ZACHARY DEL ROSARIO

MANAGING UNCERTAINTY: A PRIMER

MANAGING UNCERTAINTY – A PRIMER

First compiling, February 2018

Contents

Introduction

William Gibson's masterful short story *The Gernsback Continuum*[1] follows a man haunted by an idea. His protagonist is pursued by *semiotic ghosts*, lingering packets of meaning, there in the form of ray-gun spacemen and ten-engine airships sprung from the mind of legendary science fiction author Hugo Gernsback. This character's dilemma is Gibson's reaction to the starry-eyed view of science fiction in the 1970's – a view stuck in the chrome-plated, streamlined future of the 1930's which could never be.

We as engineers in the early 2000's are haunted by a different semiotic ghost, that of an engineering certainty which never existed. Our tools are predicated on a deterministic, omniscent view of the universe which is woefully at odds with the reality of our occupation. Our traditional approach is to make uncertainty irrelevant – to pile on margin and arbitrary safety factors until we creep away from failure, at the expense of inefficient, expensive designs, and a lack of understanding of the relevant uncertainties.

This book is an attempt at exoricism. Rather than making it irrelevant, we aim to *Manage Uncertainty*.

Dummy citation.[2]

[1] William Gibson. *Burning Chrome*. Orion Publishing Group, 1986

[2] S. Shvartsman. personal communication

Sensitivity Analysis

Motivation

Here at Stanford,[3] we host the Predictive Science Academic Alliance Program II (PSAAP II). This project is motivated by a solar receiver intended for green energy production, with heat transfer enhanced via a particle-laden working fluid. This is a system of enormous complexity, featuring particle-laden turbulent flow with radiative heat transfer. The difficulty arises (in part) due to the complex multiphysics, which necessitates careful modeling for computational tractability. For example, the Stanford PSAAP II simulation codes implement models not just for the turbulence, but also novel models for radiation transport, accounting for distributed particles smaller than the resolved gridsize.

[3] Circa 2018

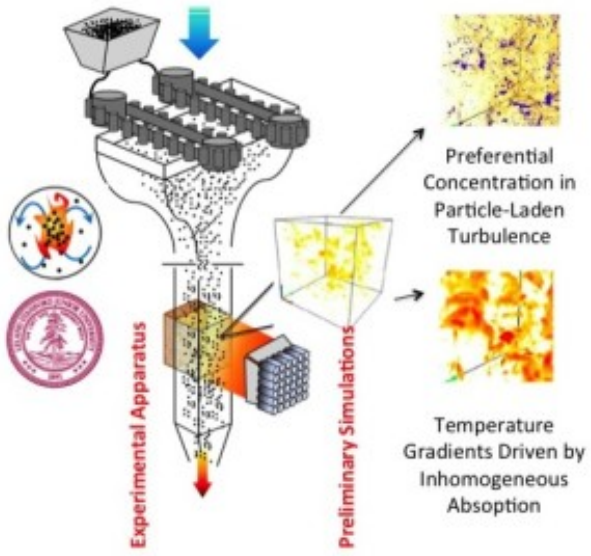


Figure 1: Schematic for the PSAAP II physical problem. This graphic highlights some of the complex interactions between traditionally separate domain physics. The complexity of the physics necessitates modeling assumptions, which introduce uncertainties.

Note that some of the techniques we will discuss are sample-intensive. Most of the techniques described below are intended for *computer experiments*; that is, simulation campaigns run on computer codes. Contrast these with *physical experiments*, in which the physical world 'solves' the equations of motion automatically.

These modeling assumptions introduce non-physical coefficients, which constitute additional uncertainties in the problem. Table 1 lists these and other examples of uncertainties in the PSAAP II problem. For practical purposes, we would be interested in how *all* the uncertainties affect practical output quantities of interest, such as the device efficiency. If the results were largely insensitive to our uncertainties over a range of physically relevant values, this would build confidence in our ability to perform accurate simulations, and would potentially enable the design of novel solar receivers.

Type	Source
Input Uncertainties	Turbulence model parameters
Model Discrepancies	Limitations of eddy viscosity
Numerical Errors	Artificial dissipation
Experimental Uncertainties	Probe accuracy limitations

Table 1: A non-exhaustive list of examples of uncertainties in the Stanford PSAAP II problem.

If our outputs were *not* insensitive to these uncertainties, that would signal that we need to either reduce the uncertainties (say through experimentation), or account for the uncertainties in our results (say through uncertainty propagation). This would necessitate a combination of experimental and computational work – both are expensive activities.

This expense is compunded by the *curse of dimensionality*, which states (roughly) that the expense of a procedure[4] tends to grow *exponentially* with dimension. Not only is gathering data expensive; we need many data points to understand high-dimensional parameter spaces!

[4] e.g. a parameter study, or numerical quadrature

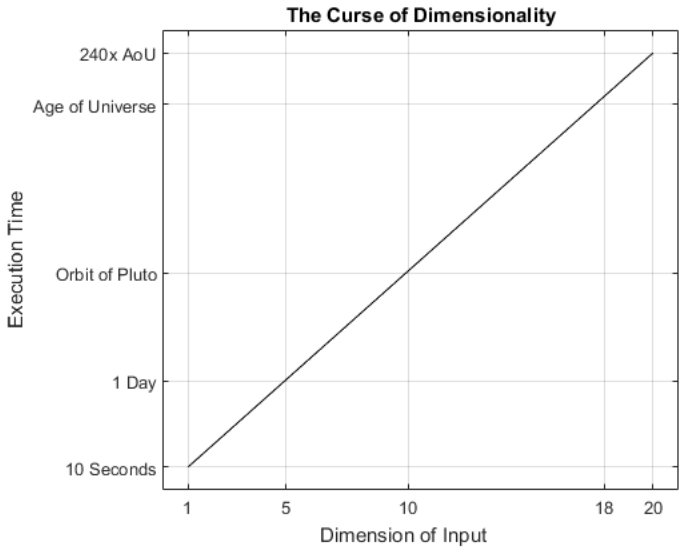


Figure 2: Cartoon depicting computational expense under the curse of dimensionality. Imagine that we are performing quadrature (numerical integration), and are using a simple tensor-product approach; i.e. using 10 evaluations per dimension. If a computer implementation of our qoi evaluates in one second, it will take 10 seconds to study a 1-dimensional function. For a 5-dimensional function, this will take a day to evaluate. This exponential growth quickly grows out of hand, reaching an execution time of nearly the Age of the Universe at just 18 dimensions.

Ideally, one would like to know which parameters are most in-

fluential, preferably before embarking on an expensive simulation campaign. This is the insight that sensitivity analysis seeks to provide. A sensible way to address this curse is to attack dimensionality directly – to reduce the number of input parameters. As a simple approach, we could perform a sensitivity analysis, identify those parameters which do not appreciably affect our qoi, and freeze them to nominal values. This effectively reduces the dimensionality of the problem, ameliorating the curse of dimensionality.

