ESTIMATION OF GLOBAL SENSITIVITY INDICES FOR MODELS WITH CORRELATED INPUTS TAKING INTO ACCOUNT MAPPING OF CORRELATION COEFFICIENTS BETWEEN NORMAL AND ARBITRARILY DISTRIBUTED RANDOM VECTORS

S. Kucherenko, A. Klimenko
Imperial College London, London, SW7 2AZ, UK
e-mail: s.kucherenko@imperial.ac.uk

1. Theory

The method employed here follows the Nataf model [1, 2]. The Nataf transformation is a way to model the dependence structure of a random vector by a normal copula, parameterized by its correlation matrix [3].

Consider arbitrarily distributed variables $(X_1, X_2, \dots, X_n)$ with correlation matrix $\Sigma_X = [r_{ij}]$ and the corresponding standard normal variables $(Y_1, Y_2, \dots, Y_n)$ with correlation matrix $\Sigma_Y = [\rho_{ij}]$ obtained by marginal transformations

$$Y_i = \Phi^{-1}(F_i(X_i)), \tag{1}$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function (CDF), $\Phi^{-1}(\cdot)$ is its inverse, and $F_i(\cdot)$ is the CDF of the corresponding arbitrarily distributed variable.

The link between the correlation coefficient, ρ_{ij}, of the normal variables and the corresponding coefficient, r_{ij}, of arbitrarily distributed ones is provided by the following integral relation [2, 4]:

$$r_{ij} = \frac{1}{\sigma_i \sigma_j}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_i^{-1}(\Phi(y_1))F_j^{-1}(\Phi(y_2))\frac{\exp\left(-\frac{y_1^2-2\rho_{ij}y_1y_2+y_2^2}{2\left(1-\rho_{ij}^2\right)}\right)}{2\pi\sqrt{1-\rho_{ij}^2}}dy_1dy_2 - \mu_i\mu_j\right], \tag{2}$$

where μ and σ denote the mean and standard deviation of each of the arbitrarily distributed variables.

Since the double integral in (2) is in general evaluated numerically rescaling of variables is applied to map the two-dimensional elliptical Gaussian probability density function (PDF) onto a circular one to improve accuracy of two-dimensional quadrature. This is done by the substitution $y_1 = z_1\sqrt{1-\rho_{ij}^2} + z_2\rho_{ij},\ y_2 = z_2$, so that the final relation used in computations is

$$r_{ij} = \frac{1}{\sigma_i \sigma_j}\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F_i^{-1}\left(\Phi\left(z_1\sqrt{1-\rho_{ij}^2} + z_2\rho_{ij}\right)\right)F_j^{-1}(\Phi(z_2))\frac{1}{2\pi}e^{-\frac{z_1^2+z_2^2}{2}}dz_1dz_2 - \mu_i\mu_j\right] \tag{3}$$

In order to obtain the inverse mapping $\rho_{ij}(r_{ij})$, equation (3) must be solved (iteratively) with respect to ρ_{ij} given a value of r_{ij}. This equation has a unique solution since r_{ij} monotonically depends on ρ_{ij} and vice versa [2].

2. Matlab implementation

The report contains three directories:
GSA_CORRELATED_MIXED_DISTRIBUTIONS
CORRELATIONS_TRANSFORMATION_TEST
CORRELATIONS_MATRIX_TEST

2.1. Directory GSA_CORRELATED_MIXED_DISTRIBUTIONS

It contains the following MATLAB files: `SOBOL_correlatedBD.m`, `Tranformation_required_distribution.m`, `ishigami.m`, `linearsum.m`, `LPTAU51.m`, `main.m`.

SOBOL_correlatedBD.m is a realization of

```
function [S ST] = SOBOL_correlatedBD(k,N,typesampling,typedistr,mu,sigma,COR,model)
```

which computes main effect and total sensitivity indices for for models with correlated inputs taking into account mapping of correlation coefficients between normal and arbitrarily distributed random vectors. The theory is presented in [4]. Formulas used in the function follow notations of [4] with relevant references.

