Sensitivity Analysis of Models with Input Codependencies

by

Sean Dougherty

A thesis submitted to the
Department of Chemcial Engineering
in conformity with the requirements for
the degree of Master of Applied Science

Queen's University
Kingston, Ontario, Canada
December 2013

For Mom and Dad...

Abstract

Assuming a set of variates are independent and normally distributed is commonplace in statistics. In this thesis, we consider the consequences of these assumptions as they pertain to global sensitivity analysis. We begin by illustrating how the notion of sensitivity becomes distorted in the presence of codependent model inputs. This observation motivates us to develop a new methodology which accommodates for input codependencies. Our methodology can be summarized through three points: First, a new form of sensitivity is presented which performs as well as the classical form but can be obtained at a fraction of the computational cost. Second, we define a measure which quantifies the extent of distortion caused by codependent inputs. The third point is regarding the modelling of said codependencies. The multivariate normal distribution is a natural choice for modelling codependent inputs; however, our methodology uses a copula-based approach instead. Copulas are a contemporary strategy for constructing multivariate distributions whereby the marginal and joint behaviours are treated separately. As a result, a practitioner has more flexibility when modelling inputs.

Table of Contents

List of Tables

List of Figures

List of Nomenclature

Some notation is localized to only a few pages and so it will not be addressed here. The notation which runs throughout this work is displayed in the following list.

$C(\cdot)$	Copula
C	Condition number
d	Number of inputs
$f(\cdot)$	Model
$F_i(\cdot)$	Continuous univariate CDF
i, j, k	Indices
N	Number of model inner loop executions (in MC simulation)
$P(\cdot)$	Proximity function
p_i	Parameter of the proximity function
R	Number of outer loop executions (in MC simulation)
U	Random vector whose elements follow $\mathcal{U}[0, 1]$
V_i	$= \mathrm{Var}\left[\mathrm{E}\left[Y \middle\vert X_i = X_i^*\right]\right]$
V_M	Eigenbasis of matrix M
w_i	Parameter of the proximity function
x_i^*	Conditioning value of X
X	Input vector
Y	Output
Z	Normally distributed random vector
δ_i	MF sensitivity
η_i	VB sensitivity
λ_i	i^{th} eigenvalue of Σ
$\widehat{\lambda}_i$	i^{th} eigenvalue of $\widehat{\Sigma}$
Λ_M	Diagonal matrix of eigenvalues from matrix M
Π	Correlation matrix of Z
Σ	Correlation matrix of X
$\widehat{\Sigma}$	Pseudo-correlation matrix
Σ^*	Salvaged correlation matrix
$\Phi_\Pi(\cdot)$	Standard normal multivariate CDF with correlation matrix Π
$\Phi(\cdot)$	Standard normal univariate CDF
ψ_i	CB sensitivity

CB	Correlation-Based
CDF	Cumulative Density Function
CP	Independence Assumed
LQ	Linear-Quadratic
MC	Monte Carlo
MF	Moment-Free
MLE	Maximum-Likelihood Estimation
PDF	Probability Density Function
RCS	Relative Change in Sensitivity
SA	Sensitivity Analysis
SD	Standard Deviation
UA	Uncertainty Analysis
VB	Variance-Based
WO	Williams-Otto

Chapter 1

Introduction

1.1 Motivation

Mathematical models are ubiquitous in science and engineering. A model relates a set of inputs to an output and this relationship is constructed to represent a system. A model can be used to study a system or make predictions about a system's response to different scenarios. Once a model has been formulated, a critical question is how does this model behave in the presence of uncertainty? Uncertainty Analysis is the practice of propagating input uncertainties through the model such that we gain insight into the uncertainty of the output. Sensitivity Analysis (SA) is a related practice where the uncertainty of an input is attributed to a portion of the output uncertainty. A *sensitivity* is a number assigned to said input which indicates the extent of this apportioning.

SA can be performed either on a local scale or a global scale. A local analysis addresses the model about a nominal set of inputs. This form of analysis is used to study how the output of a model can change. In a global analysis, the entire

space where the inputs can exist must be considered. It follows that global SA gives a practitioner insight into the structure of the model. In this work, we will only consider SA on a global scale.

SA requires that the distribution of the inputs be specified. Traditionally, a practitioner will assume that the inputs are statistically independent. This assumption allows practitioners to arbitrarily specify how each input is distributed. Moreover, several algorithms for the efficient computation of sensitivities are available under this assumption. We will consider the repercussions of relaxing the assumption of independence in this work. In particular, two questions will be investigated. First, how do input codependencies effect the interpretation of sensitivity? Second, how can one model these codependencies? Examples from process engineering will largely form the backdrop of our discussion.

