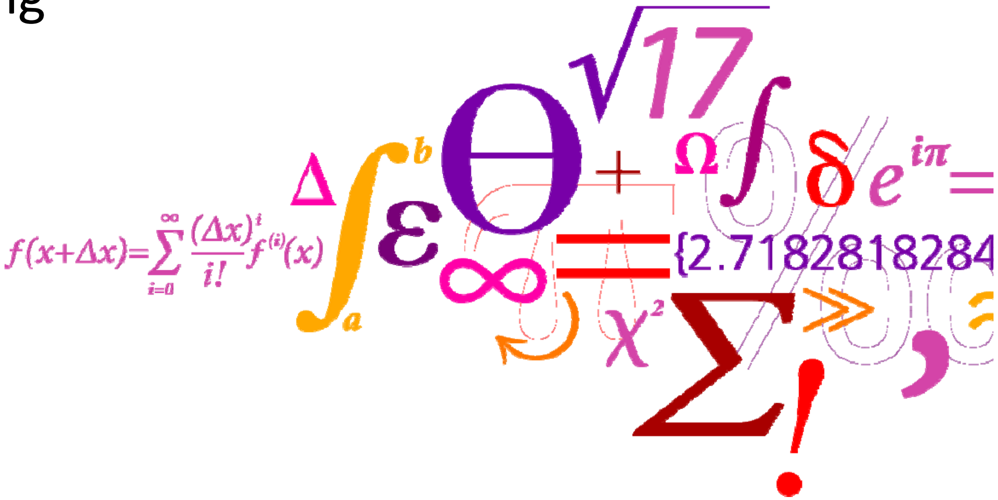
Lecture 4.2a Variance-based sensitivity analysis for models with independent inputs

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
  - Understand and use analytical, approximate and monte-carlo based solution methods to calculate variance based sensitivity indices

Outline

- Variance-based sensitivity measure
  - Sobol’s HDMR
  - Law of total variance
- Computing sensitivity indices
- Analytical example: a simple linear model
- Exercise: a nonlinear model

Variance based-sensitivity measure: Sobol's method

For simplicity, let us take a model of the form:

$$y = f(\boldsymbol{\theta}) \quad where\, \boldsymbol{\theta} = (\theta_1, \ldots, \theta_k)$$

Sobol's method considers a higher dimensional model representation (HDMR) (note: this is not series expansion as it has finite terms.)

$$f = f_0 + \sum_{i=1}^{k} f_i(\theta_i) + \sum_{i=1}\sum_{j>i} f_{ij}(\theta_i, \theta_j) + \ldots + f_{1,2,\ldots k}(\theta_i, \theta_j, \ldots, \theta_k)$$

Where each term is chosen with zero mean:

$$\int_0^1 f(\theta_i)\, d\theta_i = 0 \;\; \forall \theta_i \; i = 1,2 \ldots k$$

$$\int_0^1 \int_0^1 f(\theta_i) f(\theta_j)\, d\theta_i d\theta_j = 0 \;\; \forall \theta_i, \theta_j \; i < j$$

Variance based-sensitivity measure: Sobol's method

$$f = f_0 + \sum_{i=1}^{k} f_i(\theta_i) + \sum_{i=1} \sum_{j>i} f_{ij}(\theta_i, \theta_j) + \ldots + f_{1,2,\ldots k}(\theta_i, \theta_j, \ldots, \theta_k)$$

Then the HDMR decomposition is unique and has the following properties:

$$\int_{\Omega} f(\theta) = f_0$$

Mean/expected value of *f(θ)*

$$f_i = E(y|\theta_i) - f_0$$

Contribution of θi

$$f_{ij} = E\left(y \middle| \theta_i, \theta_j\right) - f_i - f_j - f_0$$

Joint of effects of θi & θj

Decomposition of variance

For independent inputs, the variance of y can be partitioned as follows:

$$var(y) = \sum_{i}^{k} V_i + \sum_{i}\sum_{j} V_{ij} + \sum_{i}\sum_{j}\sum_{l} V_{ijl} + \cdots V_{123..k}$$

Where

$$V_i = \int f(\theta_i)^2 d\theta_i \qquad f(\theta_i) = E(y|\theta_i) - f_0$$

Defining a sensitivity measure: Sobol's method

The variances of the terms in the HDMR decomposition are proposed as measures of sensitivity:

$$V\left(f_i(\theta_i)\right)=V\left(E\left(y|\theta_i\right)\right)$$

Hence the variance decomposition of sobol's HDMR model:

$$V=\sum_{i=1}^{k}V_i+\sum_{i=1}\sum_{j>i}V_{ij}\left(\theta_i,\theta_j\right)+\ldots+V_{i,j,\ldots k}\left(\theta_i,\theta_j,\ldots,\theta_k\right)$$

