Lecture 4.2b Variance-based sensitivity analysis for models with correlated inputs

Gürkan Sin, Associate Professor
PROSYS- DTU Chemical Engineering

DTU Kemiteknik
Institut for Kemiteknik

Objective of this lecture

- At the end of the lecture, you should be able to:
  - Perform sensitivity analysis using Variance-based sensitivity method for models with dependend/correlated inputs
  - Understand and use two methods for performing variance-based sensitivity indices for correlated inputs: random sampling with bins (scatter plot smoothing) and conditional variance based estimation of Sobol indices

Outline

- Conditional Variance-based sensitivity measure
  - formulas
  - Monte-Carlo estimates
- Computing sensitivity indices
  - Workflow
  - Sampling from conditional probability distribution
- Analytical example: bivariate normal
- Numerical examples
  - Linear additive model
  - Nonlinear model
- exercise

Conditional variance based-sensitivity measure

Consider a model function *f (x*1*, . . . , xn)* defined in *Rn* with

an input vector *x* = *(x*1*, . . . , xn)*.

Here *x* is a real-valued random variable with a continuous distribution function *p(x*1*, . . . , xn)*. Consider an arbitrary subset of the variables *y* = *(xi*1 *, . . . , xi*$_s$ *)*, 1 *s* < *n*, and a complementary subset *z* = *(xi*$_1$ *, . . . , xi*$_{n-s}$ *)*, so that *x* = *(y, z)*.

The total variance of *f (x*1*, . . . , xn)* can be decomposed as

$$D = D_y\left[E_z\left(f(y,\bar{z})\right)\right] + E_y\left[D_z\left(f(y,\bar{z})\right)\right]$$

$$E_z\left(f(y,\bar{z})\right) = \int f(y,\bar{z})p(y,\bar{z}|y)d\bar{z}$$

$$D_y\left[E_z\left(f(y,\bar{z})\right)\right] = \int \left[E_z\left(f(y,\bar{z})\right)\right]^2 p(y)dy - fo^2$$

$$E_y\left[D_z\left(f(y,\bar{z})\right)\right] = \int \left[D_z\left(f(y,\bar{z})\right)\right]^2 p(y)dy$$

Definition of sensitivity measures for correlated inputs

Notations *z* and $\bar{z}$ to distinguish a random vector *z* generated from a joint probability density function *p(y, z)* and a random vector $\bar{z}$ generated from a conditional distribution $p(y, \bar{z}|y)$. Normalized by the total variance, this expression leads to the equality to calculate sensitivity indices:

$$1 = \frac{D_y\left[E_z\left(f(y,\bar{z})\right)\right]}{D} + \frac{E_y\left[D_z\left(f(y,\bar{z})\right)\right]}{D}$$

$$S_y = \frac{D_y\left[E_z\left(f(y,\bar{z})\right)\right]}{D}$$

$$S_z^T = \frac{E_y\left[D_z\left(f(y,\bar{z})\right)\right]}{D}$$

$$S_y^T = \frac{E_z\left[D_y\left(f(\bar{y},z)\right)\right]}{D}$$

Full expression:

$$S_y = \frac{1}{D}\left[\int_{Rs} p(y)dy\left[\int_{Rn-s} f(y,\bar{z})p(y,\bar{z}|y)d\bar{z}\right]^2 - f_0^2\right]$$

S_y and S_y^T are the first and total effect for subset y

Multivariate normal distribution

- Computation of the indices for correlated inputs require sampling strategies from conditional distributions.
- Let us explore multivariate normal distribution of n-dimensional random vector with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$:

$$f_{\boldsymbol{x}}(x_1, x_2 \dots x_n) = \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma}|}} exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

- The components *y*, *z* of the vector *x* are also normally distributed

with mean vectors μy , μz and covariance matrices Σy , Σz correspondingly.

$$\boldsymbol{\mu} = \begin{bmatrix} \mu_{\boldsymbol{y}} \\ \mu_{\boldsymbol{z}} \end{bmatrix} \;\&\; \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma_y} & \boldsymbol{\Sigma_{yz}} \\ \boldsymbol{\Sigma_{zy}} & \boldsymbol{\Sigma_z} \end{bmatrix}$$

Conditional distribution of multivariate normals

- The conditional distribution of $p(y, \bar{z}|y)$ $is\ a\ normal\ distribution$:

$$p(y, \bar{z}|y) = \frac{1}{\sqrt{(2\pi)^{n-s}|\Sigma_{zc}|}} exp\left(-\frac{1}{2}(\bar{z} - \mu_{zc})^T {\Sigma_{zc}}^{-1}(\bar{z} - \mu_{zc})\right)$$

where $n - s\ is\ lenght\ of\ vector\ \bar{z}c\ and\ \mu_{zc}$ vector is:

$$\mu_{zc} = \mu_z + \Sigma_{yz}{\Sigma_z}^{-1}(y - \mu_y)$$

where Σ_{zc} vector is:

$$\Sigma_{zc} = \Sigma_z - \Sigma_{zy}{\Sigma_z}^{-1}\Sigma_{yz}$$

Simple case: bivariate normal distribution

- The conditional distribution of $p(y,z)$ *is a normal distribution*:

$$p(y,z) = \frac{1}{2\pi\sigma_y\sigma_z\sqrt{1-\rho^2}} exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(y-\mu_y)}{\sigma_{y^2}} + \frac{(z-\mu_z)}{\sigma_{z^2}} - \frac{2\rho(y-\mu_y)(z-\mu_z)}{\sigma_y\sigma_z}\right]\right)$$

- where ρ is the correlation coefficient between *y* and *z*. Here *y*, *z* are two elements of the input vector *x*.

The conditional distribution of z on y , $p(y,\bar{z}|y)$ simplifies to:

$$p(y,\bar{z}|y) = \frac{1}{\sigma_z\sqrt{2\pi(1-\rho^2)}} exp\left(-\frac{1}{2(1-\rho^2)}\left[\frac{(z-\mu_{zc})^2}{\sigma_{z^2}}\right]\right)$$

With conditional mean: $\mu_{zc} = \mu_z + \rho\frac{\sigma_z}{\sigma_y}(y-\mu_y)$

Analytical example: *Bivariate normal distribution. Linear additive model*

- Consider the following linear additive model:

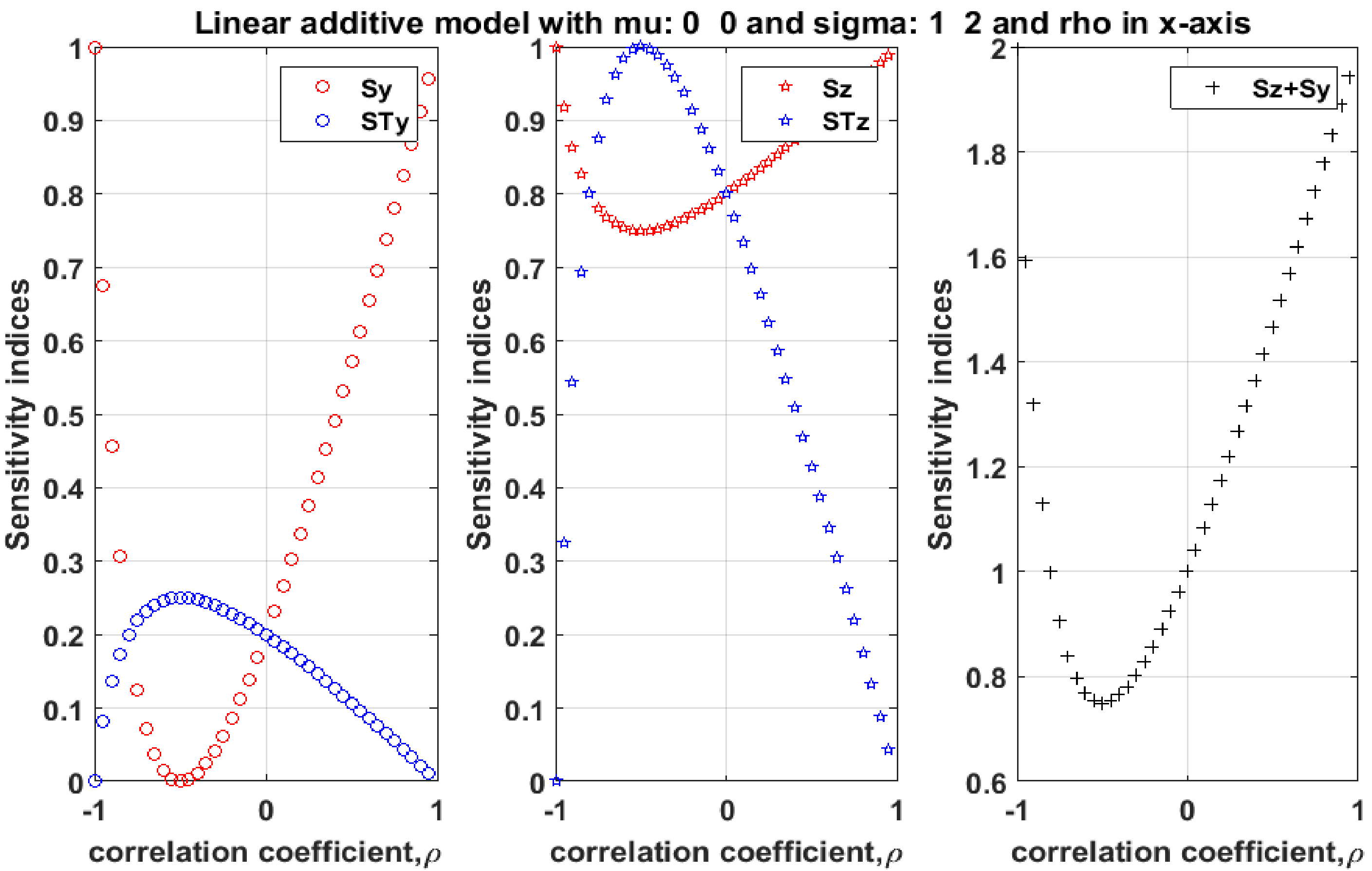
$$f(y,z) = a_1 y + a_2 z$$

Let us take a1=a2=1.

Since it is a linear additive model, we can do the calculation straightforward D=cov(y,z):

$$D = var(y+z) = \sigma_y{}^2 + \sigma_z{}^2 + 2\rho\sigma_y\sigma_z$$
$$D_y = \left(\sigma_y + \rho\sigma_z\right)^2 = \sigma_y{}^2 + \rho^2\sigma_z{}^2 + 2\rho\sigma_y\sigma_z$$
$$D_z = \left(\rho\sigma_y + \sigma_z\right)^2 = \rho^2\sigma_y{}^2 + \sigma_z{}^2 + 2\rho\sigma_y\sigma_z$$
$$S_y = \frac{D_y}{D}; S_z = \frac{D_z}{D};\ S_y{}^T = 1 - \frac{D_z}{D};\ S_z{}^T = 1 - \frac{D_y}{D}$$

Effect of correlation degree on Sensitivity indices

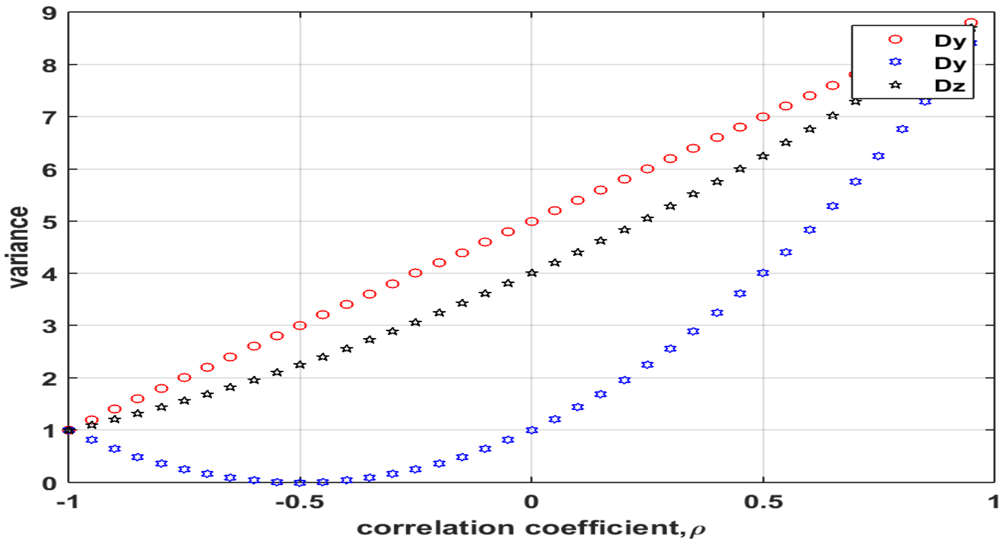


ample.m

Observations: (1) Si and STi affected as a function of rho. 2: STi not always larger than Si. (3) Sum of Si is no longer 1.

Observations

- When inputs are dependent/correlated the following observed:
  - Si and STi affected as a function of ρ (degree of correlation)
  - STi not always larger than Si
  - Sum of Si is no longer 1
- All these opposite to the features of GSA method derived for independent inputs!
- Why? Total variance depends on the degree of correlation, while in the case of indepedent inputs it is constant (just depends on the variance of inputs).



To sum up

- One can not use GSA methods developed for independent inputs when you have in your model inputs (parameters) that are correlated.

- Instead you need to use tailored methods which are developed for models with dependent/correlated inputs.

- Question: which sensitivity measure we shall use for factor importance ranking or factor fixing when inputs are correlated? Si or STi? Look at the variances!
  - E.g. in linear additive model case, when rho = -1, STi=0 but Si=1. So is this important factor or not?
  - It becomes quite complicated. we need a context then to make a decision. In the above case, it just means that when two factor are highly correlated the model can be reduced to a single factor (So that Si and STi becomes equal).

Numerical calculation of Sensitivity indices

To compute S_i and S_{Ti}, one needs to estimate the conditional variances

$$D_y\left[E_z\left(f\left(y,\bar{z}\right)\right)\right] \& E_y\left[D_z\left(f\left(y,\bar{z}\right)\right)\right]$$

There are two approaches:

1) Analytical (exact) solution (see bivariate example)
2) Numerical solutions: random sampling with bins (works only for Si) & Sampling based approaches.

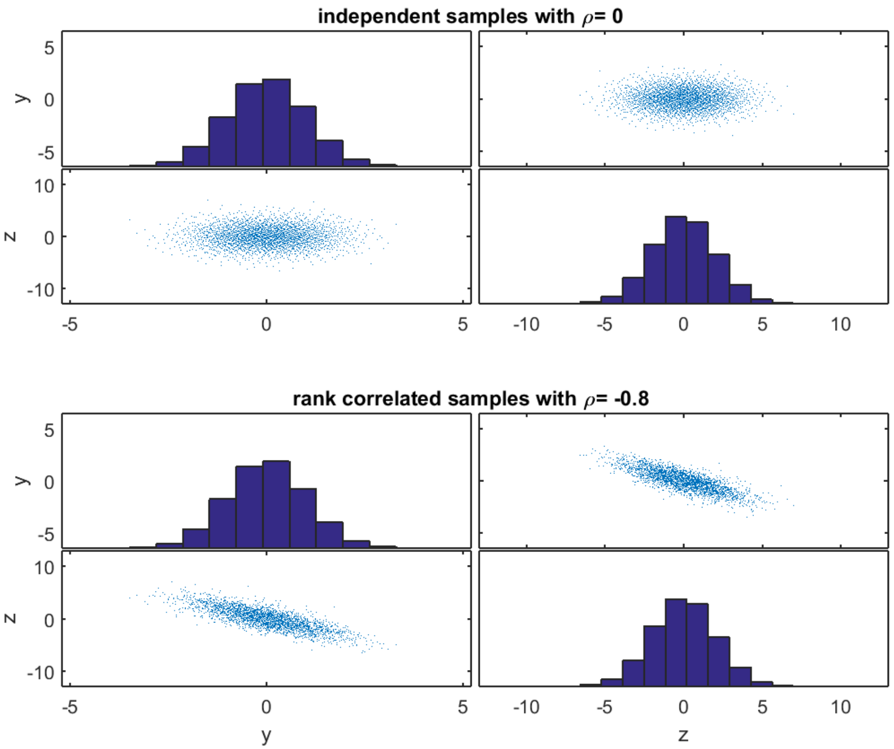
There are other methods under development that uses meta or surrogate modelling and then numerical evaluation or direct integration. We use in this course (monte-carlo) sampling based approach. It takes more model evaluaiton but the most importantly it works.