An intermediate conclusion

Sensitivity analysis is about determining how input parameters affect a chosen quantity of interest. We would apply these results at multiple stages in an analysis. A sensitivity analysis could help identify which uncertainties are worth characterizing through a physical experiment. Such an analysis could also identify parameters which can be frozen to nominal values, making further computational experiments less expensive.

Local vs Global

There are two broad philosophies to sensitivity analysis: local and global. Local sensitivity generally refers to studying the gradient of a qoi; the derivative (partial or total) of the output qoi with respect to its input parameters, computed at a set of nominal values. Global sensitivity generally refers to variance contributions; the amount of variance exhibited in the output qoi, as contributed by different sets of the input parameters.

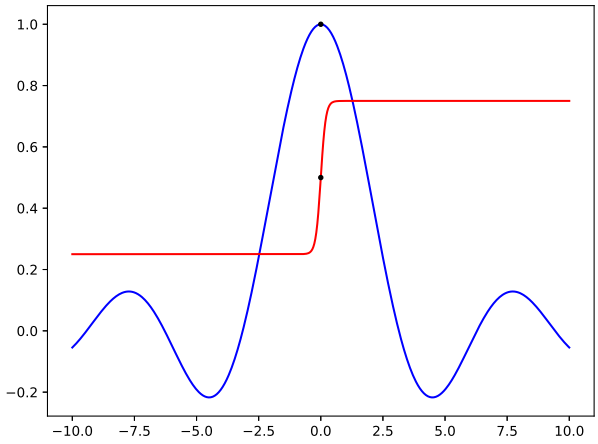


Figure 3: Cartoon functions to depict differences between local and global sensitivity. The vertical axis corresponds to our qoi, while the horizontal corresponds to an input parameter. The black dot is the point of interest for local sensitivity. The blue curve has a great deal of global sensitivity, but zero local sensitivity. The red curve has comparatively little global sensitivity, but a large local sensitivity. Which metric is important depends on the objective of the study.

As implied by the names, the two approaches give different pieces of information. Depending on the context, one or the other may

be the 'right' metric to study. Figure 6 provides a simple cartoon illustration of two cases where local and global sensitivities will be in disagreement.

In what follows, we will focus on local sensitivity, and leave global sensitivity to a following set of notes.

Preliminary Considerations

Before we jump into computing the derivative, we will first discuss some preliminary concerns. Before computing the derivative, we ought to determine *where* we ought to evaluate! Furthermore, we ought to have some clear goal in mind of what to *do* with the gradient before we set out to compute it.

To introduce some notation, let $f(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}$ be our quantity of interest (qoi), and let $\Omega \subseteq \mathbb{R}^d$ be the parameter space of interest. The first object we'll study is Ω.

Region identification

The parameter space Ω may itself be uncertain, or at least poorly characterized. It is possible that insights gained in one subset of Ω may not hold in other regions, or modeling choices which are appropriate in one region may fail in another. Figure 4 depicts a classical example from fluid mechanics where physical behavior changes dramatically in different regions of parameter space. The punchline here is that one should think carefully about the domain Ω to be studied, as this will affect the conclusions drawn from sensitivity analysis, either local or global.

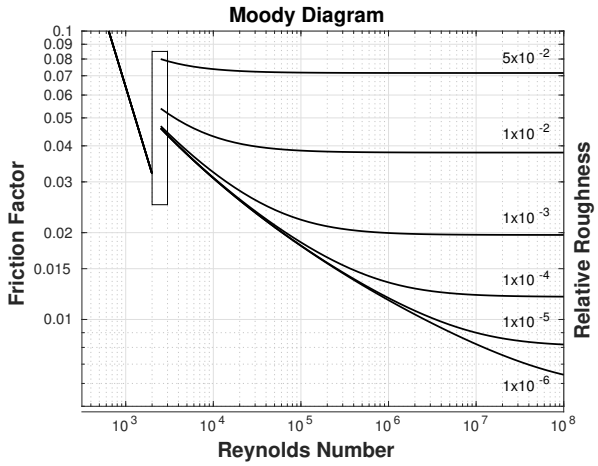


Figure 4: The Moody Diagram depicts the dimensionless pressure losses (Friction Factor) against relevant dimensionless parameters. Note that the Relative Roughness only becomes relevant at high Reynolds Number. This sort of information would not be discovered through a global sensitivity analysis, and instead requires careful probing of local properties.

Pseudoglobal approach

Studying the gradient at every point in parameter space is both expensive and conceptually challenging in high dimensions. Visually inspecting a high dimensional space is not possible, and generally requires some form of dimension reduction for visualization.

One could generate a summary of local behavior by computing an expectation of the gradient, for example

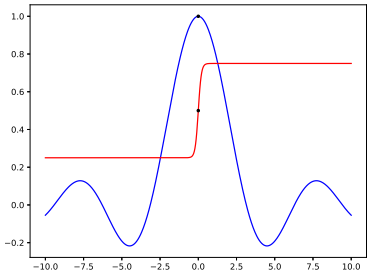
$$\mathbb{E}\left[\left|\frac{\partial f}{\partial x_i}\right|\right], \tag{1}$$

which can be approximated via Monte Carlo sampling.[5] This quantity could be considered a global average of local effects, mixing the two philosophies. The textbook (Ch. 6) describes this as a *pseudoglobal* approach. **??** is closely related to *Morris screening*, a global sensitivity approach we will discuss in the next set of notes, and is also discussed in Chapter 15 of the textbook.[6]

If (1) were small for some set of indices $I \subseteq \{1, \ldots, d\}$, we could freeze the associated variables to nominal values $\boldsymbol{x}_I = \boldsymbol{x}_{I,\text{Nom}}$. This would reduce the dimensionality of the problem. Note that a purely local approach would *not* endorse this pick-and-freeze approach, as illustrated by the blue curve in Figure 6, repeated in the margin. Furthermore, this pseudoglobal approach is prone to miss nonlinear behavior in the function – imagine if we had sampled outside the sharp region of the red curve in Figure 6. A pseudoglobal approach is not necessarily the best way to attack the problem of freezing variables; this is better handled via global sensitivity analysis.

[5] Monte Carlo is a technique for approximating integrals, which can be viewed as expectations against some integral weight, possibly uniform. One proceeds by drawing samples $\boldsymbol{x}_i \sim \rho$ according to the integral weight, and estimating $\mathbb{E}[f(\boldsymbol{x})] \approx \frac{1}{n}\sum f(\boldsymbol{x}_i)$.

[6] Ralph C. Smith. *Uncertainty quantification: theory, implementation, and applications*. SIAM, 2013



Determining the stability of an optimal point

Unconstrained optimization is conceptually straightforward; critical points are found where the gradient of the qoi is zero. Optimal points are then found where the curvature meets an additional criterion.[7]The magnitude of the curvature gives some sense of how sensitive the optimal point is to changes in parameter values.