Dividing by V one obtains (first order) sensitivity indices (also known as Sobol index):

$$1=\sum_{i=1}^{k}S_i+\sum_{i=1}\sum_{j>i}S_{ij}+\ldots+S_{i,j,\ldots k}$$

Properties of sensitivity index, S_i

First-order sensitivity index has the following properties :

$$1 = \sum_{i=1}^{k} S_i + \sum_{i=1} \sum_{j>i} S_{ij} + \ldots + S_{12,\ldots k}$$

Hence, the following interpretation holds:

$\sum_{i=1} S_i \leq 1$ always

$\sum_{i=1} S_i = 1$ model is additive

$1 - \sum_{i=1} S_i$ indicates presence of interactions

Properties of sensitivity measure: Total effects, S_{Ti}

Total effects index, S_{Ti}, is total contribution to the output variance, var(*y*), due to parameter, θ_i:

$$1 = \sum_{i=1}^{k} S_i + \sum_{i=1} \sum_{j>i} S_{ij} + \ldots + S_{12,\ldots k}$$

Example: for a three-parameter model, total effects would be the sum of all the terms in S_i eqn (above):

$$S_{T1} = S_1 + S_{12} + S_{13} + S_{123}$$
$$S_{T2} = S_2 + S_{12} + S_{23} + S_{123}$$
$$S_{T3} = S_3 + S_{13} + S_{23} + S_{123}$$

Provides answer to :"Which parameter can be fixed arbitrarily in its range without affecting output variance, *var(y)?*" $S_{Ti} = 0$ is sufficient condition for this answer.

Hence, S_{Ti} useful information for factor fixing.

Law of total variance: a different look at the sensitivity measures

- Law of total variance

$$V(y) = V\left(E\left(y|\boldsymbol{\theta}_i\right)\right) + E\left(V\left(y|\boldsymbol{\theta}_i\right)\right)$$

Explained variance (due to variation in θi)

Residual variance (any variance due to sources other than θi)

law of total variance concept: revisiting S_i and S_{Ti} measures

Both S_i and S_{Ti} measure can be calculated from variance decomposition (law of variance) in fact.

Main effect of θi

$$V(y) = \boxed{V_{\boldsymbol{\theta}_i}\left(E_{\boldsymbol{\theta}_{\sim i}}\left(y|\boldsymbol{\theta}_i\right)\right)} + E_{\boldsymbol{\theta}_i}\left(V_{\boldsymbol{\theta}_{\sim i}}\left(y|\boldsymbol{\theta}_i\right)\right)$$

$$V(y) = V_{\boldsymbol{\theta}_{\sim i}}\left(E_{\boldsymbol{\theta}_i}\left(y|\boldsymbol{\theta}_{\sim i}\right)\right) + \boxed{E_{\boldsymbol{\theta}_{\sim i}}\left(V_{\boldsymbol{\theta}_i}\left(y|\boldsymbol{\theta}_{\sim i}\right)\right)}$$

Total effect of θi

$$S_i = \frac{V_{\boldsymbol{\theta}_i}\left(E_{\boldsymbol{\theta}_{\sim i}}\left(y|\boldsymbol{\theta}_i\right)\right)}{V(y)} \qquad S_{Ti} = \frac{E_{\boldsymbol{\theta}_{\sim i}}\left(V_{\boldsymbol{\theta}_i}\left(y|\boldsymbol{\theta}_{\sim i}\right)\right)}{V(y)}$$

Main (First Order), Si vs Total effect indices, S_{Ti}

Each measure has a different meaning obviously

$$S_i = \frac{V_{\boldsymbol{\theta}_i}\left(E_{\boldsymbol{\theta}_{\sim i}}\left(y|\boldsymbol{\theta}_i\right)\right)}{V\left(y\right)}$$

First order effect=
= the expected reduction in variance which would be achieved if factor θi could be fixed.

Si used for Factors prioritisation

$$S_{Ti} = \frac{E_{\boldsymbol{\theta}_{\sim i}}\left(V_{\boldsymbol{\theta}_i}\left(y|\boldsymbol{\theta}_{\sim i}\right)\right)}{V\left(y\right)}$$

Total effect
= the expected variance which would be left if all parameters but θi could be fixed.