Approximate method: random sampling & bining

$$f(x) = x_1 + x_2$$
$$x_1 \sim N(0;1)\ ;\ x_2 \sim N(0;2)$$
$$\rho := -0.999 : 0.05 : 0.999$$

$$D_y = \frac{1}{M}\sum_{j=1}^{M}\left(\frac{1}{N_m^j}\sum_{k=1}^{N_m^j} f(y_k, z_k)\right)^2 - f_0^2$$

First generate rank correlated random samples (do quasi-random sampling. Then use gaussian copula to generate desired rank correlation.

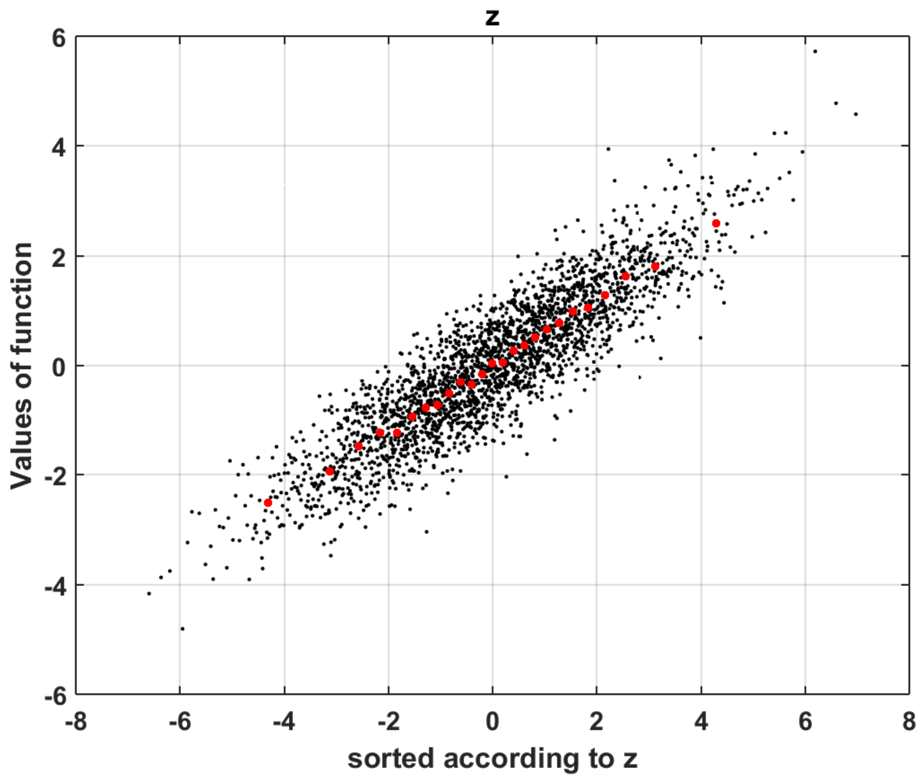


Approximate method: random sampling & bining

$$f(x) = x_1 + x_2$$
$$x_1 \sim N(0;1)\ ;\ x_2 \sim N(0;2)$$
$$\rho := -0.999: 0.05: 0.999$$

$$D_y = \frac{1}{M}\sum_{j=1}^{M}\left(\frac{1}{N_m^j}\sum_{k=1}^{N_m^j} f(y_k, z_k)\right)^2 - f_0^2.$$

Then for each given rho, do MC simulations, sort x, divide by M bins and compute the variance in each bin

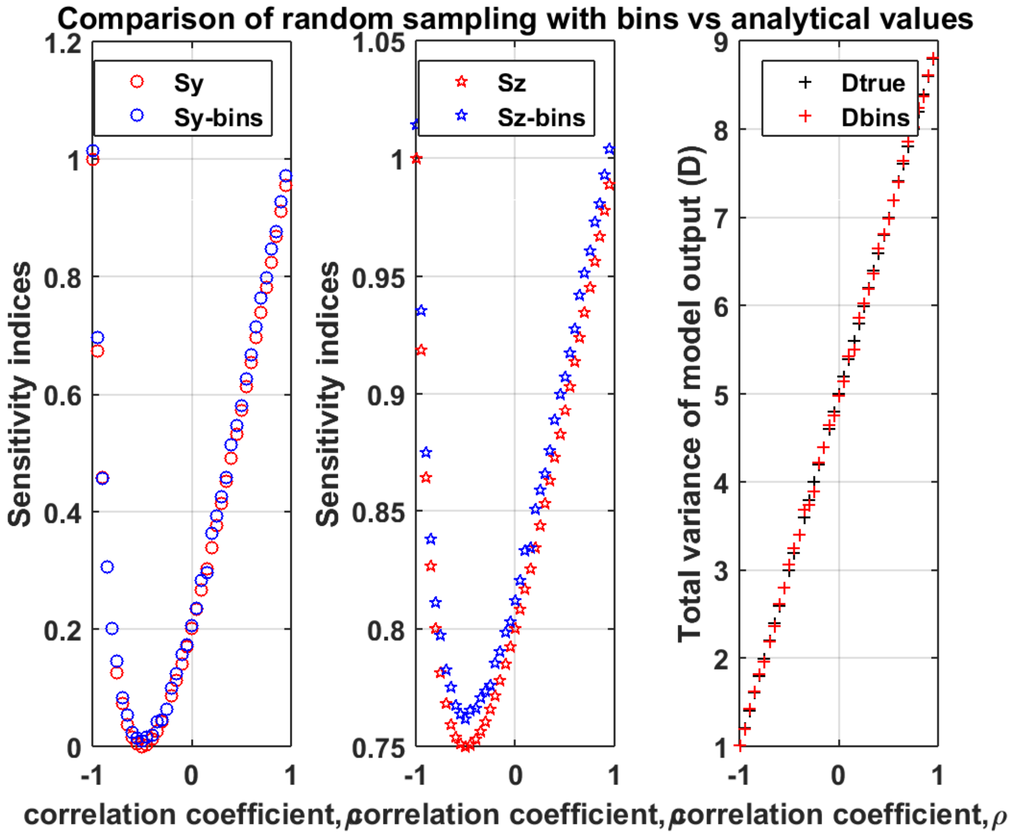


Approximate method: random sampling & bining

$$f(x) = x_1 + x_2$$
$$x_1 \sim N(0;1)\ ;\ x_2 \sim N(0;2)$$
$$\rho \coloneqq -0.999{:}\,0.05{:}\,0.999$$

$$D_y = \frac{1}{M}\sum_{j=1}^{M}\left(\frac{1}{N_m^j}\sum_{k=1}^{N_m^j} f(y_k, z_k)\right)^2 - f_0^2.$$

Finally compare the estimated Si with the analytical methods:

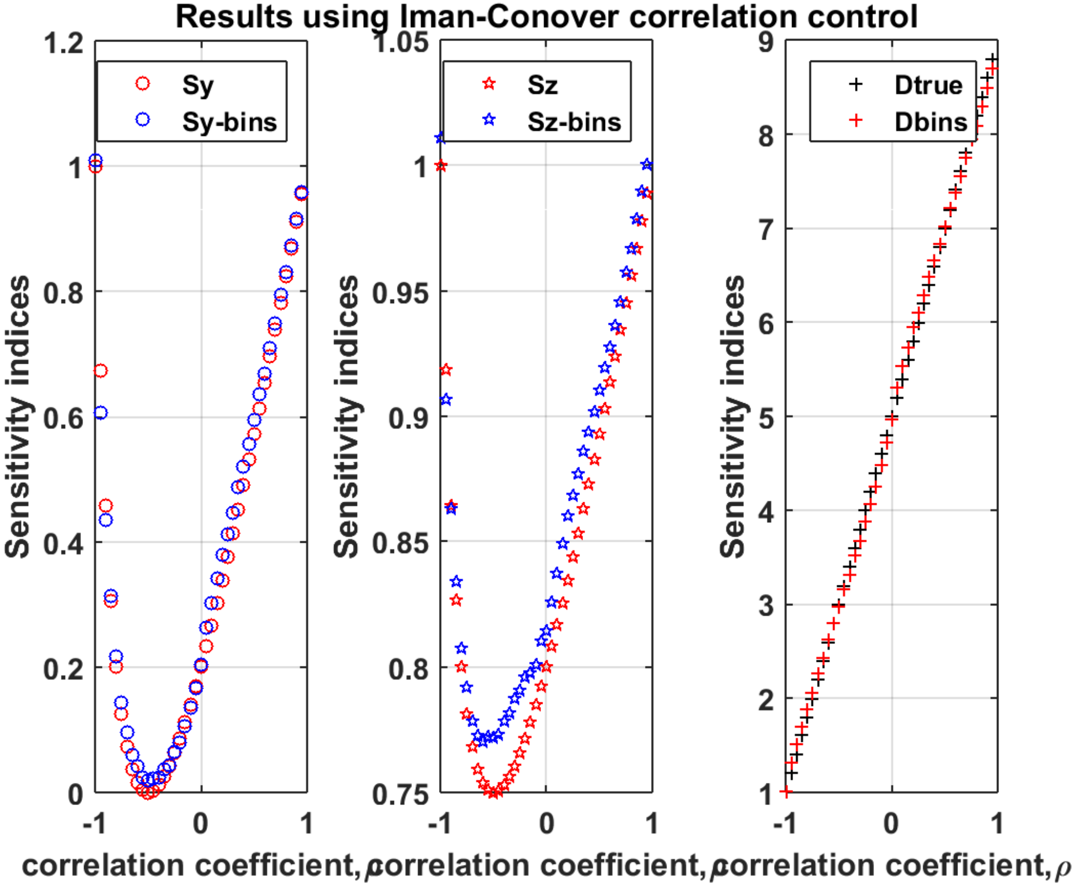


Approximate method: random sampling & bining

$$f(x) = x_1 + x_2$$
$$x_1 \sim N(0;1)\ ;\ x_2 \sim N(0;2)$$
$$\rho := -0.999 : 0.05 : 0.999$$

$$D_y = \frac{1}{M}\sum_{j=1}^{M}\left(\frac{1}{N_m^j}\sum_{k=1}^{N_m^j} f(y_k, z_k)\right)^2 - f_0^2.$$

Results are similar when generating dependent samples using IC method



Monte carlo estimators & Rosenblatt transformation (T.Mara algorithm)

Variance based sensitivity within Bayesian inferenecE (conditional distributions)

$$S_i^{full} = \frac{D_{x_i}\left[E_{\boldsymbol{x}_{\sim \boldsymbol{i}}|x_i}(\boldsymbol{y}|x_i)\right]}{D} \rightarrow (\overline{\boldsymbol{x}}_{\sim \boldsymbol{i}}, x_i) \sim p(\overline{\boldsymbol{x}}_{\sim \boldsymbol{i}}|x_i)p(x_i)$$: Full Main effects

$$S_i^{ind} = \frac{D_{x_i|\boldsymbol{x}_{\sim \boldsymbol{i}}}\left[E_{\boldsymbol{x}_{\sim \boldsymbol{i}}}(\boldsymbol{y}|x_i)\right]}{D} \rightarrow (\boldsymbol{x}_{\sim \boldsymbol{i}}, \overline{x}_{\boldsymbol{i}}) \sim p(\boldsymbol{x}_{\sim \boldsymbol{i}})p(\overline{x}_i|\boldsymbol{x}_{\sim \boldsymbol{i}})$$: Independent main effects

$$ST_i^{ind} = \frac{E_{\boldsymbol{x}_{\sim \boldsymbol{i}}}\left[D_{x_i|\boldsymbol{x}_{\sim \boldsymbol{i}}}(\boldsymbol{y}|\boldsymbol{x}_{\sim \boldsymbol{i}})\right]}{D} \rightarrow (\boldsymbol{x}_{\sim \boldsymbol{i}}, \overline{x}_i) \sim p(\boldsymbol{x}_{\sim \boldsymbol{i}})p(\overline{x}_i|\boldsymbol{x}_{\sim \boldsymbol{i}})$$: Ind. Total effects

$$ST_i^{full} = \frac{E_{\boldsymbol{x}_{\sim \boldsymbol{i}}|x_i}\left[D_{x_i}(\boldsymbol{y}|\boldsymbol{x}_{\sim \boldsymbol{i}})\right]}{D} \rightarrow (\overline{\boldsymbol{x}}_{\sim \boldsymbol{i}}, x_i) \sim p(\overline{\boldsymbol{x}}_{\sim \boldsymbol{i}}|x_i)p(x_i)$$: Full total effects

T.A. Mara et al. / Environmental Modelling & Software 72 (2015)

Monte carlo estimators & Rosenblatt transformation (T.Mara algorithm)

Variance based sensitivity within Bayesian inferenecE (conditional distributions). Some features:

- $(S_i^{full}, ST_i^{full}, S_i^{ind}, ST_i^{ind}) \in [0,1]$
- $S_i^{full} \leq ST_i^{full}$ and $S_i^{ind} \leq ST_i^{ind}$ $(S_i^{full} \nleq ST_i^{ind}, S_i^{ind} \nleq ST_i^{full})$
- $ST_i^{ind} = 0$, x_i is mainly influential because of its dependence with $\mathbf{x}_{\sim i}$

T.A. Mara et al. / Environmental Modelling & Software 72 (2015)

Monte Carlo estimators for variance based indices

Here **x** and **x'** are two random generated N input samples. V is the total variance of model output y (^ denotes the fact that these are estimators):

$$\widehat{S}_i = \frac{\frac{1}{N}\sum_{k=1}^{N} f(\mathbf{x}_k) \times \left(f\left(\mathbf{x}_k^i\right) - f\left(\mathbf{x}_k'\right)\right)}{\widehat{V}}$$

$$\widehat{ST}_i^{ind} = \frac{\frac{1}{N}\sum_{k=1}^{N} \left(f\left(\mathbf{x}_k^{i-1}\right) - f\left(\mathbf{x}_k'\right)\right)^2}{2\widehat{V}}$$

$$\widehat{S}_{i-1}^{ind} = \frac{\frac{1}{N}\sum_{k=1}^{N} f(\mathbf{x}_k) \times \left(f\left(\mathbf{x}_k^{i-1}\right) - f\left(\mathbf{x}_k'\right)\right)}{\widehat{V}}$$

$$\widehat{ST}_i = \frac{\frac{1}{N}\sum_{k=1}^{N} \left(f\left(\mathbf{x}_k^i\right) - f\left(\mathbf{x}_k'\right)\right)^2}{2\widehat{V}}$$

T.A. Mara et al. / Environmental Modelling & Software 72 (2015)

Monte Carlo sampling algorithm with permutation & Iman-Conover rank correlation (T.Mara code)

Sampling-based estimators & IC procedure

The sampling-based algorithm for GSA:

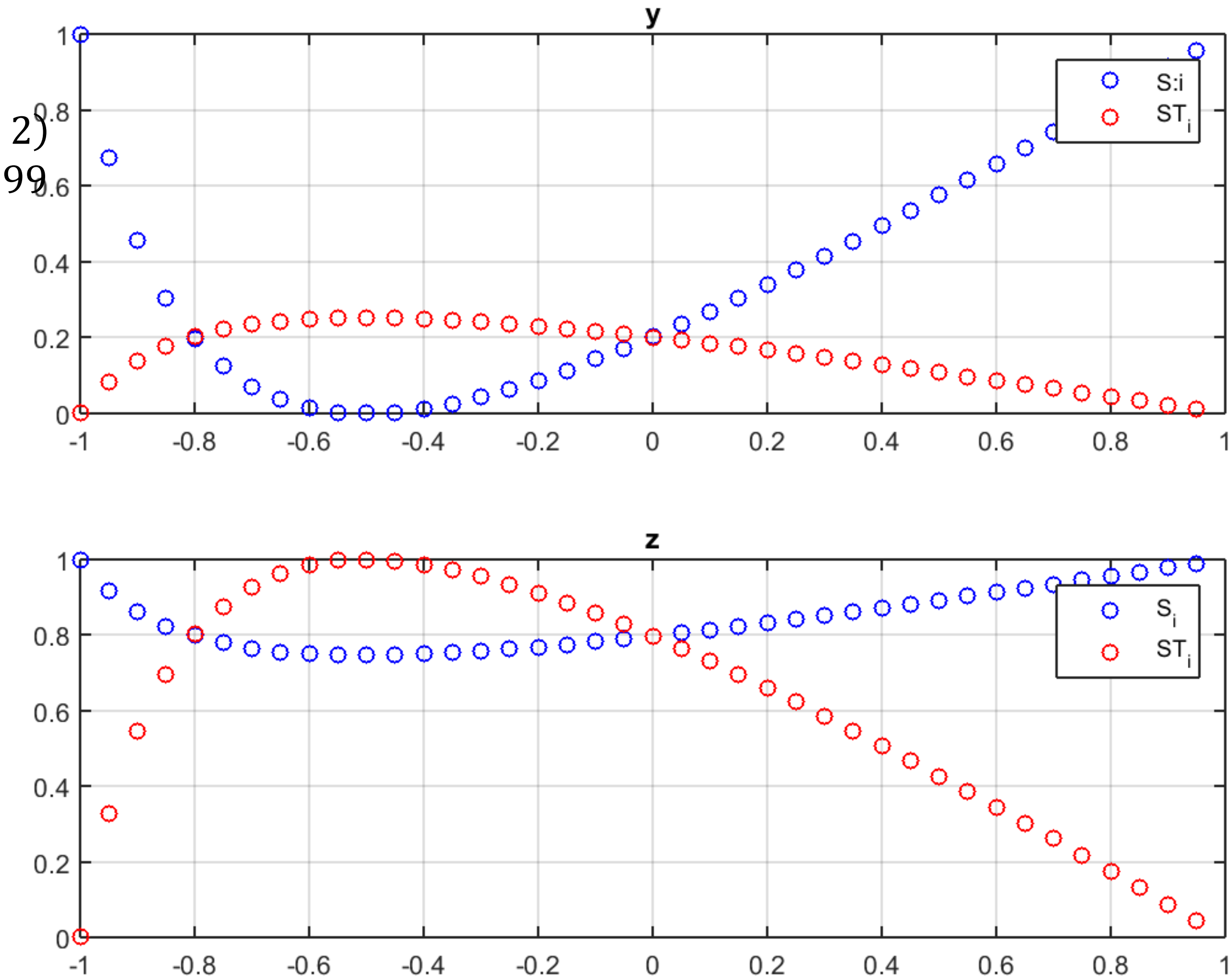
1. Generate two independent standard normal samples $\mathbf{Z}$ and $\mathbf{Z}'$
2. For $k = 1, \dots, n$, from the IC procedure, generate two samples
   - $\mathbf{X} = [X_k, \bar{X}_{k+1}, \dots, \bar{X}_n, \bar{X}_1, \dots, \bar{X}_{k-1}]$ from $\mathbf{Z}$
   - $\mathbf{X}' = [X'_k, \bar{X}'_{k+1}, \dots, \bar{X}'_n, \bar{X}'_1, \dots, \bar{X}'_{k-1}]$ from $\mathbf{Z}'$
3. Evaluate $Y = f(\mathbf{X})$, $Y' = f(\mathbf{X}')$
   - Set $\mathbf{Z}^{(k)} = [Z'_1, Z_2, \dots, Z_n]$
   - Set $\mathbf{Z}^{(n+k)} = [Z_1, \dots, Z_{n-1}, Z'_n]$
4. As previously (IC) guess:
   - $\mathbf{X}^{(k)} = [\bar{X}_1, \dots, \bar{X}_{k-1}, X'_k, \bar{X}_{k+1}, \dots, \bar{X}_n]$
   - $\mathbf{X}^{(n+k)} = [X_1, \dots, X_{k-2}, \bar{X}'_{k-1}, X_k, \dots, X_n]$
5. Evaluate $Y^{(k)} = f(\mathbf{X}^{(k)})$ and $Y^{(n+k)} = f(\mathbf{X}^{(n+k)})$
6. Infer $(\hat{S}_k^{full}, \hat{ST}_k^{full}, \hat{S}_{k-1}^{ind}, \hat{ST}_{k-1}^{ind})$

Total computational cost
4*n*N

T.A. Mara et al. / Environmental Modelling & Software 72 (2015)

Sampling based estimators & IC based transformation: Simple example

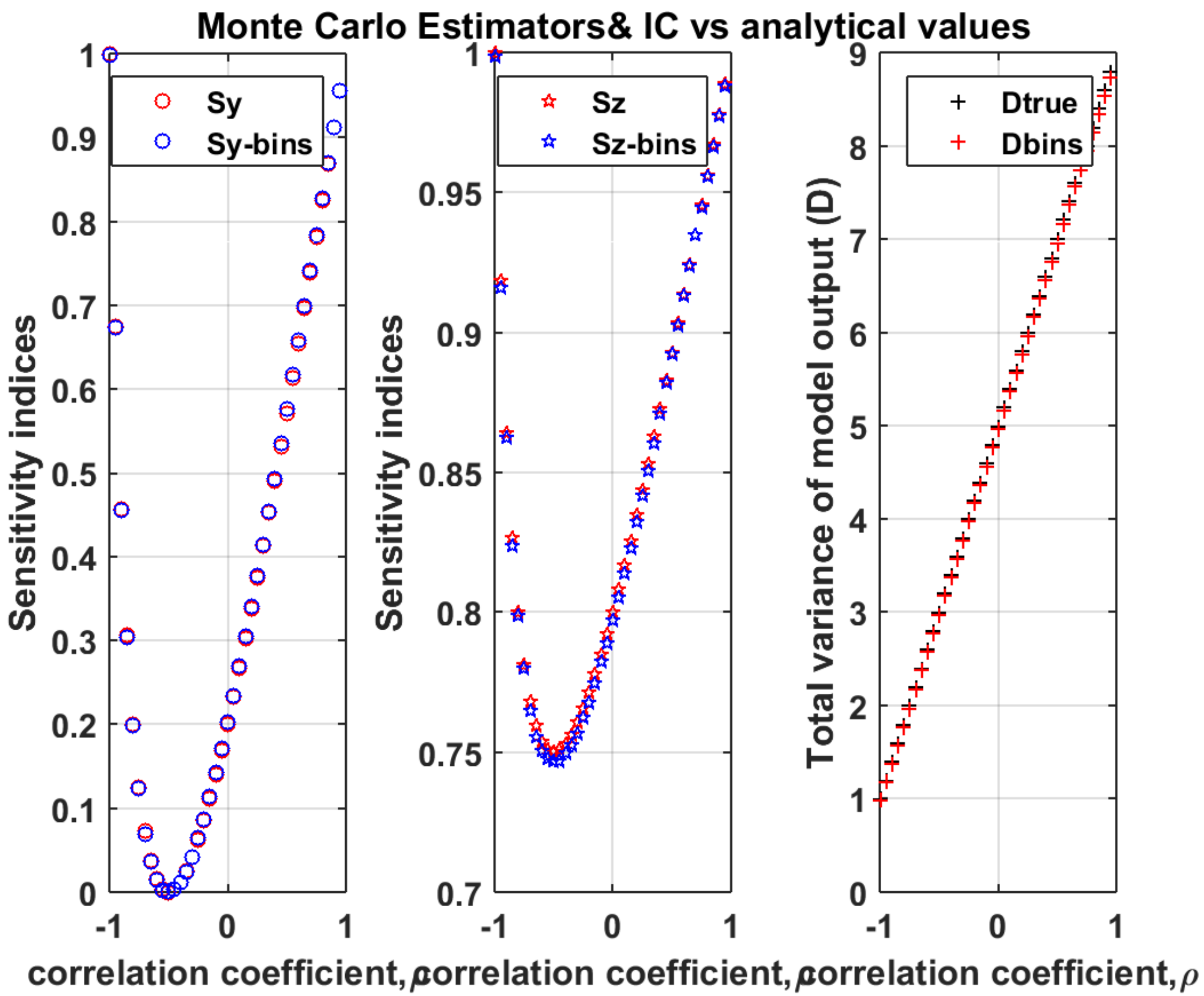
$$f(x) = x_1 + x_2$$
$$x_1 \sim N(0;1)\ ;\ x_2 \sim N(0;2)$$
$$\rho := -0.999 : 0.05 : 0.999$$