The code supports six typical probability distributions of input factors: uniform, triangular, exponential, normal, loguniform and lognormal (see next section for the detailed description of the distributions and their parameters), which are denoted in the code with the following strings:

```
'UNIF' 'TRIANG' 'NORM' 'LOGUNIF' 'LOGNORM' 'EXPO'
```
(4)

The mapping from the correlation coefficient, ρ_{ij}, of two standard normal variables to the corresponding correlation coefficient, r_{ij}, of arbitrarily distributed variables according to equation (3) is performed by the following function:

```
function r = rho_to_r( typedistr1, typedistr2, mu1, sigma1, mu2, sigma2, rho )
```

where `typedistr1` and `typedistr2` are the distribution types of the two variables (i.e., strings from the above list (4) of supported distribution types), `mu1` and `mu2` their mean values, and `sigma1` and `sigma2` are the standard deviations (*see next section for the exceptions in the meaning of values passed*).

The inverse mapping of r_{ij} of arbitrarily distributed variables to ρ_{ij} of standard normal ones is provided by (the meanings of parameters passed are the same as for function `rho_to_r`)

```
function rho = r_to_rho( typedistr1, typedistr2, mu1, sigma1, mu2, sigma2, r )
```

Transformation of correlation matrix of a vector of standard normal variables into the corresponding correlation matrix of arbitrarily distributed variables is performed by

```
function COR_ARB = cor_matr_N_to_Arb(COR,typedistr,mu,sigma)
```

where `COR` is the correlation matrix of a vector of standard normal variables, `COR_ARB` the correlation matrix of variables of types defined by the string array `typedistr` (with entries from list (4)) with means and standard deviations defined by vectors `mu` and `sigma` (*see next section for the exceptions in the meaning of values passed*).

The following function provides the inverse transform from the correlation matrix of arbitrarily distributed variables (`COR`) to the corresponding one of standard normal variables (`COR`):

```
function COR_N = cor_matr_Arb_to_N(COR,typedistr,mu,sigma)
```

The meanings of other parameters are the same as for function `cor_matr_N_to_Arb`. Note that although correlation coefficients of normal variables can range from -1 to 1 this is not the case for a pair of differently distributed variables. Function `cor_matr_Arb_to_N` checks whether each entry falls in the range $r_{ij}^{min} \le r_{ij} \le r_{ij}^{max}$ which depends on distribution types of X_i and X_j as well as on the values of their means and standard deviations. In case r_{ij} is out of the attainable range the closest lower or upper bound is used instead and a warning message is displayed.

The values of r_{ij}^{min} and r_{ij}^{max} are determined from the integral relation (3) upon substituting -1 and 1 in place of the correlation coefficient, ρ_{ij}, of the corresponding standard normal variables.

The performance of these functions is demonstrated by the scripts `main.m`. The ishigami test function is used with three different types of input distributions:

```
typedistr={ 'UNIF'  'NORM'  'LOGNORM'}
```

and the following correlation matrix:

$$\mathtt{COR} = \begin{matrix} 1 & 0 & 0.7 \\ 0 & 1 & 0 \\ 0.7 & 0 & 1 \end{matrix}.$$

This matrix was transformed to a corresponding correlation matrix of normal inputs:

$$\mathtt{COR\_N} = \begin{matrix} 1 & 0 & 0.7 \\ 0 & 1 & 0 \\ 0.7 & 0 & 1 \end{matrix}.$$

Dependence of S_i and ST_i for i = 1,2,3 versus the number of sampled points N is presented in Fig.1.