1.2 Organization of the Thesis

Chapter 2: We begin by introducing the idea of sensitivity through a small example. In this example, the effects of uncertain plant parameters on the performance of a feedback control loop are considered. The notion of sensitivity is then formalized by presenting its classical definition. This definition is computationally favourable when inputs are independent. Alternative definitions of sensitivity are addressed; one of which is novel to this work. The effects of input codependencies and model nonlinearities on SA are illustrated using a linear-quadratic model. Finally, the controller-performance model is revisited and a formal SA is conducted. All the sensitivities discussed in this chapter are compared in this analysis.

CHAPTER 3: In this chapter, we address how input codependencies are modelled. The multivariate normal distribution is a natural choice for modelling inputs. However, a practitioner has limited flexibility when this option is used. As an alternative, we promote the use of copulas which is a contemporary method used for constructing multivariate distributions. The flexibility of copulas is illustrated using a bivariate example.

CHAPTER 4: The strategy discussed in the previous chapter requires that a practitioner specify a correlation matrix. This matrix must possess certain properties to ensure the joint behaviour is well-defined. One of these properties is particularly difficult to ensure, especially when there are many inputs. In this chapter, we present an optimization problem based upon an ill-posed correlation matrix which is supplied by the practitioner. Solving this problem will generate a valid correlation matrix. The Williams-Otto Plant is used to show the role of this algorithm in SA

CHAPTER 5: A new methodology is presented which utilizes the novel sensitivity from Chapter 2 and the procedure for modelling codependent inputs from Chapters 3 and 4. This methodology is less computationally expensive than current methods and gives practitioners more latitude with the modelling of inputs. Suggestions for directions of future work are also provided.

Chapter 2

Global Sensitivity Analysis

Consider the model $Y = f(X)$ where $f : \mathbb{R}^d \mapsto \mathbb{R}$ and X is a random vector of length d. Because the input of the model, X, is stochastic, it follows that the output, Y, is also random. Suppose we wish to find the distribution of Y given that X follows a user-specified distribution. The study of this problem is referred to as Uncertainty Analysis (UA). SA is a closely related field. Let us denote the elements of X through $X = [X_1, \ldots, X_d]^T$. SA examines the effect of the uncertainty of X_i on the uncertainty of Y. We will consider a small model ($d = 3$) to compare the difference between these two forms of analysis. This example will be referenced throughout this chapter for illustrating various concepts.

2.1 Motivating Example

Control systems are used extensively for regulating chemical processes. The majority of processes are stable, that is, the system's state naturally remains about some equilibrium point in the absence of disturbances. Disturbances are a major concern in

the operation of chemical processes. Typical disturbances are unexpected changes in feed temperature and composition. When a stable system is subject to a disturbance, its state moves to a different point. Shifts of this nature translate to an inconsistent product and thus lost profits and safety and environmental concerns.

Control systems are implemented to increase robustness. Every process has elements which can be manually adjusted to influence a system's state. Here, theses elements will be referred to as actuators. If these actuators were to remain static, then the disturbances would largely dictate the state of a system. However, if these actuators were dynamic, then they could be manipulated to insulate the system from disturbances. A controller calculates how these actuators should be adjusted.

We will consider a First-Order Plus Dead Time (FOPDT) process which is being regulated by a Proportional-Integral controller. A block diagram representation of this system is shown in Figure 2.1. Note that s is the Laplace variable. The nominal process parameters are $K_p = 0.06$, $\tau_p = 12.57$ and $\theta = 0.5$. The controller parameters are $K_c = 209$ and $K_I = 0.0796$. Let $W(s)$ be the system's current state and let $R(s)$ be the set point of the control loop. We will assume that $R(s) = 0$ and that $R(s) = W(s)$ at the beginning of operation. The control system is subject to a step disturbance, $D(s)$, going from 0 to 0.15. This disturbance propagates through a first-order process (nominally, $K_d = 0.67$) before it is added to $W(s)$. We will not go into the theory which supports this control system. Any introductory control theory text can be consulted for supplemental information, for example [52]. The nominal response of this system is shown in Figure 2.2. We have used w to indicate the state of the process in the time domain.