GSA_MC_Correlation.m

Sampling based estimators & IC based transformation: Simple example

$$f(x) = x_1 + x_2$$
$$x_1 \sim N(0;1)\ ;\ x_2 \sim N(0;2)$$
$$\rho := -0.999 : 0.05 : 0.999$$



Example 2: Nonlinear model (Ishigami)

COMPUTE SENSITIVITY INDICES OF A NONLINEAR MODEL

Monte Carlo estimators & IC with permutation of correlation matrix

Monte Carlo estimators & IC method with permutation using Mara algorithm:

Step 1. Define distribution & its parameters for the inputs

Step 2. Random Sampling: generate two standard normal samples **x** and **x'**

Step 3. for each input factor k=1:nVar, starting from original correlation matrix for nVar 1, generate 4 mixture matrices (x,x') and do IC correlation control & evaluate the model

Repeat for other inputs this time permuting correlation matrix

Step 4. Compute and plot Si and S_{Ti} indices

Step 5. Interpret results

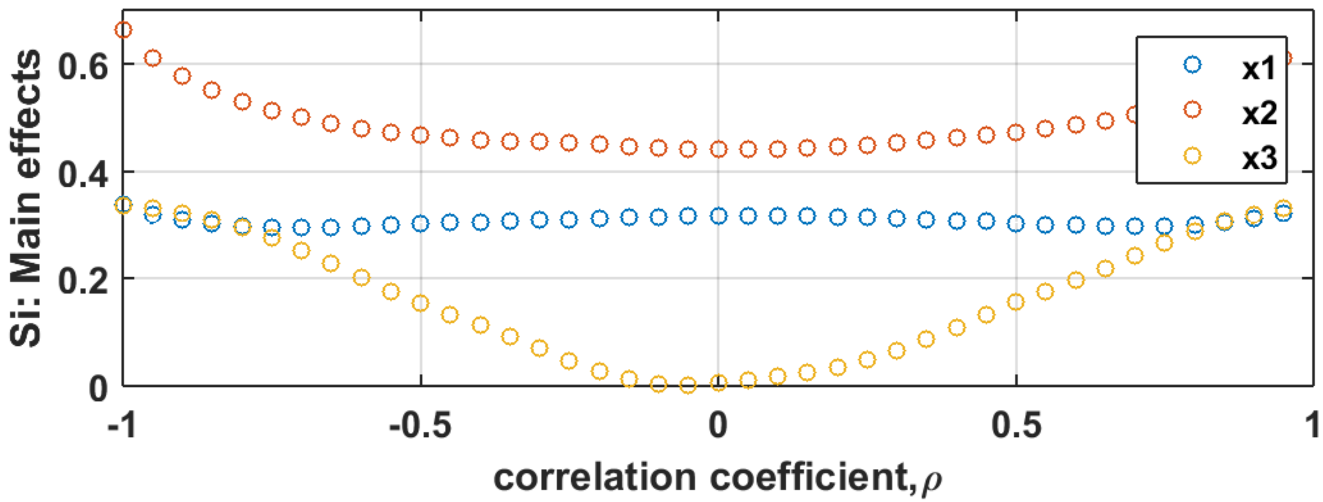
Ishigami model

The Ishigami model is a very nonlinear model with uniformly distributed inputs:

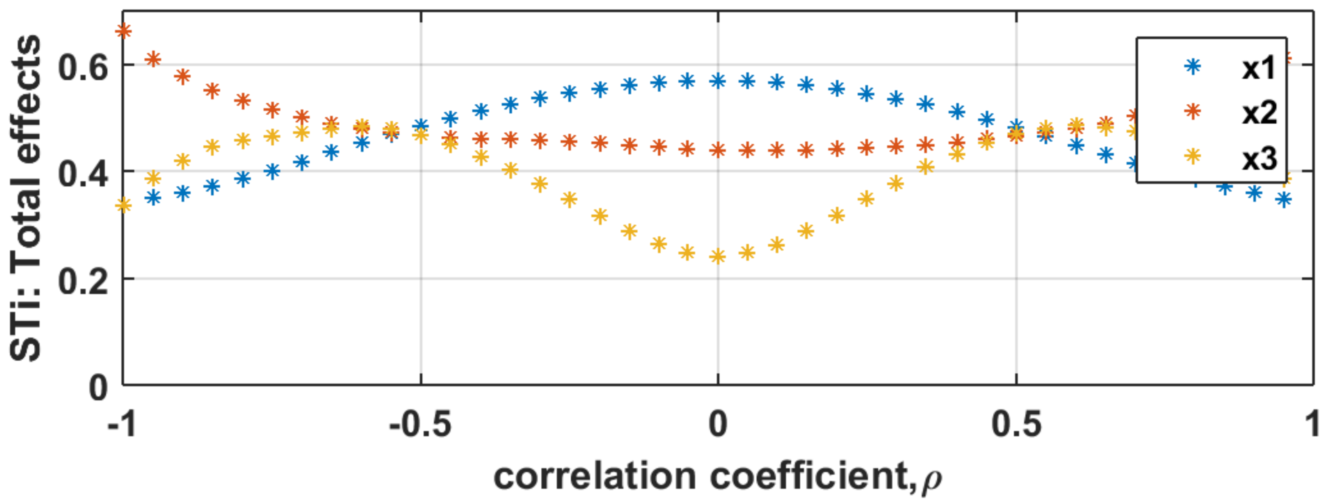
$$f(x) = sin(x_1) + 7sin(x_2)^2 + 0.1(x_3)^4 sin(x_1) \quad \text{where } x_i \sim U(-pi, pi)$$

