How to compute variance-based sensitivity indicators with your spreadsheet software

Elmar Plischke[a]

[a]*Institute of Disposal Research, Clausthal University of Technology, 38678 Clausthal-Zellerfeld, Germany*

Abstract

The use of sensitivity indicators is explicitly recommendend by authorities like the EC, the US EPA and others in model valuation and audit. In this note, we want to draw the attention to a numerical efficient algorithm that computes first order global sensitivity effects from given data using a discrete cosine transformation.

Key words: Variance-based sensitivity indicators, first order effects, discrete cosine transformation

1. Introduction

The use of sensitivity analysis for the output of simulation models is propagated in the books by Saltelli et al. [10, 13, 11]. Unfortunately, *computationally effective methods* working on given data (i.e. estimators) are not in the main scope of the aforementioned books. These seem, however, to be a valuable tool in the hands of practitioners. The paper [7] argues against the usage of one-at-a-time sampling strategies which are computationally cheap. Alternatively, let us take a look at variance-based sensitivity indicators. We present a method which works on given data instead of a special design, hence allowing the sample strategies to range from simple random sampling over Latin hypercube sampling to quasi Monte Carlo sampling. A sensitivity analysis might then be performed on data acquired for an uncertainty analysis or even on measured values. For such a sensitivity analysis, we introduce a simple method of estimating the *first order effect* (also called *main effect*, *Sobol' index* or *correlation ratio*, among others) from given data. This algorithm may even be implemented in a spreadsheet software program. Hence for small analytical simulation models ("one line of code") the steps of creating a random input sample, simulating a model and analysing its model output can all be handled within the speadsheet software.

Email address: elmar.plischke@tu-clausthal.de (Elmar Plischke)

2. Setup

Given two random variables X and Y we want to determine

$$\eta^2 = \frac{\mathrm{Var}[\mathbb{E}[Y|X]]}{\mathrm{Var}[Y]}, \tag{1}$$

the ratio between the variance of the conditional expectation of Y given X and the unconditional variance of Y. This quotient ranges between 0 and 1 and shows the degree of functional dependence of Y on X (or the degree of functional influence of X on Y). In a way, it takes over the role of the squared correlation coefficient $\varrho^2(Y,X)$ for a non-linear regression model. And indeed, $\eta^2 = \varrho^2(Y,\mathbb{E}[Y|X])$ [2]. The term $\mathbb{E}[Y|X]$ is called the nonparametric regression curve. There are many ways of determing a suitable estimate [15].

Thinking in realisations of (X,Y), an estimate of this nonparametric regression curve might be adequately described by the *backbone* of the scatterplot of xs vs. ys. Hence (1) computes the gain in the variance when each point in the scatterplot is replaced by a local mean value. With that in mind, η^2 will be close to one, if $\mathbb{E}[Y|X]$ is a

good approximation of the data, i.e. there are only subtle differences between the regression curve and the data. The less structure there is in the scatterplot, the more η^2 tends to zero.

For an analysis of model output, one considers a set of random variables X^i, $i = 1, \ldots, k$, of known probability distributions, named *input* parameters, and a random variable Y, the *output*, which is the result from a complex simulation model $Y = f(X^1, \ldots, X^k)$. Most of the available algorithms use a designed sample of $(X^1, \ldots, X^k)$ combined with a model evaluation in the loop to estimate η^2 for each of the parameters, see [16, 14, 8] for recent results. We are interested in the influence or sensitivity of the single input parameters on the output. As we use a post-processing method we neither need assumptions about the distributions of the input parameters nor access to the simulation model. We only assume that the input data is a representative sample of the underlying distribution and therefore, together with the corresponding output data, they may be used for estimating (1).

3. Implementation

The algorithm to estimate (1) from given pairs of data $\{(x_i, y_i), i = 1, \ldots, n\}$ consists of the following three steps.

1. Sort the output data (y_i) using the input data (x_i) as a key.
2. Compute the first few coefficients (frequencies, if you like) of the just rearranged output using a suitable orthogonal transformation.
3. Form the quotient between the sum of squares of these coefficients and the variance of Y.

The dependency of the output Y on the input X only enters the estimation procedure through reordering the output realisations. This idea of rearranging the output data with respect to sorting the input data was first used in the RBD method [16] and later in the EASI method [5].