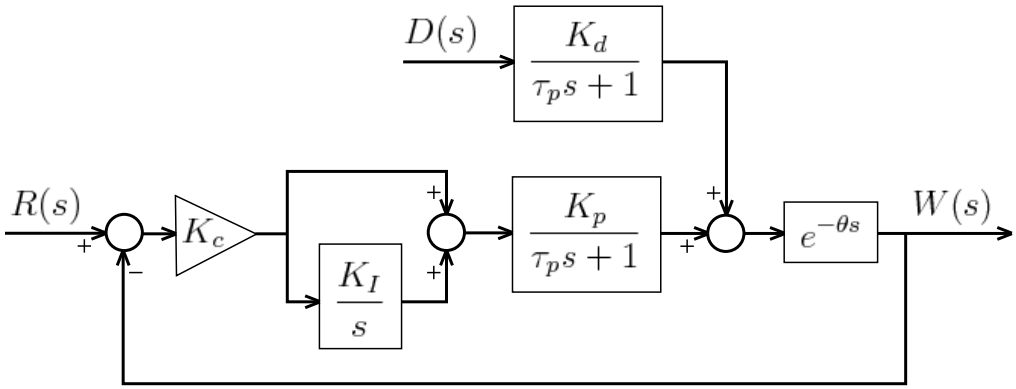

Figure 2.1: Control of a FOPDT process subject to a disturbance

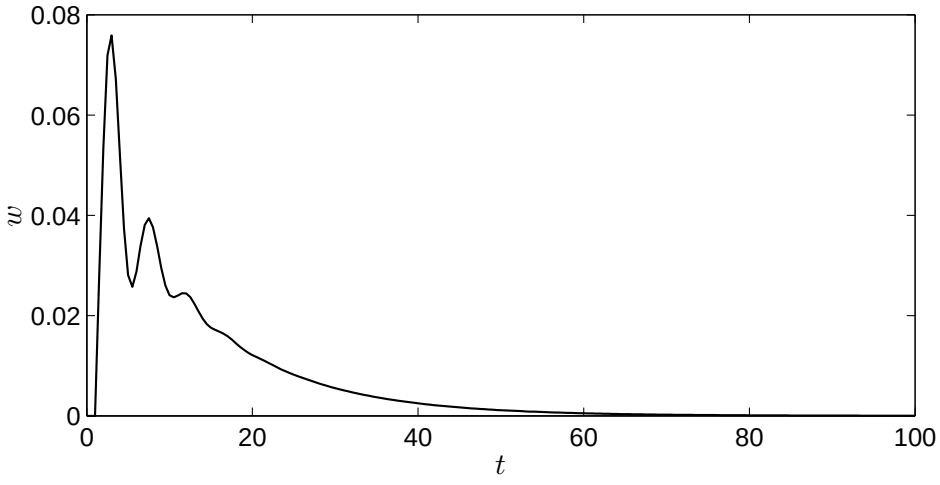

Figure 2.2: Nominal response of the control system

2.1.1 Uncertainty Analysis

Suppose we wish to investigate how the controller performance will vary from day-to-day operation. We must formulate a model to be studied. The parameters which govern the FOPDT process and disturbance dynamics are subject to uncertainty. We will assume that the process gain (K_p), disturbance gain (K_d) and the dead time (θ) are the only sources of this uncertainty. With regards to UA, $X = [K_p \ K_d \ \theta]^T$. For the moment, the inputs are assumed to be independent and normally distributed. For each input, its nominal value is taken to be its mean. The standard deviation is assumed to be 20 percent of the nominal value. The model's output, a controller performance index, will be defined by

$$Y = \int_0^T |w(t)| \ dt \tag{2.1}$$

where T is the time which operation is terminated, $T = 100$. Throughout this work, we will refer to this model as the controller-performance (CP) model.

We are now faced with the challenge of finding the distribution of the output. Clearly, any analytical effort would be fruitless. Monte Carlo (MC) methods are a class of numerical algorithms which propagate uncertainty through nonlinear mappings. A simple MC simulation works as follows. Draw a random sample from the distribution of X. Call this sample $X^{(1)}$. Compute an output which we will call $Y^{(1)}$ from $X^{(1)}$, that is $Y^{(1)} = f(X^{(1)})$. $Y^{(1)}$ constitutes a sample of the output distribution. By iterating this process N times, we can generate a data set for the model's output. The j^{th} sample of the input and output will be denoted by $X^{(j)}$ and $Y^{(j)}$, respectively. This algorithm will be referred to as a *single-loop MC simulation*. In this work, we will not go into much depth regarding MC methods. The monograph of Hammersley and Handscomb can be consulted for supplemental information [23].