GSA_MC_Correlation_ishigamimodel.m

Results

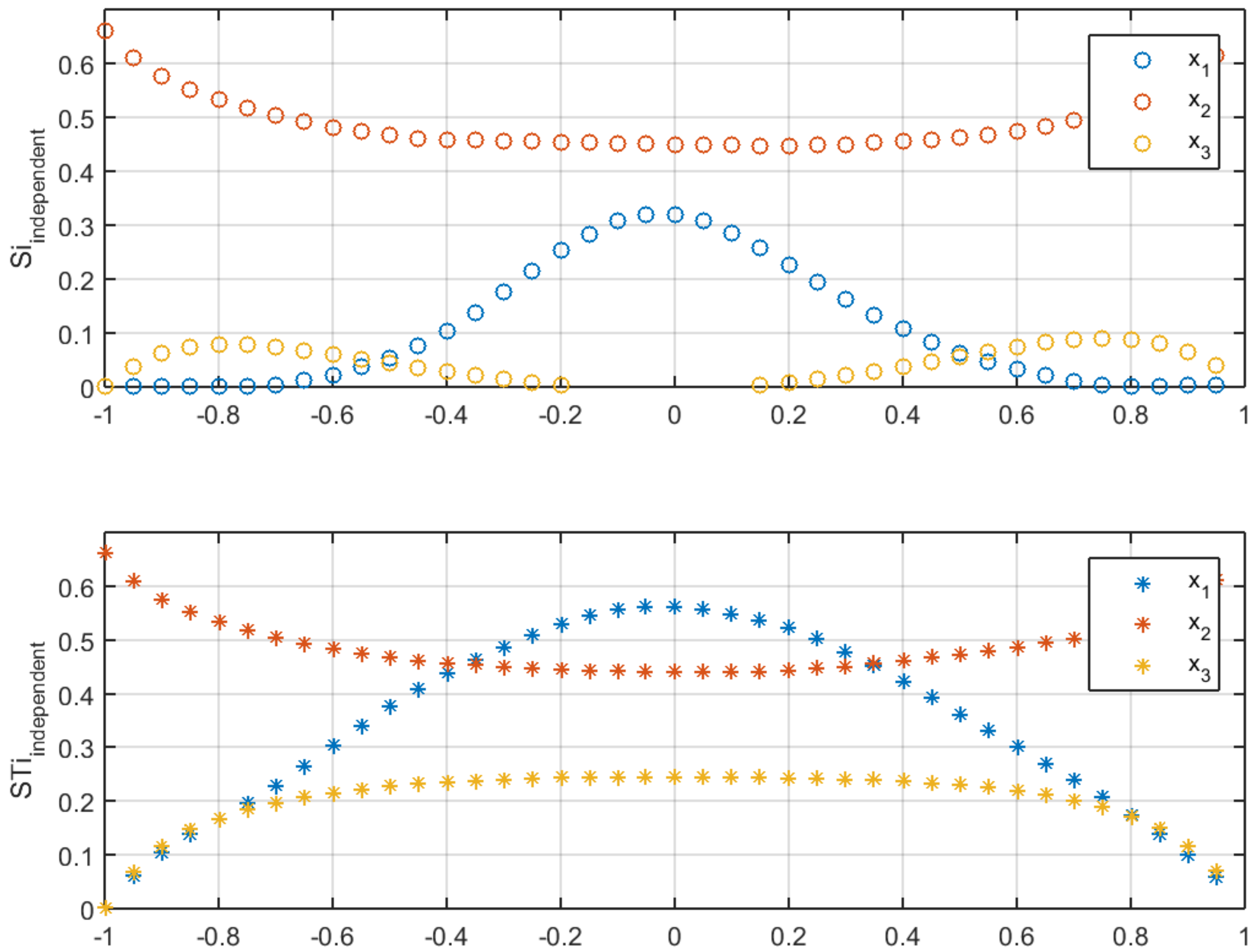


Si: Main effects
0.6
0.4
0.2
0
-1
-0.5
0
0.5
1
correlation coefficient, ρ
x1
x2
x3



STi: Total effects
0.6
0.4
0.2
0
-1
-0.5
0
0.5
1
correlation coefficient, ρ
x1
x2
x3

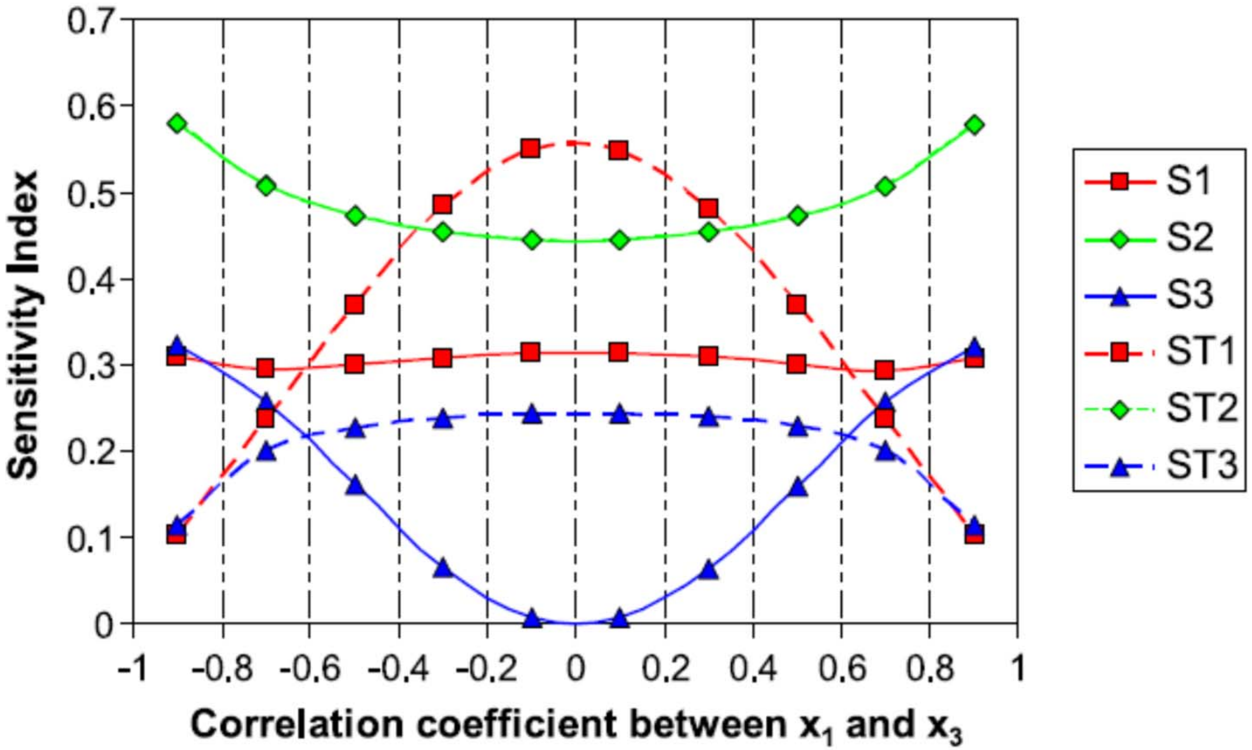
Results



0.6
0.5
0.4
0.3
0.2
0.1
0
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
$Si_{independent}$
x_1
x_2
x_3
$STi_{independent}$