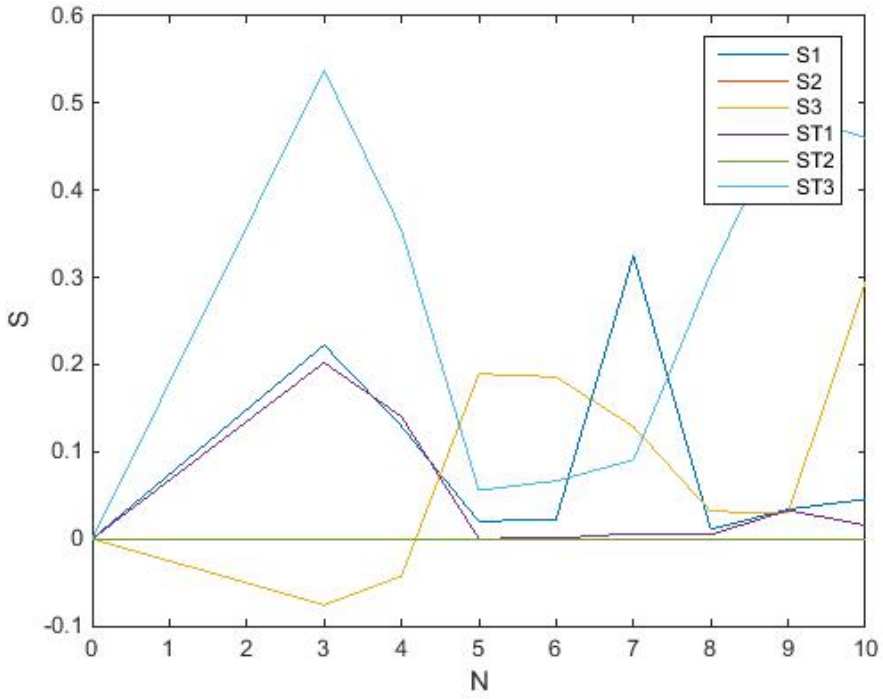

Fig. 1. Dependence of S_i and ST_i for i = 1,2,3 versus the number of sampled points N.

2.2. CORRELATIONS_TRANSFORMATION_TEST

It contains the following MATLAB files: `correlation_test.m`, `rho_to_r.m`, `r_to_rho.m`.

The performance of these functions is demonstrated by the scripts `correlation_test.m`. The script applies the integral relation (3) to pairs of identically distributed variables and then to pairs of differently distributed variables for the whole range $[-1, 1]$ of values of the corresponding correlation coefficient of standard normal variables. The results are plotted as deviations from the identical linear dependence in each case (Fig. 2). The plot shows that pairwise correlations of differently distributed variables (and in some cases of those identically distributed as shown for the case of two loguniform variables) do not necessarily reach the values -1 and 1. It is also clear that dependence r (r_{ij}) versus rho (ρ_{ij}) in many cases is highly nonlinear.

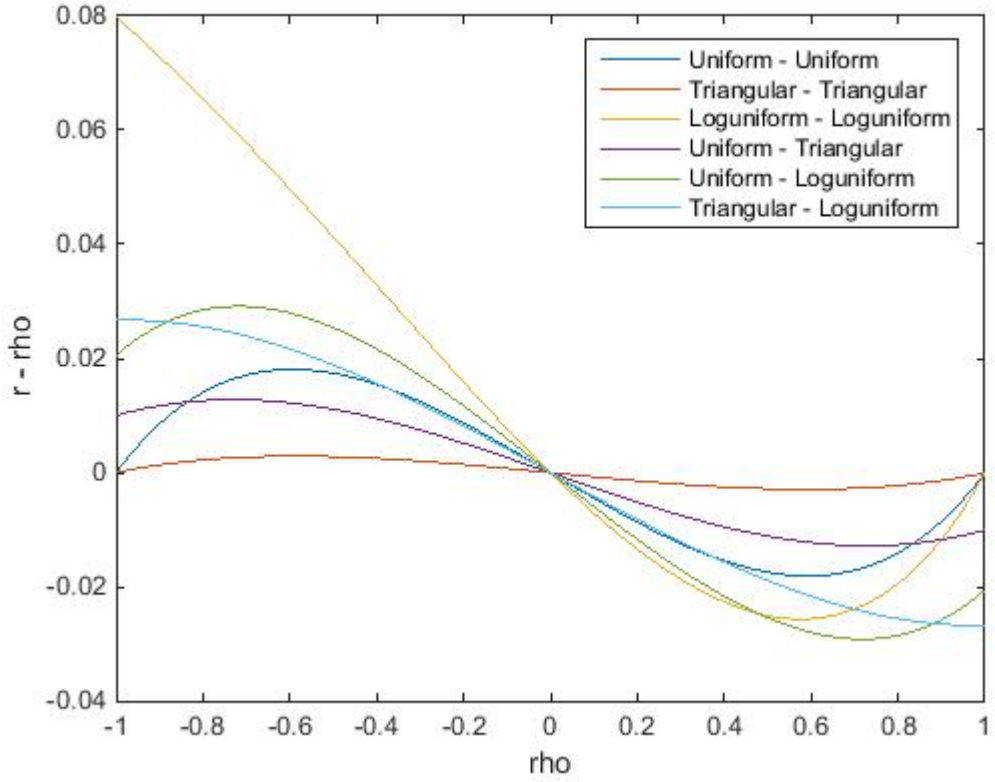

Fig. 2 Deviations from the identical linear dependence r-rho versus the corresponding correlation coefficient rho (ρ_{ij}) of standard normal variables. Correlation coefficient r is computed for 6 pairs of differently distributed variables.