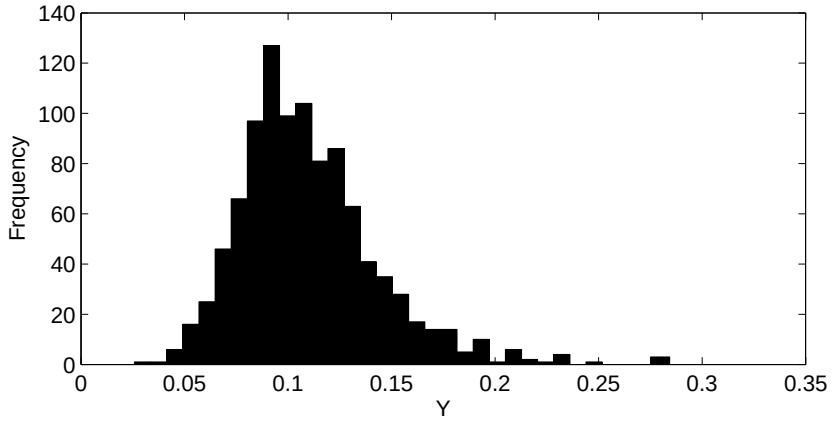

Figure 2.3: Histogram of Y (CP model), mean=0.1098, SD=0.0336

A single-loop MC simulation was run for the controller performance model, $N = 1000$. A histogram can be used to approximate the Probability Density Function (PDF) of the controller's performance. This histogram is displayed in Figure 2.3. Statistical methods, such as Maximum-Likelihood Estimation, could be applied to characterize the output PDF using a parametrized distribution function. Our discussion will mostly rely upon the histogram of the output and its moments.

2.1.2 Preliminary Sensitivity Analysis

UA studies the effect of the collective impact of stochastic inputs on output uncertainty. Suppose we wish to attribute a proportion of output uncertainty to a particular input. This problem is at the crux of SA. How should this task be undertaken? To begin, we will present an intuitive approach. This approach will then be used as a platform to develop some of the more advanced SA methods.

Suppose we wish to examine the impact of X_i's uncertainty on the uncertainty of Y. The MC simulation used for UA drew samples from X. Here, samples from

