GSAT a Global Sensitivity Analysis Toolbox for Matlab

GSAT package includes routines for generic global sensitivity analysis (GSA). In particular it implements Sobol' analysis and FAST analysis to estimate the sensitivity indexes for models with up 50 different input parameters. In the literature of Sensitivity Analysis (SA), local sensitivity refers to the sensitivity at a fixed point in the parameter space (typically at the optimal fit point to the real data), while global sensitivity refers to an integrated sensitivity over the entire input parameter space. GSAT deals only with scalar models, which have one scalar and real output:

$$y = f(x)$$
$$y \in \mathbb{R}, x \in \mathbb{R}^n$$

In the field of GSA, variance-based methods refer to ways of quantifying the contribution of each input parameter (x_i) to the total variance of the output (y).

To use GSAT in Matlab you have to define first the model function:

```
File: mymodel.m

function y = mymodel(x, otherstuffifneeded)

% x is the vector of the input parameters
```

Let's assume you have a model (*mymodel.m*) with 3 input parameters x(1) x(2) and x(3) varying in the ranges of [0 1], [-100 100] and [5 7] respectively.

Once you have coded your own model function, in the command line (or in a script) you have to create a GSAT project and link it to the model and its parameters as following:

```
% create a new project

pro = pro_Create();
```

Add to the project the 3 input parameters, named param*, with their distribution domain and indicate that the parameters will be sampled following a Sobol set (the order you add the parameters will be the order you find them in the *x* vector passed to the model function to calculate *y*):

```
pro = pro_AddInput(pro, @()pdf_Sobol([0 1]), 'param1');

pro = pro_AddInput(pro, @()pdf_Sobol([-100 100]), 'param2');

pro = pro_AddInput(pro, @()pdf_Sobol([5 7]), 'param3');
```

If you wanted to sample parameters with a pdf uniformely distributed in a range [a b] (instead of Sobol sampling), you can write:

```
% pro = pro_AddInput(pro, @(N)pdf_Uniform(N, [a b]), 'parametername');
```

where N is the number of samples.

Set the model, and name it as 'model', to the project

```
pro = pro_SetModel(pro, @(x)mymodel(x otherstuffifneeded), 'model');
```

Set the number of samples for the quasi-random Monte Carlo sampling

```
pro.N = 20000;
```

Initialize the project by calculating the model at the sample points

```
pro = GSA_Init(pro);
```

Now you are ready to calculate the sensitivity indexes.

For example, calculate the first order global sensitivity coefficient for the second parameter (param2):

```
[S2 eS2 pro] = GSA_GetSy(pro, {2});
```

Or equivalent:

```
[S2 eS2 pro] = GSA_GetSy(pro, {'param2'});
```

Calculate the joint global sensitivity coefficient for the first two parameters:

```
[S12 eS12 pro] = GSA_GetSy(pro, {1,2});
```

Or equivalent:

```
[S12 eS12 pro] = GSA_GetSy(pro, {'param1','param2'});
```

Calculate the first order global sensitivity coefficients by using FAST

```
Sfast = GSA_FAST_GetSi(pro);
```

You will get a vector (*Sfast*) with the first order sensitivity coefficients for all the parameters.

Reference: Cannavo' F., Sensitivity analysis for volcanic source modeling quality assessment and model selection, Computers & Geosciences, Vol. 44, July 2012, Pages 52-59, ISSN 0098-3004, http://dx.doi.org/10.1016/j.cageo.2012.03.008.