2.3. CORRELATIONS_MATRIX_TEST

It contains the following MATLAB files: `cor_matrix_test.m`, `cor_matr_Arb_to_N.m`, `cor_matr_N_to_Arb.m`, `rho_to_r.m`, `r_to_rho.m`

The script `cor_matrix_test.m` transforms correlation matrix COR of six variables of distinct types (given in (4)) into the matrix COR_N for standard normal variables and then performs the inverse mapping to show that the result corresponds to the original matrix COR. Secondly, correlation matrix COR1 of three variables with distinct distributions and all off-diagonal correlation coefficients lying outside their attainable ranges is transformed into matrix COR_N1 for standard normal variables to demonstrate restriction of correlation by function `cor_matr_Arb_to_N`. Warning messages are displayed and unfeasible values are replaced with closest feasible ones. Note, that the inverse transform by `cor_matr_N_to_Arb` does not give the original matrix COR1. Instead, all its entries are restricted to their permissible ranges.

`cor_matrix_test.m` performs two tests:

Test N1 performs inverse transform to verify that this direct and inverse transform gives the identity mapping

Test N2 performs bracketing of feasible correlation values. Correlation matrix whose off-diagonal entries are deliberately chosen to lie outside the respective feasibility bounds

3. Supported probability distributions

This section details the definitions of probability density functions (PDF), means, standard deviations, cumulative distribution functions (CDF), and inverse cumulative distribution functions for the six distributions listed in (4) as implemented in the code.

3.1. Uniform distribution

The general uniform distribution has the probability density function of the form:

$$p(x) = \begin{cases} \dfrac{1}{b-a}, & a \le x \le b \\ 0, & x < a \text{ or } x > b \end{cases} \tag{5}$$

The mean and standard deviation are given by

$$\mu = \frac{a+b}{2}, \quad \sigma = \frac{b-a}{2\sqrt{3}} \tag{6}$$

The corresponding CDF and inverse CDF are readily obtained as

$$F(x) = \begin{cases} 0, & x < a \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ 1, & x > b \end{cases} \tag{7}$$

and

$$F^{-1}(p) = a + (b-a)p, \qquad 0 \le p \le 1 \tag{8}$$

Note, that in the case of a uniform distribution it is more natural to work with parameters a and b (as the lower and upper bounds of a parameter variation) rather than with the mean and standard deviation. Hence the user is required to supply the values of a and b in the arrays 'mu' and 'sigma'.

3.2. Triangular distribution

We consider here only the symmetrical case corresponding to the following PDF taking nonzero values in the interval $[a, b]$:

$$p(x) = \begin{cases} \dfrac{4(x-a)}{(b-a)^2}, & a \le x \le \dfrac{a+b}{2} \\ \dfrac{4(b-x)}{(b-a)^2}, & \dfrac{a+b}{2} < x \le b \\ 0, & x < a \text{ or } x > b \end{cases} \tag{9}$$

The mean and standard deviation are given by

$$\mu = \frac{a+b}{2}, \quad \sigma = \frac{b-a}{2\sqrt{6}} \tag{10}$$

The corresponding CDF and inverse CDF are

$$F(x) = \begin{cases} 0, & x < a \\ 2\left(\frac{x-a}{b-a}\right)^2, & a \le x \le \frac{a+b}{2} \\ 1 - 2\left(\frac{b-x}{b-a}\right)^2, & \frac{a+b}{2} < x \le b \\ 1, & x > b \end{cases} \tag{11}$$

and

$$F^{-1}(p) = \begin{cases} a + (b-a)\sqrt{p/2}, & 0 \le p \le 1/2 \\ b - (b-a)\sqrt{(1-p)/2}, & \frac{1}{2} < p \le 1 \end{cases} \tag{12}$$

Note, that as for the uniform distribution, parameters a and b describe the distribution in a more natural way than the mean and standard deviation. Thus the values of a and b should be passed as entries of the 'mu' and 'sigma' arrays.