For constrained optimization, the intuition is slightly different. Rather than having zero gradient, a function must meet a stationarity condition; its gradient must 'balance' that of the constraints.[8] Figure 5 gives a cartoon example of a constrained optimization problem. The punchline is that the gradient of our qoi need not be zero for a constrained optimal point. This is not a big deal if our parameters are known exactly.

However, if our parameters are uncertain, a nonzero gradient provides some information about how much this uncertainty can affect our optimal value. As a more concrete example in an engineering

[7] e.g. for a maximum, we want negative curvature

[8] More formally, the Karush-Kuhn-Tucker (KKT) conditions give a set of first-order necessary conditions for optimality.

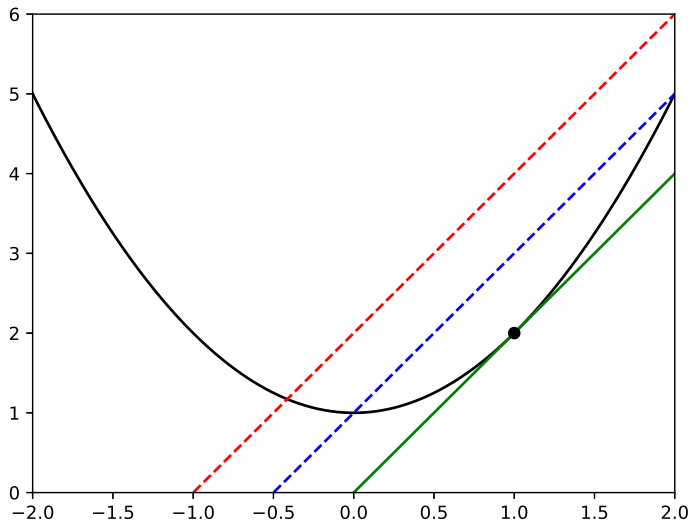


Figure 5: Cartoon of constrained optimization: The black curve is the constraint, while the red, blue, and green lines are isocontours of the qoi. If the gradient of our qoi is not aligned with the gradient of the constraint at their intersection – if they are not *tangent* – then one can find an improved function value by sliding along the constraint. Here, the green isocontour is tangent at the black dot, which corresponds to an optimal point. The gradient of the qoi is nonzero at this point; if there is uncertainty in the input parameters, this could lead to the realized value of the response being significantly less than desired. One may quantify these effects through local sensitivity analysis

design context, one could do a Taylor approximation to determine how much the qoi may fluctuate, then add margin to the design to account for this uncertainty.

Finite Differencing

Conceptually, approximating the derivative is quite simple. One simply chooses a base point $\boldsymbol{x}$ and considers a finite step-size Δx approximation to the derivative[9]

$$\frac{df}{dx_i} \approx \frac{f(x_1, \ldots, x_i + \Delta x, \ldots, x_d) - f(\boldsymbol{x})}{\Delta x}. \tag{2}$$

However, there are some complications with this approach. First, this computational procedure requires additional evaluations, with a cost that grows linearly with dimensionality d. If we require N evaluations of the gradient for a desired procedure (e.g. building a map of the gradient or performing quadrature), then our total cost will be $O(Nd)$.

The other complication is choosing a sensible Δx. Recent work has studied the selection of Δx based on measuring the *empirical noise* of a function; that is, tailoring the step-size based on the observered variability in a computed qoi arising from a computational procedure.[10]

[9] Note that this is the *partial derivative*, which is usually denoted with ∂. We're going to reserve the ∂ symbol for a different operation in these notes.

[10] Jorge J Moré and Stefan M Wild. Estimating derivatives of noisy simulations. *ACM Transactions on Mathematical Software (TOMS)*, 38(3):19, 2012

Adjoint Method

The adjoint is clever method for cheaply computing the sensitivity of a scalar qoi. This is particularly useful if evaluating our qoi requires solving some parameterized governing equation $F(\boldsymbol{y}, \boldsymbol{x})$ for a state variable (e.g. flow field) on which our qoi depends $f(\boldsymbol{y}, \boldsymbol{x})$. Practically, it allows us to compute the total derivative of our qoi $\frac{df}{d\boldsymbol{x}}$ at the expense of one additional solution comparable to $F(\boldsymbol{y}, \boldsymbol{x})$. Steven Johnson [11] has some very nice introductory notes on the topic, which I follow for the next example. He also points to a paper by Cao et al.[12] which gives a more general treatment of the adjoint.

[11] Steven G. Johnson. Notes on adjoint methods for 18.335, 2012

[12] Yang Cao, Shengtai Li, Linda Petzold, and Radu Serban. Adjoint sensitivity analysis for differential-algebraic equations: The adjoint dae system and its numerical solution. *SIAM Journal on Scientific Computing*, 24(3):1076–1089, 2003

Example: Linear System

The spirit of the adjoint method is well-illustrated by a simple linear system. Suppose we have a governing equation for the state $A_{jk}(\boldsymbol{x})y_k = b_j(\boldsymbol{x})$ [13], where our equation is parameterized by $\boldsymbol{x}$. Our qoi is some known function of the state and our parameters $f(\boldsymbol{y}, \boldsymbol{x})$. By chain rule, the sensitivity is

[13] I'm going to use Einstein notation here to help make this easier to follow.

$$\frac{df}{dx_i} = \frac{\partial f}{\partial x_i} + \frac{\partial f}{\partial y_j}\frac{\partial y_j}{\partial x_i}. \tag{3}$$

Since f is known, the partials $\frac{\partial f}{\partial x_i}$ and $\frac{\partial f}{\partial y_j}$ are easy to evaluate.[14] However, the quantity $\frac{\partial y_j}{\partial x_i}$ is more challenging. Using the matrix inverse, we have $y_j = A^{-1}_{jk}b_k$. Taking the partial, we have[15]

[14] Here, we use ∂ to denote a 'direct' derivative. For example, if $f(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=j}^{m} y_j$, then $\frac{\partial f}{\partial x_i} = 0$ and $\frac{\partial f}{\partial y_j} = 1$.