STi is used for Fixing (dropping) non-important factors

One remark: Si vs SRC

It is important to remark that for linear and additive models the following relationship between Si (first-order sensitivity index) and SRC (standardized regression coefficient) holds:

$$S_i = \beta_i^2$$

The proof can be checked at Salltelli et al (2009) pp 23.

This provides a nice data quality check on the results.

Numerical calculation of Sensitivity indices

To compute S_i and S_{Ti}, one needs to estimate the conditional variances $V(E(y|\theta_i))$ and $E(V(y|\theta_{\sim i}))$

There are two approaches:

1) Analytical (exact) solution
2) Numerical solutions using Brute force, random sampling with bins, efficient Monte-Carlo sampling methods.

Analytical solution

This uses symbolic integral evaluation of HDMR terms. A simple example:

$$f(x) = x_1 + x_2 + x_3$$

$$x_1 \sim U(0.5; 1.5);\ x_2 \sim U(1.5; 4.5)\ \&\ x_3 \sim U(4.5; 13.5)$$

$$p(x) = \frac{1}{b-a}$$

$$f_0 = \int f(x)p(x)dx = 13$$

$$\mathrm{D} = \int f(x)^2 p(x)dx = 7.5833$$

$$f_1(x_1) = E(y|x_1) - f_0 = \int f(x)p(x_1)dx_1 - f_0 = x_1 - 1$$

$$f_2(x_2) = E(y|x_2) - f_0 = \int f(x)p(x_2)dx_2 - f_0 = x_2 - 3$$

$$f_1(x_3) = E(y|x_3) - f_0 = \int f(x)p(x_3)dx_3 - f_0 = x_3 - 9$$

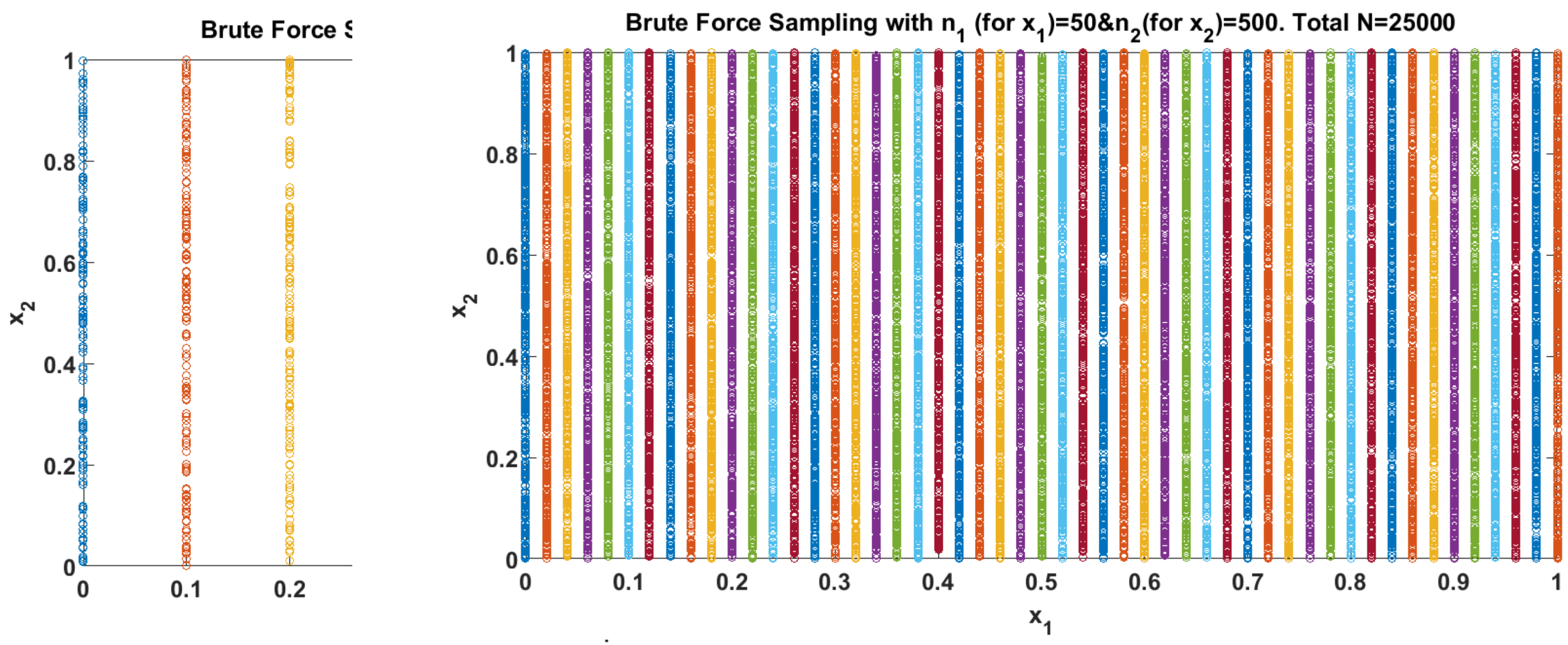
$$f_{12}(x_1, x_2) = 0;\ f_{13}(x_1, x_3) = 0;\ f_{23}(x_2, x_3) = 0$$

