Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Implementation and evaluation of nonparametric regression procedures for sensitivity analysis of computationally demanding models

Curtis B. Storlie [a,*], Laura P. Swiler [b], Jon C. Helton [c], Cedric J. Sallaberry [b]

[a] University of New Mexico, Department of Mathematics and Statistics, Albuquerque, NM 87131-0001, USA
[b] Sandia National Laboratories, Albuquerque, NM 87185-1388, USA
[c] Arizona State University, Department of Mathematics and Statistics, Tempe, AZ 85287-1804, USA

ARTICLE INFO

Article history:
Received 5 February 2009
Received in revised form
7 May 2009
Accepted 13 May 2009
Available online 27 May 2009

Keywords:
Bootstrap
Confidence intervals
Meta-model
Nonparametric regression
Sensitivity analysis
Surrogate model
Uncertainty analysis
Variance decomposition

ABSTRACT

The analysis of many physical and engineering problems involves running complex computational models (simulation models, computer codes). With problems of this type, it is important to understand the relationships between the input variables (whose values are often imprecisely known) and the output. The goal of sensitivity analysis (SA) is to study this relationship and identify the most significant factors or variables affecting the results of the model. In this presentation, an improvement on existing methods for SA of complex computer models is described for use when the model is too computationally expensive for a standard Monte-Carlo analysis. In these situations, a meta-model or surrogate model can be used to estimate the necessary sensitivity index for each input. A sensitivity index is a measure of the variance in the response that is due to the uncertainty in an input. Most existing approaches to this problem either do not work well with a large number of input variables and/or they ignore the error involved in estimating a sensitivity index. Here, a new approach to sensitivity index estimation using meta-models and bootstrap confidence intervals is described that provides solutions to these drawbacks. Further, an efficient yet effective approach to incorporate this methodology into an actual SA is presented. Several simulated and real examples illustrate the utility of this approach. This framework can be extended to uncertainty analysis as well.

1. Introduction

The analysis of many physical and engineering phenomena involves running complex computational models (computer codes). It is almost universally accepted that the sensitivity analysis (SA) and uncertainty analysis (UA) of these complex models are important and necessary components to overall analyses [1–5]. The purpose of SA is to identify the most significant factors or variables affecting the model predictions. The purpose of UA is to quantify the uncertainty in analysis results due to the uncertainty in the inputs. A computational model that sufficiently represents reality is often very costly in terms of run time. Thus, it is important to be able to characterize model uncertainty and perform SA with a limited number of model runs. In this presentation, we suggest an effective procedure for SA of such expensive computer models using meta-models and variance based sensitivity measures. This approach has several advantages over existing procedures [6–9]: (i) efficient use of computational resources, (ii) effective handling of a very large number of input variables, and (iii) generation of confidence interval (CI) estimates for sensitivity and/or uncertainty measures.

In general, we will consider complex computer models of the form

$$\boldsymbol{y} = f(\boldsymbol{x}) + \boldsymbol{\varepsilon}, \tag{1.1}$$

where $\boldsymbol{y} = (y_1, \ldots, y_q)$ is a vector of outputs, $\boldsymbol{x} = [x_1, x_2, \ldots, x_p]$ is a vector of imprecisely known inputs, and $\boldsymbol{\varepsilon}$ is a vector of errors (usually small) incurred by the numerical method used to solve for $\boldsymbol{y}$. For example, $\boldsymbol{\varepsilon}$ could result from chaotic behavior introduced by a stopping criterion where input configurations arbitrarily close to one another can fail to achieve convergence in the same number of iterations. This type of behavior would produce jumps in the resulting $\boldsymbol{y}$ surfaces, even if the output is known to be continuous in principle. In some cases, the errors may not be additive, but possibly multiplicative in nature. In these cases, the model in Eq. (1.1) would still apply if a log transform were applied to the elements of $\boldsymbol{y}$. Although analyses for real systems almost always involve multiple output variables as indicated above, the following discussions assume that a single real-valued result of the form $y = f(\boldsymbol{x}) + \varepsilon$ is under consideration. We can do this without loss of generality because of some recent work regard to multiple outputs. Higdon et al. [10] and Bayarri et al. [11] use

[*] Corresponding author.
E-mail address: storlie@stat.unm.edu (C.B. Storlie).

0951-8320/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ress.2009.05.007

principle components and wavelet decomposition, respectively, to decompose the multiple outputs (possibly even a whole function of values over time and/or space) into just several components which can be considered as independent outputs. Thus, the model in Eq. (1.1) can be applied directly to each of these components separately. The resulting models can then be converted back to the original $\boldsymbol{y}$ space if desired.

The model f can be quite large and involved (e.g., a system of nonlinear partial differential equations requiring numerical solution or possibly a sequence of complex, linked models as is the case in a probabilistic risk assessment for a nuclear power plant [12] or a performance assessment for a radioactive waste disposal facility [13]). In addition, the vector $\boldsymbol{x}$ of analysis inputs can be of high dimension and complex structure (e.g., several hundred variables, with individual variables corresponding to physical properties of the system under study or perhaps to designators for alternative models).