[15] Here, we use the identity $\frac{\partial \boldsymbol{A}^{-1}}{\partial s} = -\boldsymbol{A}^{-1}\frac{\partial \boldsymbol{A}}{\partial s}\boldsymbol{A}^{-1}$. One can derive this by re-arranging $\frac{\partial(\boldsymbol{A}^{-1}\boldsymbol{A})}{\partial s}$.

$$\begin{aligned}
\frac{\partial y_j}{\partial x_i} &= \frac{\partial A^{-1}_{jk}}{\partial x_i}b_k + A^{-1}_{jk}\frac{\partial b_k}{\partial x_i}, \\
&= A^{-1}_{jk}\left(-\frac{\partial A_{kl}}{\partial x_i}A^{-1}_{lp}b_p + \frac{\partial b_k}{\partial x_i}\right), \\
&= A^{-1}_{jk}\left(-\frac{\partial A_{kl}}{\partial x_i}x_l + \frac{\partial b_k}{\partial x_i}\right).
\end{aligned} \tag{4}$$

Substituting into (3) yields

Using Einstein notation makes it clear in (5) that the quantity $\frac{\partial A_{kl}}{\partial x_i}$ is a rank 3 tensor, as there are three indices.

$$\frac{df}{dx_i} = \frac{\partial f}{\partial x_i} + \frac{\partial f}{\partial y_j}A^{-1}_{jk}\left(-\frac{\partial A_{kl}}{\partial x_i}y_l + \frac{\partial b_k}{\partial x_i}\right). \tag{5}$$

One could evaluate (5) by carrying out the matrix multiplication (which would be expensive), or by carrying out the product $\frac{\partial f}{\partial y_j}A^{-1}_{jk} \equiv \lambda_k$. This quantity is the solution of the *adjoint equation*

$$\lambda_k A_{kj} = \frac{\partial f}{\partial y_j}. \tag{6}$$

In summary, to compute the sensitivity via the adjoint method, we need (1) the solution to the governing equation $\boldsymbol{y}$ at our chosen parameter values, (2) the partial derivatives to known functions

$\frac{\partial f}{\partial x_i}$, $\frac{\partial f}{\partial y_j}$, $\frac{\partial A_{kl}}{\partial x_i}$, $\frac{\partial b_k}{\partial x_i}$, and (3) the solution to the adjoint equation λ_k. We then evaluate the total derivative via

$$\frac{df}{dx_i} = \frac{\partial f}{\partial x_i} + \lambda_k A^{-1}_{jk} \left(-\frac{\partial A_{kl}}{\partial x_i} y_l + \frac{\partial b_k}{\partial x_i} \right). \tag{7}$$

Conclusion

This set of notes introduced the motivation behind sensitivity analysis, the broad philosophies of local and global approaches, some uses of sensitivity analysis, and approaches for performing local sensitivity analysis. Next time, we'll dig into techniques for performing global sensitivity analysis.

Appendix

Some useful derivations

Personally, I found some of the claims in Johnson's notes rather mysterious.[16] I attempt to demystify some of those points in this section.

[16] Steven G. Johnson. Notes on adjoint methods for 18.335, 2012

The derivative of a matrix exponential ends up being a convolution. This is because the derivative is more easily carried out in the frequency domain, and we pick up a convolution from doing so. Let $\boldsymbol{A} = \exp(-t\boldsymbol{B})$. First, we'll work out the Laplace transform of this quantity.

$$\mathcal{L}\{\boldsymbol{A}\}(s) = (s\boldsymbol{I} + \boldsymbol{B})^{-1}. \tag{8}$$

Next, we'll take the derivative, using the identify $\frac{\partial \boldsymbol{A}^{-1}}{\partial x} = -\boldsymbol{A}^{-1} \frac{\partial \boldsymbol{A}}{\partial x} \boldsymbol{A}^{-1}$.

$$\frac{\partial}{\partial x}\mathcal{L}\{\boldsymbol{A}\}(s) = -(s\boldsymbol{I} + \boldsymbol{B})^{-1} \frac{\partial \boldsymbol{B}}{\partial x} (s\boldsymbol{I} + \boldsymbol{B})^{-1}. \tag{9}$$

Finally, we'll take an inverse Laplace transform,[17] which yields

[17] Assuming we can commute the derivative with the integral

$$\begin{aligned} \frac{\partial \boldsymbol{A}}{\partial x} &= -\left[e^{-\boldsymbol{B}t} \frac{\partial \boldsymbol{B}}{\partial x} \right] \circledast \left[e^{-\boldsymbol{B}t} \right], \\ &= -\int_0^t e^{-\boldsymbol{B}t'} \frac{\partial \boldsymbol{B}}{\partial x} e^{-\boldsymbol{B}(t-t')} dt' \end{aligned} \tag{10}$$

This enters into the derivative equation via

$$\frac{df}{dx_i}(t) = \frac{\partial f}{\partial x_i}(t) + \lambda_k(t) \left(-\frac{\partial A_{kl}}{\partial x_i}(t) y_l(t) + \frac{\partial y_k}{\partial x_i}(0) \right). \tag{11}$$

In Johnson's notes, he states a form equivalent to

$$\frac{df}{dx_i}(t) = \frac{\partial f}{\partial x_i}(t) + \int_0^t \lambda_k(t-t') \frac{\partial \boldsymbol{B}}{\partial x_i} y_l(t') dt' + \lambda_k(t) \frac{\partial y_k}{\partial x_i}(0). \tag{12}$$

This varies from what one might expect as the 'plug in' result from manipulating (11). However, recall that the matrix exponential acts as a shift operator – it *shifts* initial data forward (or backward) in time to give the solution at a desired time. Commuting terms and applying the shifts results in (12) above.

Global Sensitivity Analysis

Last time, we talked about local sensitivity analysis, both in terms of how we might use the derivative, and how we might go about computing it. While useful for various purposes, local sensitivity fails to capture trends outside a selected neighborhood. This is what global sensitivity seeks to address. Figure 6 illustrates problematic behavior for both local and global approaches.

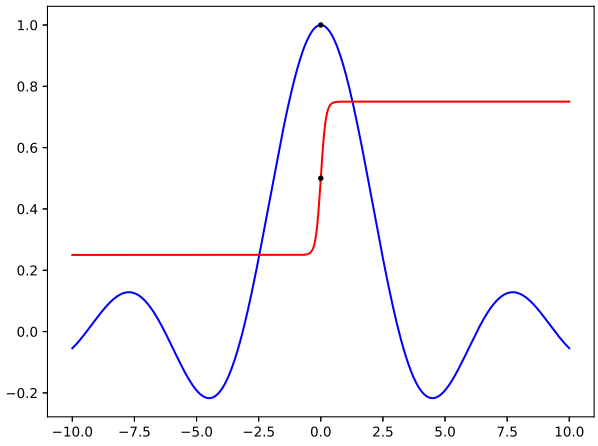


Figure 6: Local sensitivity would fail to capture the behavior of the blue function (in terms of giving zero sensitivity) at the point of interest (black dot), whereas global sensitivity would return a finite value.

Dimension reduction

Why perform global sensitivity analysis? One reason is to perform *dimension reduction*; that is, principled reduction of the dimensionality of a system. Figure 7 depicts the *curse of dimensionality*, which states (roughly) that the computational expense of a procedure tends to grow *exponentially* with the dimension of the problem. The most direct way to address this issue is to reduce dimensionality – to reduce the number of input parameters we need to consider.

Suppose we perform a sensitivity analysis and find that our qoi $f(\boldsymbol{x})$ is relatively unaffected by its input x_1. We could set $x_1 = c$ to a constant nominal value, and continue with $f(c, x_2, \ldots, x_d)$, which has dimension $d - 1$. This would reduce the expense involved with studying our qoi, and since the reduction is *exponential* this could

potentially enable studies previous intractable. Global sensitivity analysis provides principled tools to perform dimension reduction.