3.3. Exponential distribution

The exponential distribution is defined in the classical way:

$$p(x) = \lambda \exp(-\lambda x), \quad x \ge 0 \tag{13}$$

$$\mu = \sigma = 1/\lambda \tag{14}$$

The CDF and its inverse are

$$F(x) = 1 - \exp(-\lambda x), \quad x \ge 0 \tag{15}$$

$$F^{-1}(p) = -\frac{\ln(1-p)}{\lambda}, \qquad 0 \le p \le 1 \tag{16}$$

These functions are supplied in Matlab as expcdf(x,mu) and expinv(p, mu), respectively.

Note, that only one parameter is required for the exponential distribution, which is the mean value that should be passed as an entry of the 'mu' array. The content of the corresponding entry of the 'sigma' array is not used in the calculations and can contain any value.

3.4. Normal distribution

The normal PDF is

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \qquad -\infty \le x \le \infty \tag{17}$$

The CDF and inverse CDF are given by

$$F(x) = \Phi\left(\frac{x-\mu}{\sigma}\right) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{x-\mu}{\sigma\sqrt{2}}\right)\right], \qquad -\infty \le x \le \infty \tag{18}$$

$$F^{-1}(p) = \mu + \sigma\sqrt{2}\,\operatorname{erf}^{-1}(2p-1), \qquad 0 \le p \le 1 \tag{19}$$

Note, that the normal CDF and its inverse are provided by the built-in Matlab functions normcdf(x, mu, sigma) and norminv(p, mu, sigma). The user is required to supply the values of μ and σ in the corresponding arrays.

3.5. Loguniform distribution

This distribution describes the random variable $Y = \ln(X)$ where X is uniformly distributed on $[a, b]$, $a > 0$, and has the following PDF:

$$p(x) = \frac{1}{x(\ln b - \ln a)}, \qquad a \le x \le b \tag{20}$$

with the mean and standard deviation defined as

$$\mu = \frac{b-a}{\ln b - \ln a} \tag{21}$$

$$\sigma^2 = \frac{b^2 - a^2}{2(\ln b - \ln a)} - \frac{(b-a)^2}{(\ln b - \ln a)^2} \tag{22}$$

The loguniform CDF and its inverse are

$$F(x) = \begin{cases} 0, & x < a \\ \dfrac{\ln x - \ln a}{\ln b - \ln a}, & a \le x \le b \\ 1, & x > b \end{cases} \tag{23}$$

$$F^{-1}(p) = a\left(\frac{b}{a}\right)^p, \qquad 0 \le p \le 1 \tag{24}$$

Note, that analogously to the uniform and triangular distributions, parameters a and b should be passed as 'mu' and 'sigma' instead of the mean and standard deviation values.

3.6. Lognormal distribution

The PDF of the lognormal distribution is

$$p(x) = \frac{1}{x\,\xi\sqrt{2\pi}} \exp\left(-\frac{(\ln(x) - \lambda)^2}{2\xi^2}\right), \quad x > 0 \tag{25}$$

with parameters ξ and λ related to the mean and standard deviation in the following way:

$$\xi = \sqrt{\ln\left(1 + \frac{\sigma^2}{\mu^2}\right)}, \quad \lambda = \ln\frac{\mu^2}{\sqrt{\mu^2 + \sigma^2}} \tag{26}$$

The CDF of this distribution is

$$F(x) = \Phi\left(\frac{\ln x - \lambda}{\xi}\right) \tag{27}$$

It is provided by the built-in Matlab function logncdf (x, lambda, ksi) while its inverse is given by function logninv(p, lambda, ksi). **Note,** that values of μ and σ (not λ and ξ) must be passed as 'mu' and 'sigma'.

References

1. A. Nataf. Determination des distributions dont les marges sont donnees. *C. R. Acad. Sci. Paris* (1962) 225, 42-43.
2. P.-L. Liu, A. Der Kiureghian. Multivariate distribution models with prescribed marginal and covariances. *Probabilist. Eng. Mech*. 1 (1986) 105-112.
3. Lebrun, R. & Dutfoy, A. An innovating analysis of the Nataf transformation from the copula viewpoint. *Prob. Eng. Mech*., (2009) 24, 312-320.
4. S. Kucherenko, S. Tarantola, P. Annoni. Estimation of global sensitivity indices for models with dependent variables. *Comput. Phys. Commun*. 183 (2012) 937-946.