The uncertainty in each element of $\boldsymbol{x}$ is typically characterized by a probability distribution. Such distributions are intended to numerically capture the existing knowledge about the elements of $\boldsymbol{x}$ and are often developed through an expert review process; see [6,7] for more on the characterization of input variable uncertainty. After the characterization of this uncertainty, a number of approaches to SA are available, including differential analysis, variance decomposition procedures, Monte-Carlo (sampling-based) analysis, and response surface methods [6–8]. Variance decomposition is perhaps the most informative and intuitive means with which to summarize the uncertainty in analysis output resulting from uncertainty in individual input variables. This procedure uses measures such as

$$s_j = \frac{\mathrm{Var}(\mathrm{E}[f(\boldsymbol{x})|x_j])}{\mathrm{Var}(f(\boldsymbol{x}))} \tag{1.2}$$

and

$$T_j = \frac{\mathrm{E}(\mathrm{Var}[f(\boldsymbol{x})|\boldsymbol{x}_{(-j)}])}{\mathrm{Var}(f(\boldsymbol{x}))} = \frac{\mathrm{Var}(f(\boldsymbol{x})) - \mathrm{Var}(\mathrm{E}[f(\boldsymbol{x})|\boldsymbol{x}_{(-j)}])}{\mathrm{Var}(f(\boldsymbol{x}))}, \tag{1.3}$$

where $\boldsymbol{x}_{(-j)} = \{x_1, \ldots, x_{j-1}, x_{j+1}, \ldots, x_p\}$, to quantify this uncertainty. The use of these measures is reviewed in [6,14]. The quantity s_j corresponds to the fraction of the uncertainty in y that can be attributed to x_j alone, while T_j corresponds to the fraction of the uncertainty in y that can be attributed to x_j and its interactions with other variables. The calculation of s_j and T_j requires the evaluation of p-dimensional integrals, which are typically approximated with Monte-Carlo sampling from the joint distribution of $\boldsymbol{x}$. Unfortunately, this is too computationally intensive to be feasible for most complex computer models.

An alternative procedure to the direct evaluation of T_j and similar measures is to use a meta-model (or surrogate) for f to perform the necessary model evaluations [9,15]. A meta-model, denoted $\hat{f}$, is much simpler in form and faster to evaluate than the actual computer model. This approach involves taking a sample of size n from the joint distribution of $\boldsymbol{x}$ (e.g., a simple random sample or Latin hypercube sample [16,17]) and evaluating the actual computer model, f, at each of the n design points. The data can then be used to create a meta-model for f. It is assumed that n is a fairly modest number of model evaluations, but large enough to allow for a flexible meta-model estimation. The most commonly used method for function estimation is to perform a linear regression with a model that is linear in each of the inputs; i.e., the model

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \varepsilon. \tag{1.4}$$

This model has been used with much success for SA when the underlying function is approximately linear. However, it is often the case that the linear regression model in Eq. (1.4) can fail to appropriately identify the effects of the elements of $\boldsymbol{x}$ on y when nonlinear relations are present. Rank regression works very well to identify the strength of relationships between inputs and output in nonlinear situations as long the relationships between inputs and output are approximately monotonic [7,18]. However, rank regression does not provide a meta-model as the resultant regression model does not directly provide useful output predictions at new $\boldsymbol{x}$ locations. In nonlinear situations, nonparametric regression methods can be used to achieve a better approximation than can be obtained with the linear regression model in Eq. (1.4) [15].

In this presentation, we describe several modern nonparametric regression methods and compare their performance in calculating sensitivity measures. We also present a general approach to calculating confidence intervals for these measures. This allows a practitioner to account for the variability (i.e., sampling based error) involved in the assessment of variable importance. This presentation continues the investigation of the use of nonparametric regression procedures in SA initiated in [15] by presenting (i) comparisons of several state-of-the-art meta-models, (ii) more effective and precisely defined sensitivity measures based on these meta-models, (iii) confidence intervals for these measures, and (iv) a general method for fast yet effective sensitivity analysis of complex models.

In Section 2 we describe how to use a flexible meta-model to calculate sensitivity measures and associated confidence intervals. We then discuss some of the more useful nonparametric regression procedures available to fit meta-models in Section 3. A simulation study to illustrate the properties of the proposed methodology is given in Section 4. Section 5 describes an efficient procedure for implementation of the proposed methodology and gives an example of this methodology in practice. Finally, a concluding discussion is given in Section 6.

2. Calculating sensitivity measures

Assume for the following that a design has been generated for the uncertain inputs (either fixed or random) and the model has been evaluated at the design points. In this section we consider the calculation of sensitivity indexes for two cases: (i) A linear and/or rank regression is used to fit this data or (ii) a more flexible modeling procedure is needed.

2.1. Sensitivity measures for linear and rank regression

If the fit from linear or rank regression is adequate, then we recommend using the familiar approach of calculating standardized regression coefficients (SRCs) and partial correlation coefficients (PCCs) or the analog of these quantities for rank data as described in [7,19,20]. The main reasons for this are (i) these quantities are familiar and simple to understand and (ii) they are very fast and easy to calculate.

If the fit from both the linear and rank regressions is inadequate, then we recommend calculating the quantities described in Section 2.2. The definition of an “adequate” fit is of course somewhat arbitrary. We recommend the following rule: if the R^2 value of the model is greater than a cut-off value, then the fit is adequate. In practice, the appropriate cut-off value to use is clearly problem dependent. We discuss this issue further in Section 5.