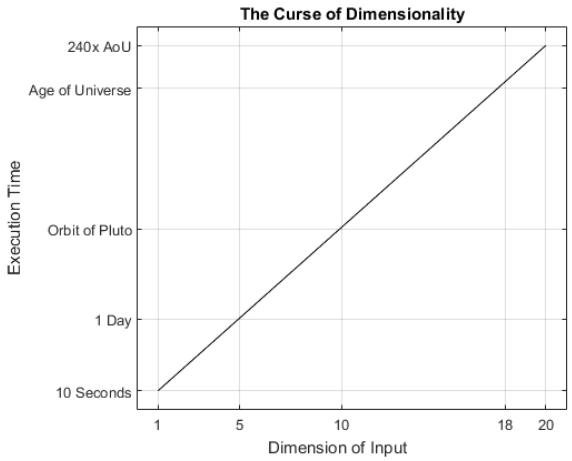


Figure 7: Cartoon depicting computational expense under the Curse of Dimensionality. The most direct way to address this curse is to reduce the effective dimension of our problem. One way to reduce dimensionality is to identify parameters which are unimportant, and freeze them to nominal values. Global sensitivity analysis allows us to determine such unimportant factors.

We'll study three types of global sensitivity analysis:

1. Morris Screening, which seeks to identify unimportant parameters in an economical fashion.

2. Active Subspaces, which identify important *directions* in parameter space.

3. Sobol Indices, which attribute variability in the output to particular parameters.

Parameter interactions

So far in sensitivity analysis, we have only talked about derivatives. For the first derivative, we consider the change in a function as we perturb a single variable. In the case where parameters only affect our qoi linearly, this is sufficient, as effects are additive. However, if there are nonlinear *interactions* between parameters, this presents an issue for this one-at-a-time style of sensitivity analysis. As a concrete example, consider the function

$$f = 0.3x_1 + 8.0x_1x_2, \tag{13}$$

on the domain $x \in [0,1]^2$. Clearly, the mixed term dominates the behavior of the function. The first derivative does not capture this interactive effect, by definition. Capturing interactions of parameters could be accomplished by considering the second derivative, as is done in generalizations of Morris screening. However, the Sobol indices handle this by a different means entirely.

Morris Screening

Morris screening is closely related to the derivative, and can be thought of as studying the mean and variance of the derivative. The textbook [18] goes through this procedure in gory detail; these notes will give only a brief introduction, as we'll focus more on Sobol indices. Morris screening considers statistics of the *elementary effects*, defined by

[18] Ralph C. Smith. *Uncertainty quantification: theory, implementation, and applications*. SIAM, 2013

$$d_i(\boldsymbol{x}_j) = \frac{f(\boldsymbol{x}_j + \Delta \boldsymbol{e}_i) - f(\boldsymbol{x}_j)}{\Delta}, \tag{14}$$

where $\boldsymbol{e}_i$ is the i-th standard basis vector,[19] and Δ is a large stepsize. Note that (14) is simply a first-order approximation of the derivative; however, since Δ is relatively large, this is a coarse approximation to the derivative. These elementary effects are used to construct global sensitivity measures

[19] The vector with all zeros, except for a one in the i-th entry

$$\begin{aligned} \mu_i^* &= \frac{1}{r}\sum_{j=1}^{r} |d_i(\boldsymbol{x}_j)|, \\ \sigma_i^2 &= \frac{1}{r-1}\sum_{j=1}^{r} (d_i(\boldsymbol{x}_j) - \mu_i)^2, \mu_i = \frac{1}{r}\sum_{j=1}^{r} d_i(\boldsymbol{x}_). \end{aligned} \tag{15}$$

Note that considering elementary effects misses interactions, for the reasons discussed above. Generalizations of this technique consider higher-order derivatives, which probe interactions. This procedure is called *screening* because it allows us to determine whether individual parameters are unimportant, but does not quantify relative variable importance. To determine relative importance, we can turn to Sobol indices. However, Morris screening is a relatively cheap procedure, and thus is a good tool to be aware of.

Active Subspaces

The active subspace is a more recent dimension reduction technique, and is somewhat different in spirit from other global sensitivity analysis procedures.[20] Consider the function

[20] Paul G. Constantine. *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*. SIAM Philadelphia, 2015

$$f(\boldsymbol{x}) = \frac{1}{2}(0.3x_1 + 0.7x_2)^2. \tag{16}$$

For this qoi, both its parameters are important. However, the function clearly varies only along the direction $(0.3, 0.7)^T$, and not at all along the direction $(0.7, -0.3)^T$. Rather than picking and freezing individual parameters, the active subspace considers *linear combinations* of the inputs which are important. These correspond to directions in parameter space. Figure 8 depicts the active subspace for (16).

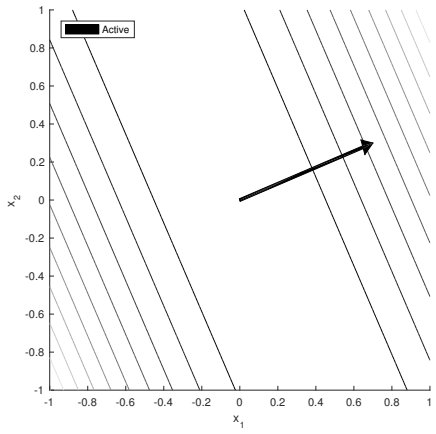


Figure 8: Isolines and active direction for (16)

The active subspace is computed by studying the outer product of the gradient; taking an expectation with respect to the density for $\boldsymbol{x}$, we define

$$\boldsymbol{C} = \mathbb{E}[\nabla_{\boldsymbol{x}} f \nabla_{\boldsymbol{x}} f^T]. \tag{17}$$

Note that $\boldsymbol{C}$ is symmetric semi-positive definite, thus it admits an eigenvalue decomposition $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{\Lambda}\boldsymbol{W}^T$. A basis for the active subspace is identified by separating the magnitude-ordered eigenvalues $\boldsymbol{\Lambda} = \text{Diag}([\boldsymbol{\Lambda}_a, \boldsymbol{\Lambda}_i])$ and their associated eigenvectors $\boldsymbol{W} = [\boldsymbol{W}_a, \boldsymbol{W}_i]$. This split is often chosen based on the relative magnitude of the eigenvalues.

In contrast with Morris screening, we may use the eigenvalues λ_i to rank the associated directions $\boldsymbol{w}_i$. The λ_i are related to the mean-squared directional derivative associated with the appropriate eigenvector. Thus, ranking the directions can only be done in an average sense; the eigenvalues may hide extreme local behavior. Furthermore, the active subspace requires gradient information; this is fine if we access to this information, say through an adjoint solver. If gradients are unavailable, we would need to turn to other approaches, such as finite differences or surrogate modeling, which increases the expense.

