Implementation Algorithm

Since the computation of the Sobol indices requires high-dimensional integration, the indices are approximated numerically. If one uses M Monte Carlo evaluations to approximate the mean $\mathbb{E}(Y|q_i)$ and repeats the procedure M times to approximate the variance $\text{var}[\mathbb{E}(Y|q_i)]$, a total of M^2 evaluations will be required to evaluate a single index. The total number of function evaluations required is M^2p, which is computationally prohibitive for a large parameter dimensions p. This motivated the author of [2] to provide a more efficient algorithm to compute Sobol indices that reduces the required evaluations to $M(p+2)$, based on Sobol's original approach in [4]. The algorithm was further improved by the authors of [3, 5, 6] and is summarized here.

Algorithm

1. Create two sample matrices A and B

$$A = \begin{bmatrix} q_1^1 & \dots & q_i^1 & \dots & q_p^1 \\ \vdots & & & & \vdots \\ q_1^M & \dots & q_i^M & \dots & q_p^M \end{bmatrix}, \text{ and } B = \begin{bmatrix} \hat{q}_1^1 & \dots & \hat{q}_i^1 & \dots & \hat{q}_p^1 \\ \vdots & & & & \vdots \\ \hat{q}_1^M & \dots & \hat{q}_i^M & \dots & \hat{q}_p^M \end{bmatrix}. \tag{1}$$

   The entries q_i^j and $\hat{q}_i^j$ are pseudo-random numbers drawn from the respective densities.

2. Create $A_B^{(i)}$

$$A_B^{(i)} = \begin{bmatrix} q_1^1 & \dots & \hat{q}_i^1 & \dots & q_p^1 \\ \vdots & & & & \vdots \\ q_1^M & \dots & \hat{q}_i^M & \dots & q_p^M \end{bmatrix} \tag{2}$$

   which is the matrix A except that i^{th} column is taken from B. Similarly, create $B_A^{(i)}$.

3. Create C which is the matrix B appended to matrix A such that

$$C = \begin{bmatrix} A \\ - \\ B \end{bmatrix}. \tag{3}$$

   This matrix is used when estimating the total variance.

4. Compute column vectors $f(A)$, $f(B)$, $f(A_B^{(i)})$ and $f(B_A^{(i)})$ by evaluating the model at input values from the rows of matrices A, B, $A_B^{(i)}$ and $B_A^{(i)}$. Let $f(A)_j$ denote the output computed from the j^{th} row of A. The computation of $f(A)$ and $f(B)$ requires $2M$ model evaluations, whereas the evaluation of $f(A_B^{(i)})$ and $f(B_A^{(i)})$ for $i = 1, \dots, p$ requires $2Mp$ evaluations. The total number of model evaluations is $2M(1+p)$.

5. Estimate the Sobol indices. The first-order Sobol indices are approximated by

$$S_i \approx \frac{\displaystyle\frac{1}{M}\sum_{j=1}^{M}\Big[f(A)_j f(B_A^{(i)})_j - f(A)_j f(B)_j\Big]}{\displaystyle\frac{1}{2M}\sum_{j=1}^{2M} f(C)_j f(C)_j - \mathbb{E}^2[f(C)]} \tag{4}$$

and the total Sobol indices are approximated by

$$S_{Ti} \approx \frac{\frac{1}{2M}\sum_{j=1}^{M}\left[f(A)_j - f(A_B^{(i)})_j\right]^2}{\frac{1}{2M}\sum_{j=1}^{2M} f(C)_j f(C)_j - \mathbb{E}^2[f(C)]}. \tag{5}$$

In the last step, variances are approximated using Monte Carlo approximation. The denominator in (4) and (5) is the approximation for the total variance with $\mathbb{E}(Y^2) \approx \frac{1}{2M}\sum_{j=1}^{2M} f(C)_j f(C)_j$ and $(\mathbb{E}(Y))^2$ approximates the squared expectation of $f(C)$. In (4), the term $\frac{1}{M}\sum_{j=1}^{M} f(A)_j f(B_A^{(i)})_j$ approximates $\mathbb{E}(\mathbb{E}(Y|q_i))^2$. In essence, we are taking the mean of responses when all input parameters are varied except q_i. The effect of q_i is fixed since the i^{th} column is the same in both A and $B_A^{(i)}$.

The second term in (4),

$$\frac{1}{M}\sum_{j=1}^{M} f(A)_j f(B)_j, \tag{6}$$

represents the squared mean, f_0^2, using the identity

$$f_0^2 = \int_{\Gamma^2} f(x)f(x')dxdx'. \tag{7}$$

This approximation is shown in [5] to reduce the loss of accuracy when computing D, compared to

$$f_0^2 \approx \left(\frac{1}{M}\sum_{j=1}^{M} f(A)_j\right)\left(\frac{1}{M}\sum_{j=1}^{M} f(B)_j\right), \tag{8}$$

which is used in the previous versions of the algorithm.

The computation of S_{Ti} follows from the derivations in [1], which uses the approximation

$$\mathbb{E}[\mathrm{var}(Y|q_{\sim i})] \approx \frac{1}{2M}\sum_{j=1}^{M}\left[f(A)_j - f(A_B^{(i)})_j\right]^2 \tag{9}$$

instead of the approximation

$$\mathrm{var}[\mathbb{E}(Y|q_{\sim i})] \approx \frac{1}{M}\sum_{j=1}^{M} f(A)_j f(A_B^{(i)})_j - f_0^2 \tag{10}$$

The comparison of different versions of the algorithm can be found in [3].

References

[1] Jansen, M. J. W., "Analysis of variance designs for model output," *Computer Physics Communications*, 117(1-2), pp. 35-43, 1999.

[2] Saltelli, A., "Making best use of model evaluations to compute sensitivity indices," *Computer Physics Communications*, 145(2), pp. 280–297, 2002.

[3] Saltelli, A. and Annoni, P. and Azzini, I. and Campolongo, F. and Ratto, M. and Tarantola, S., "Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index," *Computer Physics Communications*, 181(2), pp. 259–270, 2010.

[4] Sobol', I. M., "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates," *Mathematics and Computers in Simulation*, 55(1-3), pp. 271–280, 2001.

[5] Sobol', I. M. and Tarantola, S. and Gatelli, D. and Kucherenko, S. S. and Mauntz, W, "Estimating the approximation error when fixing unessential factors in global sensitivity analysis," *Reliability Engineering & System Safety*, 92(7), pp. 957–960, 2007.

[6] Weirs, V. G. and Kamm, J. R. and Swiler, L. P. and Tarantola, S. and Ratto, M. and Adams, B. M. and Rider, W. J. and Eldred, M. S., "Sensitivity analysis techniques applied to a system of hyperbolic conservation laws," *Reliability Engineering & System Safety*, 107, pp. 157–170, 2012.