	Si=Vi/D
X_1	0.011
X_2	0.099
X_3	0.89

Brute force method

Brute force method, uses intuitive approach (two nested full-factorial sampling): e.g. to estimate $V(E(y|\theta_i))$, one fixes θ_i at some point in its range and perform *N* number of simulations using parameter samples minus θ_i. This is repeated r times to average over the range of θ_i.

Total costs: N*r*k (typically $k*N^2$). Too many evaluations (10^5) easily needed.



Approximate method: random sampling & bining

N monte carlo simulations

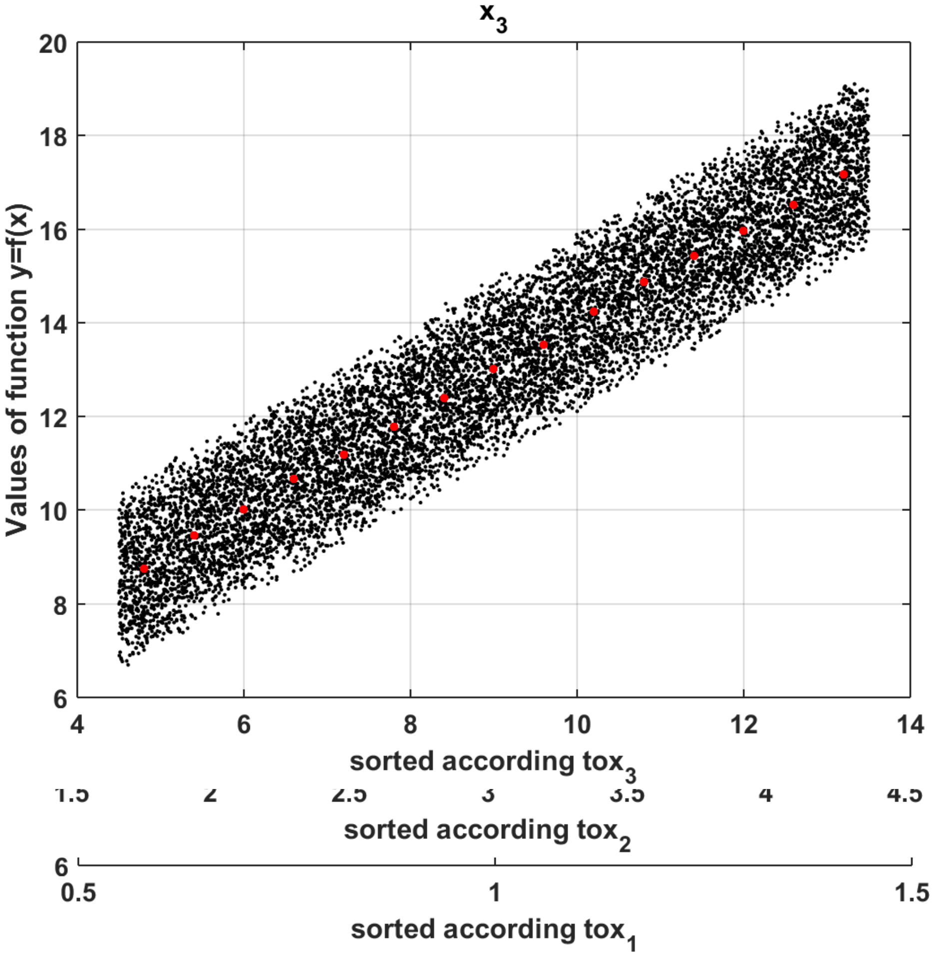
M=no of bins

j=no of points in each bin

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

$$f(x) = x_1 + x_2 + x_3$$

$x_1 \sim U(0.5; 1.5)$; $x_2 \sim U(1.5; 4.$
$x_3 \sim U(4.5; 13.5)$



Approximate method: random sampling & bining

N monte carlo simulations

M=no of bins

j=no of points in each bin

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

$$f(x) = x_1 + x_2 + x_3$$

$$x_1 \sim U(0.5; 1.5)\ ;\ x_2 \sim U(1.5; 4.5)$$
$$x_3 \sim U(4.5; 13.5)$$

	Si_analytic	**Si_rand&bin**
$\mathbf{X_1}$	0.011	0.011
$\mathbf{X_2}$	0.099	0.103
$\mathbf{X_3}$	0.89	0.949
(N,m)		(1000,15)