Exploiting the active subspace involves a number of challenges, and is both application dependent and an active area of research. To illustrate some of these challenges, suppose we define the *active* and *inactive variables* based on the decomposition above

$$\begin{aligned} \boldsymbol{x}_a &= \boldsymbol{W}_a^T \boldsymbol{x}, \\ \boldsymbol{x}_i &= \boldsymbol{W}_i^T \boldsymbol{x}. \end{aligned} \tag{18}$$

We would like to re-parameterize our qoi in terms of the $\boldsymbol{x}_a$ to reduce dimensionality. However, the naive statement $f(\boldsymbol{x}) = f(\boldsymbol{W}_a \boldsymbol{x}_a)$ is

false! Furthermore, we must determine the domain of $\boldsymbol{x}_a$; this is a projection of the original domain Ω on the subspace $\mathcal{R}(\boldsymbol{W}_a)$, which is in general quite complicated.

Sobol Indices

Both Morris screening and the active subspace consider some average of the gradient. In contrast, Sobol indices are based on *variance*. The Sobol indices may be used to attribute variance of the qoi to subsets of the input parameters. It is a useful, well-studied, and commonly-used technique for global sensitivity analysis, so we will spend some time discussing it.

Sobol indices require a random variable interpretation of our quantity of interest, so we'll introduce some notation used throughout the section. Let $\boldsymbol{X} \sim \rho$ be our random input parameters, distributed according to a joint density ρ over our parameter space Ω.[21] Then our output qoi is also a random variable $Y = f(\boldsymbol{X})$. We denote by $\mathbb{E}[Y] = \int f(\boldsymbol{x})\rho(\boldsymbol{x})d\boldsymbol{x}$ the expectation, and by $\mathrm{V}[Y] = \mathbb{E}[(Y - \mathbb{E}[Y])^2]$ the variance. Note that the following useful identity involving the variance holds

[21] Sobol indices also have some properties that rely on *independence* of the input parameters. We will return to this point later.

$$\mathrm{V}[Y] = \mathbb{E}[Y^2] - \mathbb{E}[Y]^2. \tag{19}$$

We denote conditioning by a vertical bar; this corresponds to holding particular random inputs fixed in value, and leaving them out of the integral for a particular expectation. For example, if $Y = f(X_1, X_2, X_3)$, then

$$\mathbb{E}_{X_2,X_3}[Y|X_1 = x_1] = \int f(x_1, x_2, x_3)\rho(x_1, x_2, x_3)dx_2dx_2, \tag{20}$$

where we use subscripts to make explicit the integration variables. Note that $\mathbb{E}_{X_2,X_3}[Y|X_1]$ is still a random variable[22], due to the variability arising from X_1. We will use conditional variance to decompose the total variance $\mathrm{V}[Y]$ according to different input contributions. This will involve expressions of the sort $\mathbb{E}_{X_2,X_2}[\mathrm{V}_{X_1}[Y|X_1]]$.[23]

[22] when it lacks the equality $X_1 = x_1$

[23] which is no longer a random variable, as we have integrated out all of the parameters

In what follows, we will drop the variable subscripts, as they tend to render expressions indecipherable. The same information is implied by the conditioning; the expression $\mathrm{V}[Y|X_1]$ carries out the integration keeping X_1 fixed, which is then integrated out in the expression $\mathbb{E}[\mathrm{V}[Y|X_1]]$.

Intuition

The Sobol indices are based on a decomposition of the variance of our output qoi.[24] In the next section, we'll look at a formal treat-

[24] This necessitates a random variable interpretation of both our inputs and outputs. In the absence of 'truly' random inputs, one can assign uniform distributions to the uncertain parameters.

ment using the functional analysis of variance. Before that, we'll go through a more simple treatment of the same problem, in order to build intuition.[25]

First, we will introduce some notation to help keep track of variable subsets. Let $\boldsymbol{u}$ be an *index set*; that is $\boldsymbol{u} \subseteq \{1, \dots, d\}$. We will use $\boldsymbol{u}$ to denote subsets of the input variables; in the example above, we used $\boldsymbol{u} = \{2, 3\}$ to give us $\boldsymbol{X}_{\boldsymbol{u}} = \{X_2, X_3\}$. Let $-\boldsymbol{u}$ be the set complementary to $\boldsymbol{u}$ with respect to $\{1, \dots, d\}$; in the example above $-\boldsymbol{u} = \{1\}$.

Next, we will prove a simple identity, which will allow us to attribute the variance $\mathrm{V}[Y]$ to different inputs. Note that for any index set $\boldsymbol{u}$, we have

$$\begin{aligned} \mathbb{E}[\mathrm{V}[Y|\boldsymbol{X}_{\boldsymbol{u}}]] + \mathrm{V}[\mathbb{E}[Y|\boldsymbol{X}_{\boldsymbol{u}}]] &= \mathbb{E}[\mathbb{E}[Y^2|\boldsymbol{X}_{\boldsymbol{u}}] - \cancel{\mathbb{E}[Y|\boldsymbol{X}_{\boldsymbol{u}}]^2}] \\ &\quad + \cancel{\mathbb{E}[\mathbb{E}[Y|\boldsymbol{X}_{\boldsymbol{u}}]^2]} - \mathbb{E}[\mathbb{E}[Y|\boldsymbol{X}_{\boldsymbol{u}}]]^2, \\ &= \mathbb{E}[Y^2] - \mathbb{E}[Y]^2, \\ &= \mathrm{V}[Y^2]. \end{aligned} \tag{21}$$

?? allows us to decompose the variance into two terms *that we can interpret*. First, note that the expression $\mathrm{V}[\mathbb{E}[Y|X_{\boldsymbol{u}}]]$ first averages out all the variables $\boldsymbol{X}_{-\boldsymbol{u}}$, then computes the variance due only to $\boldsymbol{X}_{\boldsymbol{u}}$. We may use this expression with a singleton index set $\boldsymbol{u} = \{i\}$ to define the *first-order sensitivity index*

$$\underline{\tau}^2_{\{i\}} = \frac{\mathrm{V}[\mathbb{E}[Y|X_i]]}{\mathrm{V}[Y]}. \tag{22}$$

?? is bounded between $[0, 1]$, and enables us to rank variables according to their importance. Practically $\underline{\tau}^2_{\{i\}}$ tells us how much variance reduction we could expect if we were able to exactly freeze $\boldsymbol{X}_i$. We could use this information to inform which variables we should better characterize, in order to reduce variability.

Note that we should *not* use $\underline{\tau}^2_{\{i\}}$ to perform dimension reduction; the first-order index does not account for interactions with other variables, so we may miss some important cross terms. Instead, we may consider the *total-order sensitivity index*

$$\overline{\tau}^2_{\{i\}} = \frac{\mathbb{E}[\mathrm{V}[Y|X_{-\{i\}}]]}{\mathrm{V}[Y]} = 1 - \frac{\mathrm{V}[\mathbb{E}[Y|X_{-\{i\}}]]}{\mathrm{V}[Y]}. \tag{23}$$

?? is also bounded between $[0, 1]$, and accounts for interactions between X_i and all other variables. We may interpret (23) in at least two ways: As the variability in Y due to X_i, averaged over all other inputs (middle expression), or as the complement of the variability arising from all the variables excluding X_i (right expression). If $\overline{\tau}^2_{\{i\}}$

[25] The primer by Saltelli et al. is a good, quick read on sensitivity analysis. I follow that text for this subsection.