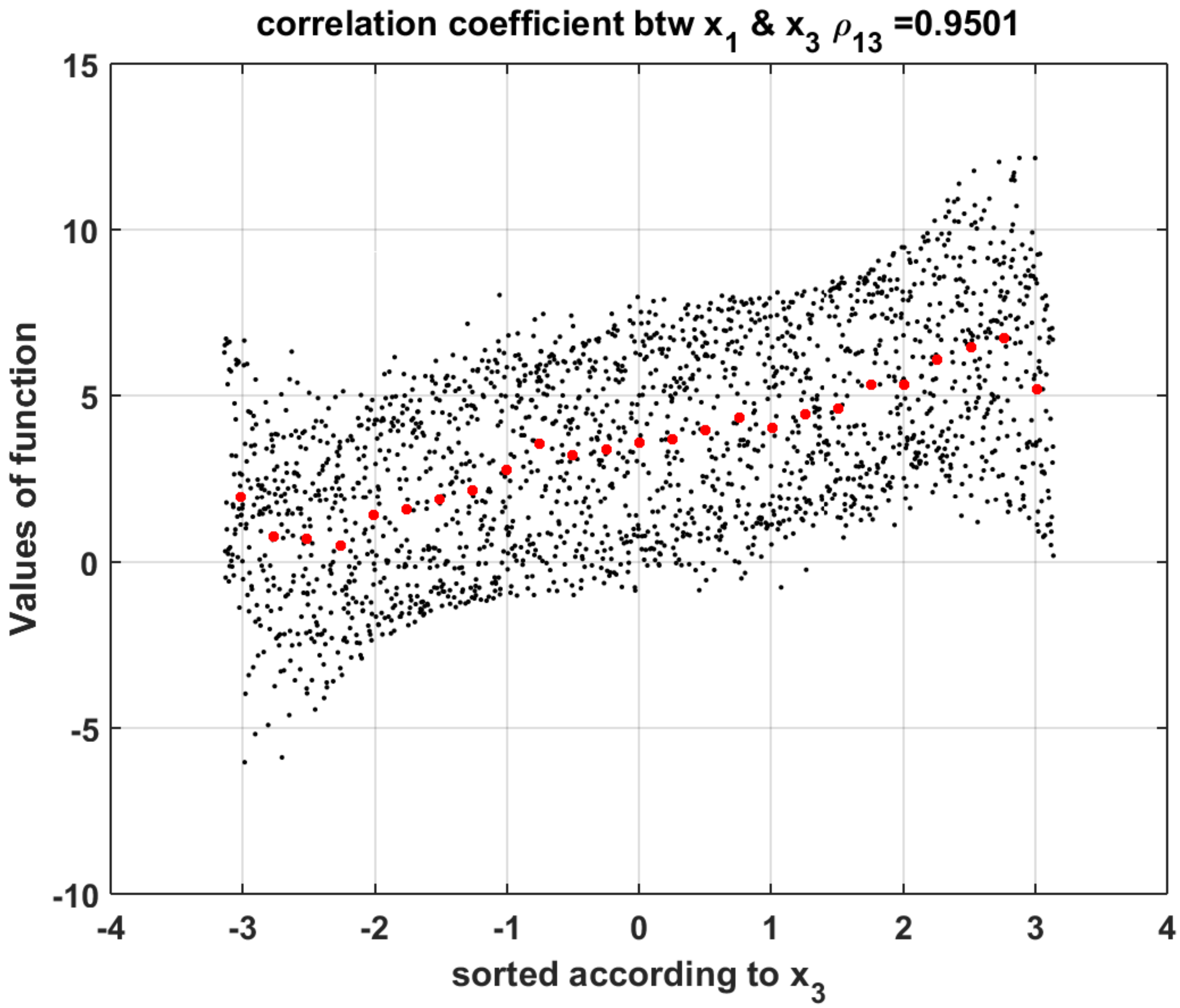
Compare with the copula method of Kucherenko

S. Kucherenko et al. / Computer Physics Communications 183 (2012) 937–946

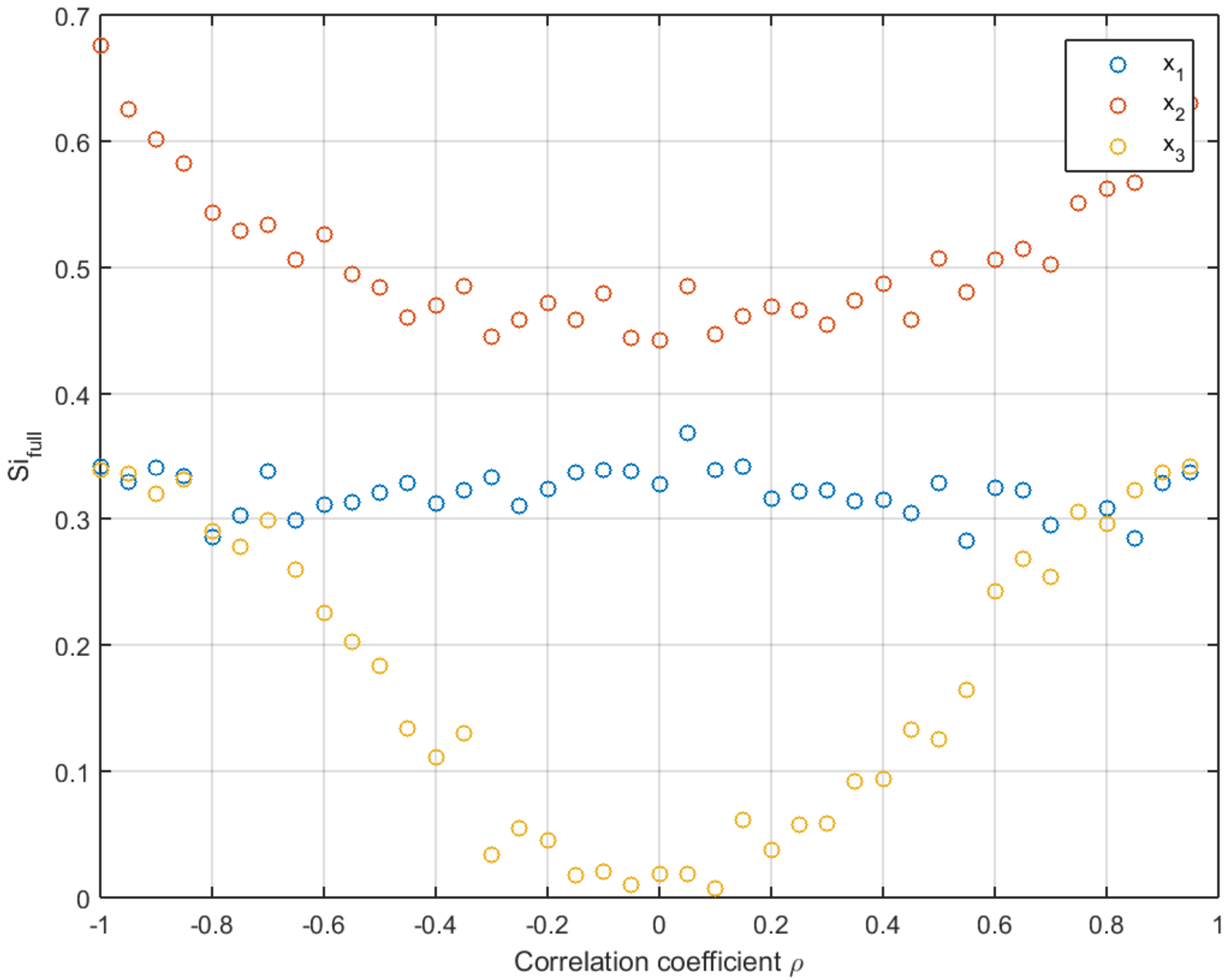


The results between T.Mara & Kucherenko mostly OK (see the behaviour of S3 in T. Mara). Overall conclusion OK.

Compare with random sampling & bining approach



Compare with random sampling & bining approach



To sum up

Random sampling with bining is an excellent method. Simple and effective. The challenge is that it is not yet possible to apply for estimating STi (full effects).

The algorithm of Mara has the advantage of being model-free (no need to develop a surrogate model) and generic, flexible. The disadvantage is that it uses Iman Conover method (not the most efficient method for imposing correlation control).

It compares well with the method of Kucherenko et al (2012) that uses Gaussian copula for conditional sampling but not exactly. The reason is that Gaussian copula is a direct method for generating dependent samples where IC is not.

So : always use random sampling with bining as a first method. Then the method of Mara.

Exercise 1:

SIMPLE NONLINEAR MODEL

Exercise details

Consider the following problem:

$$y = 4x_1^2 + 3x_2$$

$$with$$

$$x_1 \sim U\left(-0.5, 0.5\right)$$

$$x_2 \sim U\left(-0.5, 0.5\right)$$

Calculate Si & STi. Use the matlab scripts, provided to you.

Consider a range of rho: -0.999:0.05:0.999