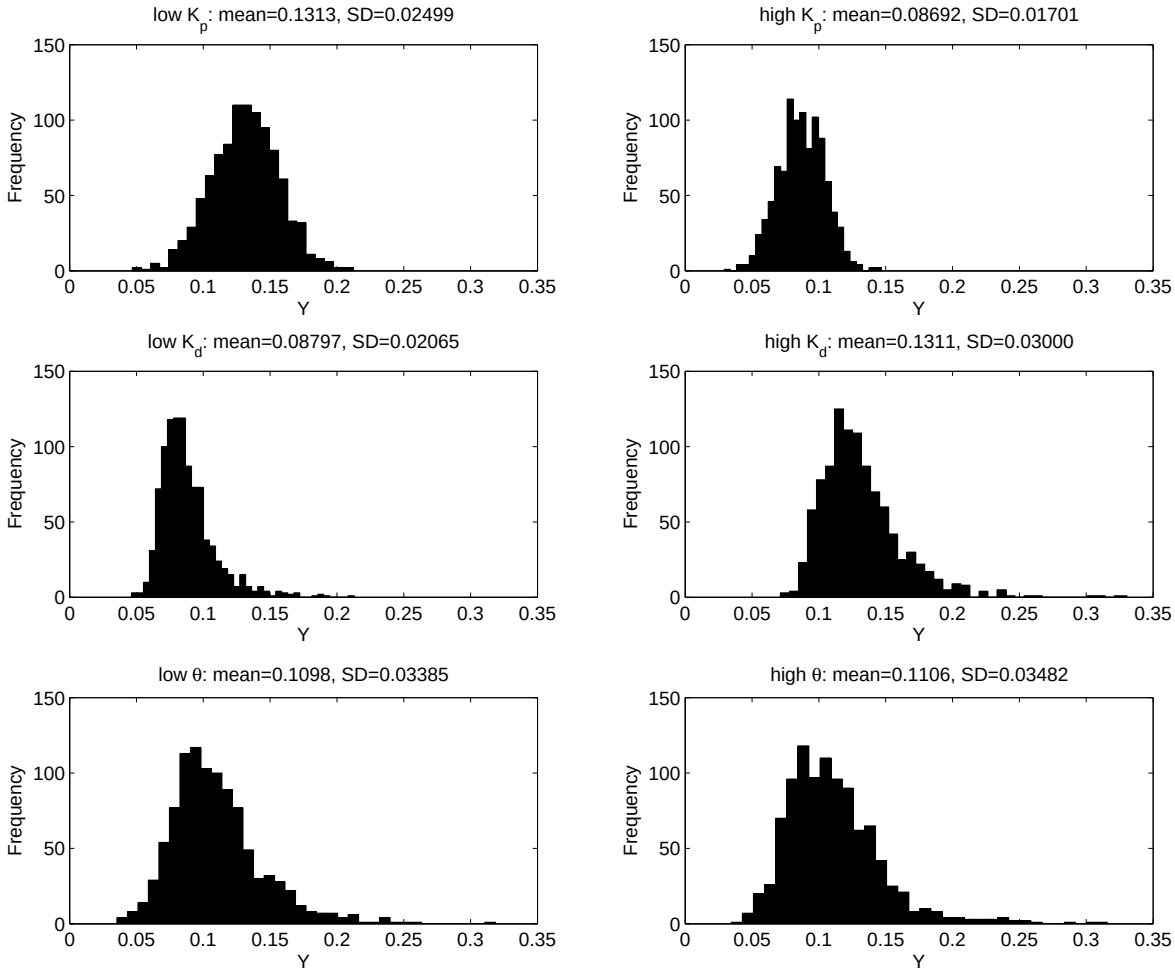

Figure 2.4: Histogram of $Y|X_i = x_i^*$ (CP model)

$X|X_i = x_i^*$ will be used to compute the output distribution using MC simulation. The output is now conditioned on the i^{th} input which is denoted by x_i^*. For illustrative purposes two x_i^*s have been evaluated for each input; a low value and a high value. The low (high) value is the mean of the input less (plus) one standard deviation (SD). The conditioned output distributions are displayed in Figure 2.4 and are accompanied by the sample mean and standard deviation of the output.

Let us compare the unconditioned output distribution, shown in Figure 2.3, with the conditioned distributions shown above. It is clear that the shape of the distribution changes depending upon the input which is being conditioned and at what value. This change in shape is also reflected in the moments of the output. For example,

consider the distributions conditioned on dead time (θ). We see that the mean and standard deviation remain very close to the unconditional mean and standard deviation. This implies that dead time has relatively low impact on controller performance. On the other hand, consider the distributions conditioned on the disturbance gain (K_d). Both the mean and standard deviations are subject to much larger shifts than seen with dead time. The same observation can be made about process gain (K_p). We infer that both gains strongly influence the distribution of controller performance. Alternatively, we say that controller performance is very sensitive to the process and disturbance gains.

Our conclusions have simply been based upon what happens when the output is conditioned on some value of an input. The manner in which the conditioning value has been selected is completely arbitrary. This strategy is flawed. When the output is conditioned on only two input values, we have very limited insight. If we were to use a different set of conditioning values, then potentially different conclusions might be made about the sensitivity of an input. It follows that a large number of conditioning values should be tested. We could generate a histogram for each conditioning value; however, an unwieldy amount of graphical information would be at our disposal. These histograms are generated using samples of the output conditioned on an input. A more practical strategy would be to design a statistic which measures sensitivity and then use these samples in said statistic.

This chapter can be thought of as being divided into two parts. In part one, we discuss three sensitivity measures. The first two measures are commonplace in SA literature; however, their computation is very costly. We also propose a new form of sensitivity which does not suffer from the same computational issues. In part two,

we investigate the effects of input codependencies on SA through several examples.

2.2 Variance-Based Sensitivity

2.2.1 Preliminaries

Variance-Base (VB) Sensitivity is the most discussed sensitivity measure in the literature [62, 63, 64, 67, 65]. We will begin with an intuitive explanation of VB sensitivity. Its formal definition will then follow. Notice that a mean has been calculated for each histogram in Figure 2.4. This is a conditional mean which we will denote by $\mathrm{E}[Y|X_i = x_i^*]$. In its current form, x_i^* is not a random quantity. Recall that we wish to study the output when it is conditioned over a wide array of input values, that is, we consider a large number of x_i^*. Suppose we let x_i^* be a variate with the same distribution as its corresponding element in X. We will denote this variate by X_i^*. Clearly, $\mathrm{E}[Y|X_i = X_i^*]$ is now a random variable. We can now define $V_i = \mathrm{Var}[\mathrm{E}[Y|X_i = X_i^*]]$. A large V_i indicates that the "centre" of the conditional output distribution can exist over a wide range of values. Large values of this quantity can be interpreted as the model is very sensitive to changes in the i^{th} input. This brings us to the accepted definition of VB sensitivity [66, p.161].