Efficient Monte Carlo sampling

Alternatively, monte carlo sampling is used. Generate 2 matrices of random samples A and B with *Nxk* dimension. And then define mixture matrices $\mathbf{A_B}^i$ and $\mathbf{B_A}^i$. Explanation: $\mathbf{A_B}^i$ where column i comes from matrix B and all other k −1 columns come from matrix A. $\mathbf{B_A}^i$ matrix, where column i comes from matrix A and all other k −1 columns come from matrix B

$$A = \begin{pmatrix} \theta_{11} & \dots & \theta_{1i} & \dots & \theta_{1k} \\ \vdots & & \vdots & & \vdots \\ \theta_{N1} & \dots & \theta_{Ni} & \dots & \theta_{Nk} \end{pmatrix} \qquad \mathbf{A}_{\mathbf{B}}^{i} = \begin{pmatrix} \theta_{11} & \dots & \theta'_{1i} & \dots & \theta_{1k} \\ \vdots & & \vdots & & \vdots \\ \theta_{N1} & \dots & \theta'_{Ni} & \dots & \theta_{Nk} \end{pmatrix}$$

$$B = \begin{pmatrix} \theta'_{11} & \dots & \theta'_{1i} & \dots & \theta'_{1k} \\ \vdots & & \vdots & & \vdots \\ \theta'_{N1} & \dots & \theta'_{Ni} & \dots & \theta'_{Nk} \end{pmatrix} \qquad \mathbf{B}_{\mathbf{A}}^{i} = \begin{pmatrix} \theta'_{11} & \dots & \theta_{1i} & \dots & \theta'_{1k} \\ \vdots & & \vdots & & \vdots \\ \theta'_{N1} & \dots & \theta_{Ni} & \dots & \theta'_{Nk} \end{pmatrix}$$

Computation of Sensitivity indices

Perform Monte Carlo simulations (evaluate the model with A, B and k times $\mathbf{A_B}^i$ and $\mathbf{B_A}^i$ matrices) obtaining three vectors of model outputs:

$$\mathbf{y}_A = f(\mathbf{A}) \quad \mathbf{y}_B = f(\mathbf{B}) \quad \mathbf{y}_{BAi} = f\left(\mathbf{B}_{\mathbf{A}}^i\right) \quad \mathbf{y}_{ABi} = f\left(\mathbf{A}_{\mathbf{B}}^i\right)$$

Compute the S_i measure (different approximations)

$V(E(y\|\theta_i))$ for Si calculation	Reference
$(1/N)\sum_j^N y_A(j)\,y_{BAi}(j) - f_0^2$	Sobol (1993)
$V(y) - (1/2N)\sum_j^N \left(y_B(j) - y_{ABi}(j)\right)^2$	Jansen (1999)
$(1/N)\sum_j^N y_B(j)\left(y_{ABi}(j) - y_A(j)\right)$	Saltelli (2010)

Computation of Sensitivity indices: Sobol's method

Perform Monte Carlo simulations (evaluate the model with A, B and k times $\mathbf{A}_\mathbf{B}^i$ and $\mathbf{B}_\mathbf{A}^i$ matrices) obtaining three vectors of model outputs:

$$\mathbf{y}_A = f(\mathbf{A}) \quad \mathbf{y}_B = f(\mathbf{B}) \quad \mathbf{y}_{BAi} = f\left(\mathbf{B}_\mathbf{A}^i\right) \quad \mathbf{y}_{ABi} = f\left(\mathbf{A}_\mathbf{B}^i\right)$$

Compute the S_i measure (different approximations)

$E(V(y\|\theta_{\sim i}))$ for S_Ti calculation	Reference
$(1/N)\sum_j^N y_A(j)\left(y_A(j)-y_{ABi}(j)\right)$	Sobol (2007)
$(1/2N)\sum_j^N \left(y_A(j)-y_{ABi}(j)\right)^2$	Jansen (1999)
$(1/2N)\sum_j^N \left(y_A(j)-y_{ABi}(j)\right)^2$	Saltelli (2010)* recommended best practice