Andrea Saltelli, Stefano Tarantola, Francesca Campolongo, and Marco Ratto. *Sensitivity analysis in practice: a guide to assessing scientific models*. John Wiley & Sons, 2004

is zero or small, we can be confident that X_i contributes little to the variablility in Y, neither through first-order nor interaction effects.

In summary: The notation for the sensitivity indices is suggestive of their use. The first-order index $\underline{\tau}^2_{\{i\}}$ can be thought of as a *lower* bound; if it is large, then we can be confident that X_i is *important*. In contrast the total-order index $\overline{\tau}^2_{\{i\}}$ is an *upper* bound; if it is small, then we can be confident that X_i is *unimportant*.

The next section gives a more formal treatment of Sobol indices coming from the functional ANOVA decomposition. It's optional reading.

Formulation

Sobol indices are based on the functional analysis of variance (ANOVA) decomposition.[26] The textbook considers only first- and second-order terms; to provide some additional details, we'll follow the notation of Owen [27] and consider all higher-order terms.

We start by assuming the input parameters are statistically independent, and uniform on $\boldsymbol{x} \in [0,1]^d$.[28] The functional ANOVA decomposition is given by

$$f(\boldsymbol{x}) = \sum_{u \subseteq \mathcal{D}} f_u(\boldsymbol{x}), \tag{24}$$

where $u = \{j_1, \ldots, j_{|u|}\}$ is an index set, $|u|$ denotes cardinality,[29] and the f_u are defined recursively by

$$\begin{aligned} f_u(\boldsymbol{x}) &= \int \left(f(\boldsymbol{x}) - \sum_{v \subset u} f_v(\boldsymbol{x}) \right) d\boldsymbol{x}_{-u}, \\ &= \int f(\boldsymbol{x}) d\boldsymbol{x}_{-u} - \sum_{v \subset u} f_v(\boldsymbol{x}), \end{aligned} \tag{25}$$

where $\{v \subset u\}$ denotes all proper subsets v of u, and $-u = u^c$ is the complementary set. To unpack this statement, first note that $f_\varnothing = \int f(\boldsymbol{x}) d\boldsymbol{x} = \mu$ is simply the mean of the function. The singletons $f_{\{i\}}$ are called the *main effects*; these retain variability in a single parameter. For example, for $i = 1$ we have

$$f_{\{1\}}(x_1) = \int f(x_1, x_2, \ldots, x_d) dx_2 \cdots dx_d - f_\varnothing. \tag{26}$$

The functions f_u with $|u| > 1$ describe the *interactions* in isolation of the main effects, which are subtracted off. In this way, each f_u considers a separate component of the variability.

The variance of the qoi is given by $\sigma^2 = \int (f(\boldsymbol{x}) - \mu)^2 d\boldsymbol{x}$, and the ANOVA identity allows us to sum the separate contributions via

$$\sigma^2 = \sum_u \sigma_u^2, \tag{27}$$

[26] For those with a statistics background, the functional ANOVA is a generalization of the multiple-way ANOVA to functions of multiple variables.

[27] Art B. Owen. Variance components and generalized sobol' indices. *SIAM/ASA Journal on Uncertainty Quantification*, 1(1):19–41, 2013

[28] If the parameters are independent and well-behaved, we can map them to a uniform distribution. The assumption of independence is a common, but rather strong assumption. More on this later.

[29] The number of elements in a set.

where $\sigma_u^2 = \int f_u(\boldsymbol{x})^2 d\boldsymbol{x}$ for $u \neq \varnothing$, and $\sigma_\varnothing^2 = 0$. **??** is extremely useful, and enables us to apportion fractions of the variance to different combinations of the input parameters.

Sobol [30] introduced two sensitivity indices, defined by

$$\begin{aligned} \underline{\tau}_u^2 &= \sum_{v \subseteq u} \sigma_v^2, \\ \overline{\tau}_u^2 &= \sum_{v \cap u \neq \varnothing} \sigma_v^2. \end{aligned} \tag{28}$$

[30] I.M. Sobol. Sensitivity estimates for nonlinear mathematical models. *Mathematical Modelling and Computational Experiments*, 1(4):407–414, 1993

The index $\underline{\tau}_u^2$ is called the *closed* sensitivity index, while $\overline{\tau}_u^2$ is called the *total* sensitivity index. In words, the closed index considers the set of parameters u and all its subsets. The total index considers u, and any component which involves this subset in any way. One can show that $\underline{\tau}_u^2 \leq \overline{\tau}_u^2$, which is intuitive given the definition. These indices are usually studied in normalized form $\underline{\tau}_u^2/\sigma^2$ and $\overline{\tau}_u^2/\sigma^2$, which in light of (27) can be interpreted as fractions of the total variance.

These two indices are used in different ways. If the total index $\overline{\tau}_u^2$ is small, it conclusively shows that the entire subset of variables u is unimportant. If the closed index $\underline{\tau}_u^2$ is large, it conclusively shows the set of variables u cannot be neglected.

Computation

The Sobol indices are defined via *expectations* $\mathbb{E}[Y] = \mathbb{E}[f(\boldsymbol{X})] = \int f(\boldsymbol{x})\rho(\boldsymbol{x})d\boldsymbol{x}$; integrals of a quantity against the random variable density. Evaluating these expressions directly requires evaluating a multi-dimensional integral; however, integration is subject to the curse of dimensionality! We started looking at sensitivity analysis to avoid this curse, so we need a different way.

A simple way to approximate an expectation is via the *Monte Carlo* method; this involves drawing independent samples from the distribution $\boldsymbol{x}_i \sim \rho$ for $i = 1, \ldots, n$, and constructing the sample estimate

$$\mathbb{E}[f(\boldsymbol{X})] \approx \frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{x}_i). \tag{29}$$

Naive estimation of the Sobol indices via Monte Carlo can be expensive, as they involve *nested expectations* for a cost of $O(n^2)$. Sobol [31] introduced a trick for a less expensive ($O(2n)$)[32] estimate using *hybrid* points. These hybrid points are denoted by $\boldsymbol{y} = \boldsymbol{x}_u : \boldsymbol{z}_{-u}$, where $y_j = x_j$ for $j \in u$, and $y_j = z_j$ for $j \notin u$. We sample n pairs

[31] I.M. Sobol. Sensitivity estimates for nonlinear mathematical models. *Mathematical Modelling and Computational Experiments*, 1(4):407–414, 1993

[32] Yes, this is formally $O(n)$; I'm just emphasizing the fact that we need to evaluate two points for each sample when using the hybrid point method.

