelli, K. Chan, M. Scott, Editors,2000, Sensitivity Analysis, John Wiley and Sons publishers, Probability and Statistics series.

[10] Saltelli, A., Andres, T., Campolongo, F., Cariboni J., Gatelli D., Ratto, M., Saisana, M., Tarantola, S., Sensitivity analysis of scientific models, John Wiley and Sons publishers, to appear in winter 2007.

[11] Saltelli, A., M. Ratto, S. Tarantola and F. Campolongo (2005) Sensitivity Analysis for Chemical Models, Chemical Reviews, 105(7) pp 2811 - 2828.

[12] Saltelli, A., 2002, Making best use of model valuations to compute sensitivity indices, Computer Physics Communications, 145, 280-297.

[13] Sobol', I. M., Sensitivity estimates for nonlinear mathematical models, Matematicheskoe Modelirovanie 2 (1990) 112-118, in Russian, translated in English as I. M. Sobol', Sensitivity analysis for non-linear mathematical models, Mathematical Modelling and Computational Experiment 1 (1993) 407-414.

European Commission

EUR 22741 EN– DG Joint Research Centre, Institute for the Protection ans Security of the Citizen IPSC

Title: Screening Active Inputs in Factor-Rich Models by Factors' Main Effect and Interaction Properties

Author: Andrea Saltelli

Abstract
 We introduce a new method for screening factors in mathematical or computational models with large numbers of factors. The method proposed here represents an improvement over the best available practice for this setting, the method of elementary effects. This approach is particularly efficient for factors screening of non-linear and non-additive models.

EUROPEAN COMMISSION
DIRECTORATE-GENERAL
Joint Research Centre

The mission of the Joint Research Centre is to provide customer-driven scientific and technical support for the conception, development, implementation and monitoring of European Union policies. As a service of the European Commission, the JRC functions as a reference centre of science and technology for the Community. Close to the policy-making process, it serves the common interest of the Member States, while being independent of special interests, whether private or national.

Publications Office
Publications.eu.int