Example 1: A simple model

COMPUTE SENSITIVITY INDICES OF A SIMPLE LINEAR MODEL

Sobol's method of sensitivity with Monte Carlo sampling

Monte Carlo simulations + Saltelli method of S_i and S_{Ti} computation

Step 1. Specify range for each input parameter

Step 2. Random Sampling: generate A, B and Ci matrices

Step 3. Model evaluations of A, B and Ci matrices

Step 4. Compute and tabulate Si and S_{Ti}

Step 5. Interpret results / compare them with SRC !

Step 1: Input ranges of model parameters

This simple exercise is taken from Saltelli et al 2009, pp 174:

$$y = x_1 + x_2 + x_3$$

$$with$$

$$x_1 \sim U(0.5, 1.5)$$

$$x_2 \sim U(1.5, 4.5)$$

$$x_3 \sim U(4.5, 13.5)$$

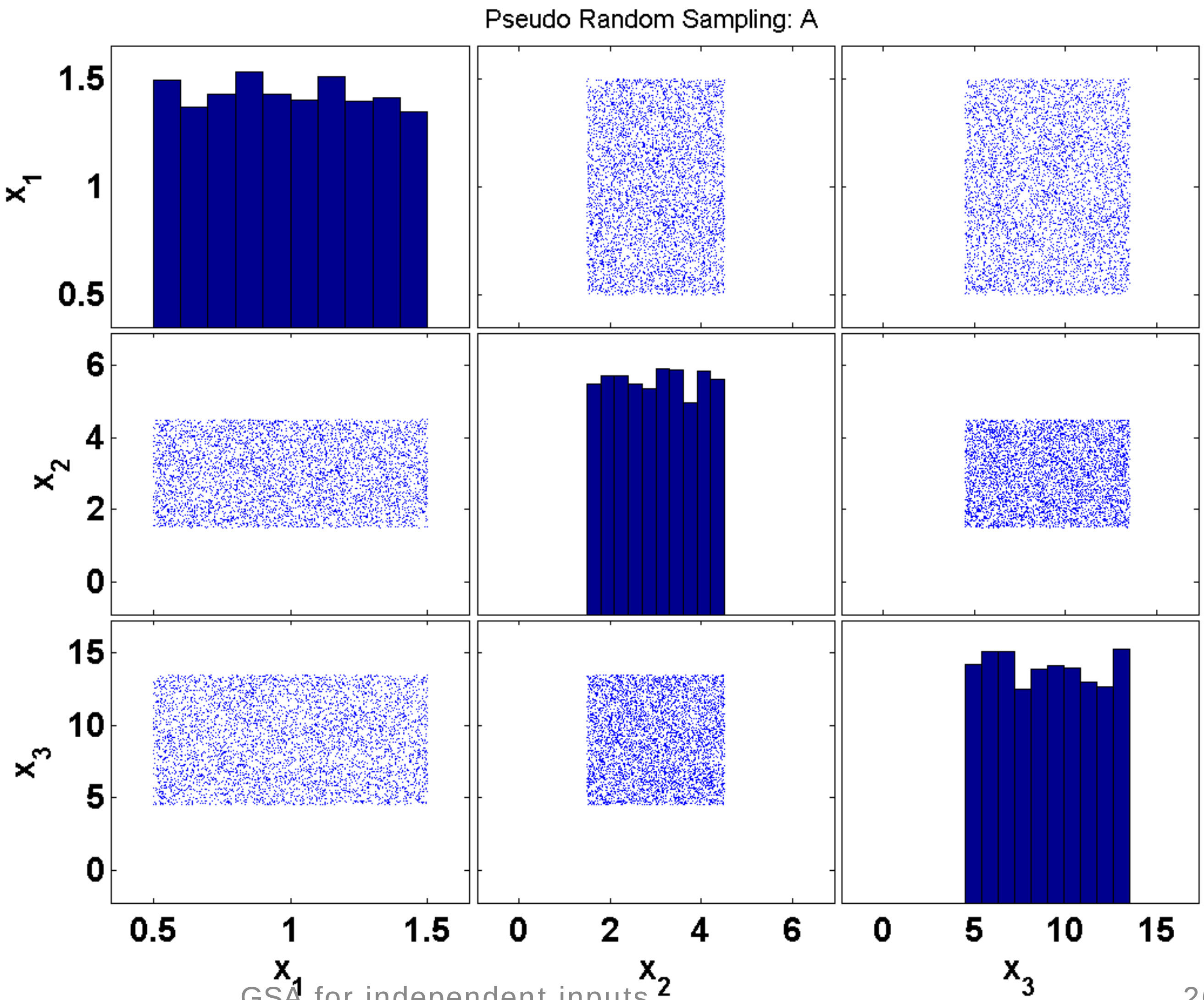
step 2: Random sampling results

Matlab code (randomsampling.m)

```
%% Define a priori probability distribution of parameters.
% uniform distribution is considered.
par = {'x_1','x_2','x_3'};
xlu =[ 0.5   1.5    4.5;
    1.5   4.5    13.5];
nvar = length(xlu);
% specify no of samples
nsample = 5000;
%% generate two matrices of random samples nsampleXnvar
Ap = rand(nsample,nvar) ; % 'rand' generates pseudo-random numbers
Bp = rand(nsample,nvar) ;
%% from probability to value
% uniform distribution
Xl = ones(nsample,1) * xlu(1,:) ; % this is needed for unifinv
Xu =  ones(nsample,1) * xlu(2,:) ;% this is needed for unifinv
A = unifinv(Ap,Xl,Xu);
B = unifinv(Bp,Xl,Xu);
```