For the second step, the key issue hides in the use of a suitable transformation. Methods like (E)FAST [9] and RBD [16] use the Discrete Fourier Transformation (DFT), those based on RS-HDMR [6, 18] use orthogonal polynomials. For our purposes we borrow one from digital image processing [3], the *discrete cosine transformation* (DCT) also used in the JPEG image standard. This transformation has good energy concentration/compaction properties which means that the first few coefficients provide a fair reconstruction of the original signal.

The weights for the jth coefficient of the DCT are given by

$$w_{i,j} = \sqrt{\frac{2}{n}} \cos\left(\frac{\pi(2i-1)j}{2n}\right),\ i = 1, \ldots, n,\ j = 1, \ldots, n-1, \tag{2}$$

and the coefficient itself is then $c_j = \sum_{i=1}^{n} w_{i,j} y_{\psi(i)}$ where $(y_{\psi(i)})$ is the rearranged output sample. Here the permutation ψ is used to sort x increasingly. Note that the DCT can be thought of a special form of a DFT for which the input signal is made symmetrical by adding a mirrored version of the original data. In [5] this data mirroring process is implemented within the sorting precess while here it is already part of the orthogonal transformation.

In the third step we compute the conditional expectation. As we have reordered the data, the input is now increasingly ordered. If there is a functional dependency between input and output, the (rearranged) output signal can be approximated by half a cosine wave (i.e., approximately linear) and its higher harmanics. Hence, the functional dependency is expressed by replacing the coefficients of the higher frequencies by 0, $c = (c_0, c1, \ldots, c_M, 0, \ldots, 0)$ thus keeping only those which are in resonance with the input signal. If one wants to gain access to the underlying harmonic regression meta-model then inverse DCT has to be applied to theses filtered coefficients c_j, $j = 1, \ldots, M$. Also, a visual inpection of the DCT power spectrum $\left|c_j\right|^2$ might show if M is chosen in the right way. Under discussion are also methods for an adaptive selection of the maximum harmonic M, see [17]. For continuous model dependencies, we expect a quadratic decay so that the choice $M = 5$ to 8 is sufficient in most cases.

In the fourth step we compute the variances using Parseval's Theorem. Due to the orthogonality of the transformation we can read it directly off the coefficients as the sum of squares. Hence there is no need for a back-transformation. With some renormalisation we then have $\hat{\eta}^2 = \frac{1}{(n-1)\hat{\mathrm{Var}}[Y]} \sum_{j=1}^{M} c_j^2$ where M is the maximum number of coefficients to consider . Again, instead of computing the variance of Y directly we may use the sum of squared coefficients for it and obtain

$$\hat{\eta}^2 = \frac{\sum_{j=1}^{M} c_j^2}{\sum_{j=1}^{n-1} c_j^2}. \tag{3}$$

Here the missing coefficient c_0 corresponds to the mean of Y. This formula is only effective when used with a fast DCT implementation [1], not when using (2) directly.

4. Examples

In this section we discuss some tests. We start with a simple and well-discussed example and then consider a more realistic setting of a contaminant transport model.

4.1. The Ishigami test function

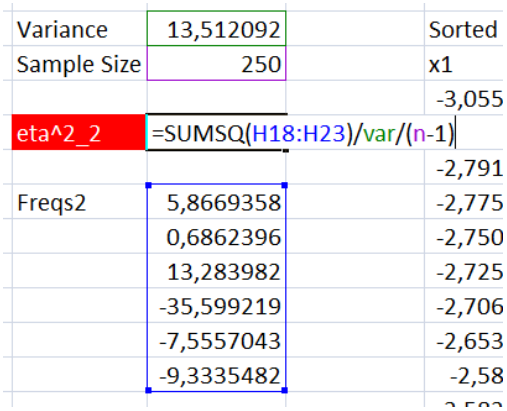

Figure 3: Computation of the first order effects as sum of squares

The Ishigami function is given by

$$Y = \sin X_1 + 7 \sin^2 X_2 + 0.1 X_3^4 \sin X_1$$

where $X_i \sim U(-\pi, \pi)$ are uniformly distributed in $[-\pi, \pi]$. This function is a three-parameter model. We add a fourth dummy input parameter which is not used during the test function evaluation. This function is in so far interesting as the second and third input factors have a Pearson Correlation Coefficient of zero. A variance-based sensitivity analysis retrieves a 44% first order effect for the second input factor, but the third and fourth input factors show no first order effect. While the effect of the third parameter is non-functional, the influence of the fourth parameter is purely random. The values of $R^2 \approx R^{2*} \approx 20\%$ imply that the results from a standard or rank-transformed linear regression are not very powerful in this case.