$\boldsymbol{x}_i, \boldsymbol{z}_i \overset{\text{iid}}{\sim} U(0,1)^d$, and approximate

$$\begin{aligned}\hat{\mu} &= \frac{1}{2n}\sum_{i=1}^{n}(f(\boldsymbol{x}_i) + f(\boldsymbol{x}_{i,u} : \boldsymbol{z}_{i,-u})),\\ \underline{\hat{\tau}}_u^2 &= \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i) f(\boldsymbol{x}_{i,u} : \boldsymbol{z}_{i,-u}) - \hat{\mu}^2,\\ \overline{\hat{\tau}}_u^2 &= \frac{1}{n}\sum_{i=1}^{n}(f(\boldsymbol{x}_i) - f(\boldsymbol{x}_{i,-u} : \boldsymbol{z}_{i,u}))^2.\end{aligned} \tag{30}$$

Stochastic codes

Note that in Monte Carlo, we draw samples *randomly*, and average them to form our estimate. This implies that our results are themselves random. Unlike a deterministic code, which will return the same output for the same input, simply re-running a stochastic code with the same input will produce variable outputs; a single output of a stochastic process is called a *realization*. Making statements about a random process based on a single realization is like rolling a die a single time, and using that single number to make statements about the entire random process. Thus, when studying the behavior of a stochastic code, it is not appropriate to study a single realization alone.

As a concrete example, the quantity $\hat{\mu}$ above is a sample estimate of the quantity $\mathbb{E}[f(\boldsymbol{x})]$; we call $\hat{\mu}$ a *point estimate*. By itself, a point estimate gives no notion of how variable our estimate is. To that end, it is standard practice to construct a *confidence interval* (CI) separate from our point estimate. A confidence interval is constructed at a desired *confidence level* (e.g. 95% confidence), and provides a quantitative notion of variability. If our CI is very wide, it demonstrates a large uncertainty about our estimate; conversely, if our CI is very narrow, we are fairly certain about our estimate.[33]

One easy way to construct an *approximate*[34] confidence interval is to make a *normal approximation*. For this, we assume our quantity $f(\boldsymbol{x}) \sim N(\mu, \sigma^2)$ is distributed according to an unknown normal distribution. If this assumption were true, then we would have

$$\frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i) \sim N(\mu, \sigma^2/n). \tag{31}$$

If we knew σ^2 exactly, we could easily construct a 95% confidence interval about our point estimate via $[\hat{\mu} - 1.96\sigma/\sqrt{n}, \hat{\mu} + 1.96\sigma/\sqrt{n}]$.[35] Note that the size of the interval shrinks at a rate $1/\sqrt{n}$; it is for this reason that Monte Carlo is said to exhibit "square-root convergence".

If n is large, we would be justified in using the 'plug-in principle' – taking the variance estimate $s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(f(\boldsymbol{x}) - \hat{\mu})^2$ and

[33] Note that formally, the confidence we have in these approaches is in the *procedures*, and not the *results* themselves. A *method* for constructing a confidence interval has a particular confidence level. An *interval* itself either contains the true value or does not.

[34] The precise definition of a confidence interval is framed in terms of *probabilistic coverage*. An exact CI contains the true value being estimated with probability equal to the confidence level. An approximate interval may be biased, or have a different coverage probability, but should be constructed such that it asymptotically ($n \to \infty$) recovers the exact interval.

[35] The value 1.96 is the approximate value of the 97.5% point of the standard normal distribution; it effectively cuts off 2.5% on the right tail. Since we construct the interval symmetrically about the estimate, this adds up to 5% in the tails to capture the desired 95% confidence.

substituting it for σ^2. However if n is small, this will not be a good approximation. Furthermore, the normal assumption may be quite poor, depending on f and the true distribution of $\boldsymbol{x}$; if we assume $\hat{\overline{\tau}}_u^2$ is normally distributed, this introduces the possibility of the index taking a negative value. This is clearly impossible for the true $\overline{\tau}_u^2$, and CI constructed under the normal assumption at low sample counts could give unrealistic results.

The normal approximation is endorsed by the *central limit theorem*, which (roughly) states that the sum of n independent, identically distributed random variables tends towards a normal distribution as $n \to \infty$.[36] For large sample sizes n, both $\hat{\mu}$ and s^2 will be very nearly normal, endorsing the simple CI procedure above.

[36] With some additional requirements; the iid random variables must have finite mean and variance as well.

As a final note, when studying a simple problem for verification purposes, one can always re-run the procedure to build an ensemble of results.[37] We could compute the variance based on the approximately-normal $\hat{\mu}$ estimates, and construct confidence intervals based on this quantity. However, since we are not re-using samples, but instead drawing new ones for each estimate, this is a more accurate, albeit more expensive, procedure.

[37] One might be tempted to iterate the random seed while re-running the code. This can create some subtle issues. For more notes on numerical paranoia, see Chapter 3 of Art Owen's book on Monte Carlo.

Art B. Owen. *Monte Carlo theory, methods and examples*. 2013

Correlated parameters

The requirement of independent parameters is quite limiting, as many practical problems of interest exhibit correlations between inputs. There exist other sensitivities which overcome these concptual issues; for example, Shapley values. [38] However, computational issues involved with correlated parameters still remain – this is considered a hard problem, and is an active area of research.

[38] Art B. Owen and Clémentine Prieur. On shapley value for measuring importance of dependent inputs. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):986–1002, 2017

Bibliography

[1] Yang Cao, Shengtai Li, Linda Petzold, and Radu Serban. Adjoint sensitivity analysis for differential-algebraic equations: The adjoint dae system and its numerical solution. *SIAM Journal on Scientific Computing*, 24(3):1076–1089, 2003.

[2] Paul G. Constantine. *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*. SIAM Philadelphia, 2015.

[3] William Gibson. *Burning Chrome*. Orion Publishing Group, 1986.

[4] Steven G. Johnson. Notes on adjoint methods for 18.335, 2012.

[5] Jorge J Moré and Stefan M Wild. Estimating derivatives of noisy simulations. *ACM Transactions on Mathematical Software (TOMS)*, 38(3):19, 2012.

[6] Art B. Owen. *Monte Carlo theory, methods and examples*. 2013.

[7] Art B. Owen. Variance components and generalized sobol' indices. *SIAM/ASA Journal on Uncertainty Quantification*, 1(1):19–41, 2013.

[8] Art B. Owen and Clémentine Prieur. On shapley value for measuring importance of dependent inputs. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):986–1002, 2017.

[9] Andrea Saltelli, Stefano Tarantola, Francesca Campolongo, and Marco Ratto. *Sensitivity analysis in practice: a guide to assessing scientific models*. John Wiley & Sons, 2004.

[10] S. Shvartsman. personal communication.

[11] Ralph C. Smith. *Uncertainty quantification: theory, implementation, and applications*. SIAM, 2013.

[12] I.M. Sobol. Sensitivity estimates for nonlinear mathematical models. *Mathematical Modelling and Computational Experiments*, 1(4):407–414, 1993.