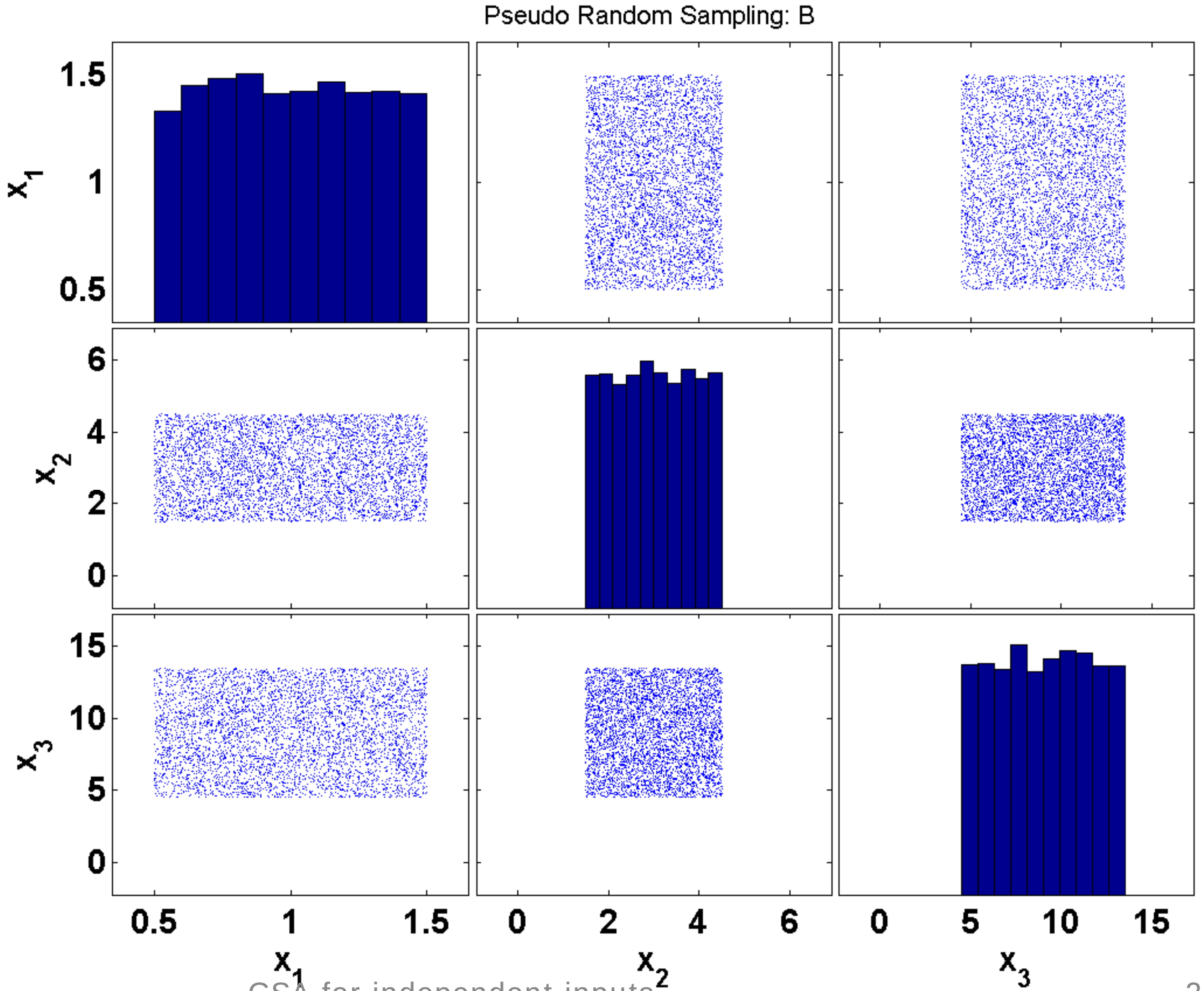
step 2: Random sampling results

N=5000 and k = 3



step 2: Random sampling results

N=5000 and k = 3



step 3-4. Perform MCs and compute Si and STi

Matlab scripts (mcsims.m and computeSiandSTi.m)

```
%% run Monte Carlo simulations for sampling matrix mcsims.m
for i=1:n
    % up
    par
     %%
    X1 =
    X2 =
    X3 =
     y(i
 end
cd(drnm)
hd = 'yA
save(hd)
cd(dr0)
```

```
%% calculate Si for each parameter                    computeSiandSTi.m
[n2 m2] = size(yA);
for i=1:m2 % for each model output
    ya = yA ;
    yb = yB ;
    mu = mean([ya; yb]); % to improve the estimate of mean
    vary = var([ya; yb]) ;
    for j=1:m % for each parameter
        %% sobol's method
        ybai = yBA(:,j) ;
        yabi = yAB(:,j) ;
        vx1(j,1)= mean(ya .* ybai) - mu^2 ;
        Si1(j,i)= vx1(j,1) / vary ; % first-order sensitivity index
        ex1(j,1)=  mean(ya .* (ya - yabi)) ;
        STi1(j,i)= ex1(j,1) / vary ; % total effects index
```

step 3-4. Perform MCs and compute Si and STi

	Si	*Si*	*Si*	*Si*	*Si*
N: samples	500	1000	2000	5000	Analytical Solution
x1	0.0193	-0.011	0.023	0.010	0.011
x2	0.177	0.125	0.122	0.096	0.099
x3	0.872	0.852	0.879	0.888	0.890
Sim cost: N*(k+2)	2500	5000	10000	25000	-

Sampling number is very important!
While exact solution may not be reached, however the relative ranking of parameter importance doesnt change.

Quasirandom sampling improves the efficiency

- See the quasirandomsampling script for generating Sobol sequences

	Si	*Si*	*Si*	*Si*	*Si*
N: samples	500	1000	1500	2000	Analytical Solution
x1	0.063	0.039	0.035	0.023	0.011
x2	0.14	0.108	0.11	0.108	0.099