Figures 1-3 show details on how to perform the calculation from within a spreadsheet. For a sample size of 250, we obtain the estimates $\hat{\eta}_1^2 = 0.29$ (expected 0.31), $\hat{\eta}_2^2 = 0.48$ (expected 0.44), $\hat{\eta}_3^2 = 0.00$ (expected 0.00) and $\hat{\eta}_4^2 = 0.03$ (expected 0.00). For the second parameter, the major contribution can be found in the 3rd and 4th frequencies, see Figure 3, which suggests that the functional dependency is highly non-linear. Note that all of the results are in the range given by the Monte-Carlo sampling error $\frac{1}{\sqrt{n}} \approx 0.06$.

4.2. The Level-E geosphere transport model

In various publications (see [12] for a review), the PSACOIN Level E code [4] is used both as a benchmark of Monte Carlo simulations and as a benchmark for sensitivity analysis methods. This computational model predicts the radiological dose to humans over geological time scales due to the underground migration of radionuclides from a hypothetical nuclear waste disposal site through a system of idealised natural and engineered barriers. The model has a total of 33 parameters, 12 of which are taken as independent uncertain parameters. The uncertainties are either uniformly or log-uniformly distributed. The parameters of the distributions have been selected on the basis of expert judgement.

Luckily, as our proposed method of computation is able to act as a post-precessor, we can dust off a dataset from the electronic shelf and analyse it. An Excel 2007 add-in written in C++ was used that offers the same functionality, but is a more comfortable and compact form of executing the algorithm. The results obtained from a quasi-Monte Carlo sample of size 4096 are shown in Figure 4. This analysis shows that parameters v^1, the velocity in geosphere layer 1, and W, the biosphere stream flow rate, have dominant first order effects on the total dose rate. However, the sum of all first order sensitivity indices is well below 1, indicating that there are interactions in effect. These results are in agreement with [16, Figure 7] and [5, Figure 6].

5. Conclusions

The use of the Discrete Cosine Transform for estimating first order effects offers a refreshing look on variance-based sensitivity indicators. For example, first order effects can be routinely computed accompanying a linear regression. Estimating higher order effects and, in particular, total effects from given data still remains a challenge.

Variance	13,512092		Weights for cosine transform			
Sample Size	250			1	2	3
			1	=COS(PI()*$S6*T$5/(2*n))		
eta^2_1	0,2901714		3	0,9998224	0,9992895	0,9984016
			5	0,9995066	0,9980267	0,995562
Freqs1	-28,470303		7	0,9990329	0,9961336	0,9913076
	-2,3371886		9	0,9984016	0,9936113	0,9856446
	10,986826		11	0,9976125	0,9904614	0,9785809
	2,86638		13	0,9966659	0,9866859	0,9701266
	5,5359294		15	0,995562	0,9822873	0,9602937
	0,8313084		17	0,9943008	0,9772681	0,9490961

Figure 1: Computation of the DCT weights by cosine evaluation

Variance	13,512092		Sorted values (y on key x)		Weights for cosine transform		
Sample Size	250		x1	y		1	
			-3,0559822	2,0082658	1	0,9999803	0,999
eta^2_1	0,2901714		-3,0080999	0,3959592	3	0,9998224	0,9992
			-3,000707	2,5175552	5	0,9995066	0,998(
Freqs1	=SQRT(2/n)*SUMPRODUCT($K$6:$K$255;T$6:T$255)				7	0,9990329	0,9961
	-2,3371886		-2,927713	6,7081583	9	0,9984016	0,9936
	10,986826		-2,8963091	6,7483784	11	0,9976125	0,9904
	2,86638		-2,8836269	1,2574168	13	0,9966659	0,9866
	5,5359294		-2,8611234	1,3419504	15	0,995562	0,9822
	0,8313084		-2,8420763	3,9967747	17	0,9943008	0,9772

Figure 2: Computation of the DCT coefficients as inner product of reordered outputs and weights

References

[1] M. Frigo and S. G. Johnson. The design and implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231, 2005.

[2] M. Fréchet. Sur le coefficient, dit de corrélation et sur la corrélation en géneral. *Revue de l'Institut International de Statistique*, 1(4):16–23, 1934.

[3] B. Jähne. *Digitale Bildverarbeitung (Digital Image Processing)*. Springer Verlag, Heidelberg, 7th edition, 2011.