x3	0.89	0.89	0.89	0.89	0.890
Sim cost: N*(k+2)	2500	5000	7500	10000	-

- Is this more efficient than random sampling?
- To be fair, it is better to use an average of several repetition and report some standard deviation on the indices

step 5. Interpretation

The higher the S_i, the more important a factor is. Plus sum of Si close to 1, hence model is additive.

	S_i	Rank	Analytical Solution	Rank
x1	0.010	1	0.011	1
x2	0.096	2	0.099	2
x3	0.888	3	0.890	3
Sum	0.994		1.00	

S_{Ti} which factor is non-influential. As none of S_{Ti} are zero, none of these factors can be deemed non-influential.

	S_{Ti}
x1	0.0112
x2	0.1018
x3	0.8919

To sum up

S_i is a global measure of sensitivity of model parameters

S_i indicates by how much one could reduce on average the output variance if a parameter could be fixed.

S_{Ti} is useful measure to fix non-influential parameters of the model (S_{Ti} =0). (though Morris screening more efficient computational wise).

The sum of all S_i equal to 1 for additive model, while sum of all S_{Ti} is always greater than 1.

Many methods to compute sensitivity indices: Analytical solution (integral evaluation of HDMR terms), brute force (not recommended), approximate random sampling with bins & efficient Monte Carlo sampling

Quasirandom sampling (e.g. Sobol or Halton sequences) improves computations efficiency

These methods are based on the assumption that inputs are independent.

Exercise 1: Sobol indices of a simple nonlinear model

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. Use the matlab scripts, provided to you.

Define your own sampling number. Repeat if necessary your calculations.

Analytical solution: Si=[0.106 0.894]