[4] Nuclear Energy Agency. PSACOIN level E intercomparison. Technical report, OECD, Paris, 1989.

[5] E. Plischke. An effective algorithm for computing global sensitivity indices (EASI). *Reliab. Eng. Syst. Saf.*, 95(4):354–360, 2010.

[6] H. Rabitz and Ö. F. Alış. General foundations of high-dimensional model representations. *J. Math. Chem.*, 25(2–3):197–233, 1999.

[7] A. Saltelli and P. Annoni. How to avoid a perfunctory sensitivity analysis. *Environmental Modelling&Software*, 25:1508–1517, 2010.

[8] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola. Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index. *Comput. Phys. Commun.*, 181(2):259–270, 2010.

[9] A. Saltelli and R. Bolado. An alternative way to compute Fourier amplitude sensitivity test (FAST). *Computational Statistics&Data Analysis*, 26:445–460, 1998.

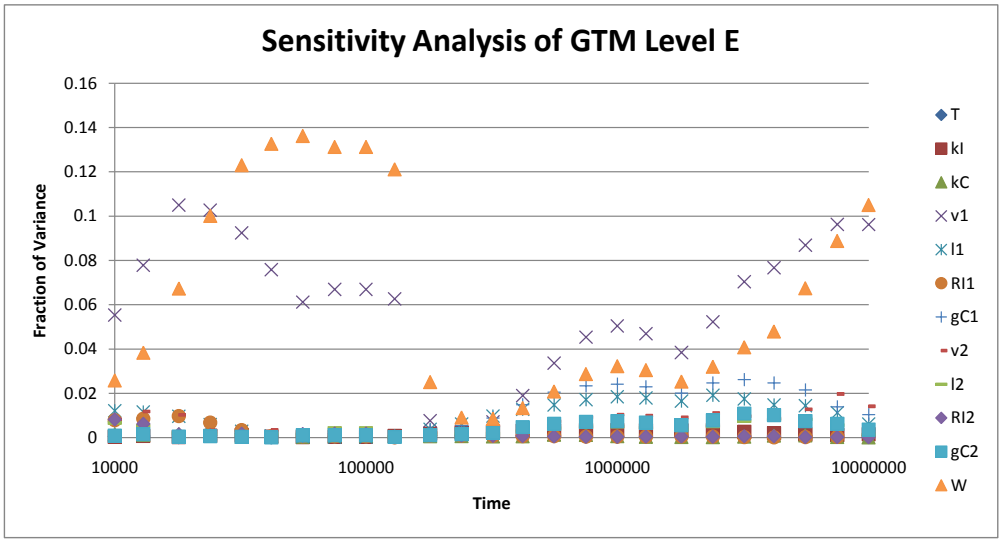

Figure 4: Time-dependent first order effects for the Level E transport model

[10] A. Saltelli, K. Chan, and E. Scott. *Sensitivity Analysis*. John Wiley&Sons, Chichester, 2000.

[11] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola. *Global Sensitivity Analysis – The Primer*. John Wiley&Sons, Chichester, 2008.

[12] A. Saltelli and S. Tarantola. On the relative importance of input factors in mathematical models: Safety assessment for nuclear waste disposal. *J. Amer. Statist. Assoc.*, 97(459):702–709, 2002.

[13] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto. *Sensitivity Analysis in Practise – A Guide to Assessing Scientific Models*. John Wiley&Sons, Chichester, 2004.

[14] I. Sobol´, S. Tarantola, D. Gatelli, S. Kucherenko, and W. Mauntz. Estimating the approximation error when fixing unessential factors in global sensitivity analysis. *Reliab. Eng. Syst. Saf.*, 92:957–960, 2007.

[15] K. Takezawa. *Introduction to Nonparametric Regression*. John Wiley&Sons, Hoboken, NJ, 2006.

[16] S. Tarantola, D. Gatelli, and T. Mara. Random balance designs for the estimation of first order global sensitivity indices. *Reliab. Eng. Syst. Saf.*, 91:717–727, 2006.

[17] S. Tarantola and M. Koda. Improving random balance designs for the estimation of first order sensitivity indices. *Procedia – Social and Behavioral Sciences*, 2(6):7753–1754, 2010.

[18] T. Ziehn and A. S. Tomlin. GUI-HDMR - a software tool for global sensitivity analysis of complex models. *Environmental Modelling&Software*, 24:775–785, 2009.