$$\eta_i = \frac{V_i}{\mathrm{Var}[Y]} = \frac{\mathrm{Var}[\mathrm{E}[Y|X_i = X_i^*]]}{\mathrm{Var}[Y]}; \quad \eta_i \in [0, 1] \tag{2.2}$$

We implied that V_i could be used as a sensitivity measure but note that V_i simply must be greater than zero. It is natural to ask what would constitute a large V_i? Scaling the sensitivity by $\mathrm{Var}[Y]$ results in it existing on the unit interval.

Sensitivity measures similar to (2.2) have also been proposed [28, 31]. The *ad hoc* nature of these measures limited their adoption by the statistics community [65]. The

benefit of VB sensitivity presented here is apparent when we examine its derivation. Assuming independent inputs, Sobol' showed that the variance of the output could be decomposed into V_i and its higher-order equivalents [71]. Specifically,

$$\mathrm{Var}\left[Y\right] = \sum_i V_i + \sum_i \sum_{j>i} V_{ij} + \cdots + V_{12\ldots d} \tag{2.3}$$

where $V_{ij} = \mathrm{Var}\left[\mathrm{E}\left[Y|X_i = X_i^*, X_j = X_j^*\right]\right] - V_i - V_j$ and so on. Dividing through by $\mathrm{Var}\left[Y\right]$, we obtain

$$\sum_i \eta_i + \sum_i \sum_{j>i} \eta_{ij} + \cdots + \eta_{12\ldots d} = 1 \tag{2.4}$$

Formally, the sensitivity defined in (2.2) is a first-order sensitivity because it pertains to one input. It is now clear that higher-order sensitivities, such as η_{ij}, can be defined as well. Higher-order sensitivities tend to be small relative to first-order sensitivities. Moreover, higher-order sensitivities are often too costly to compute. Hence, analysis is generally based upon only η_i.

Remark 1. *Note that by (2.4) the sum of all sensitivities (both first and higher order) is one. The inputs must be independent for the decomposition in (2.3) to be unique. If input codependencies are present, then this uniqueness is lost and the equality in (2.4) no longer holds.*

2.2.2 Computation of η_i

MC methods are a necessity for SA when the model is non-trivial. A VB sensitivity can not be computed from a single-loop MC simulation. VB sensitivity is based upon the idea of conditioning the output on an input. It follows that a more elaborate sampling scheme must be used. We will refer to this algorithm as a *double-loop MC simulation* and it is structured as follows [72].

1. Draw a sample of X_i^* and call it $X_i^{*(j)}$.
2. Draw a sample from $X|X_i = X_i^{*(j)}$ and call it $X^{(j,k)}$.
3. Evaluate $Y^{(j,k)} = f(X^{(j,k)})$.
4. Repeat Steps 2 and 3 for $k = 1, 2, \ldots, N$.
5. Repeat Steps 1 to 4 for $j = 1, 2, \ldots, R$.

The *inner loop* and *outer loop* are Steps 4 and 5, respectively. Referring back to (2.2), a double-loop MC simulation is required to compute V_i. Note that the algorithm, as described above, will generate the samples necessary to compute only one V_i. If a η_i is to be computed for each input, then d double-loop MC simulations must be run.

We now address the computation of V_i. Let $\Diamond$ denote that we are considering all samples with respect to that index. For example, $Y^{(j,\Diamond)}$ means that we are considering all N samples conditioned on $X_i^{*(j)}$. V_i can be computed as follows.

1. Compute the sample mean of $Y^{(j,\Diamond)}$ and call it $m^{(j)}$.
2. Repeat Step 1 for $j = 1, \ldots, R$.
3. V_i is estimated by the sample variance of $m^{(\Diamond)}$.

The choice of R and N is critical. In our motivating example, the generated output data was visualized in a histogram. If N is small, then the shape of the histogram might change substantially for small changes in N. If N is large, the shape of the histogram is stable and small changes in N will go unnoticed. This same idea must be applied when computing η_i. Note that in the computation of V_i, the sample mean and sample variance are used. Both of these estimators are unbiased and so we are assured that they tend to their population values for large sample sizes. The question becomes how large of a sample size is required?

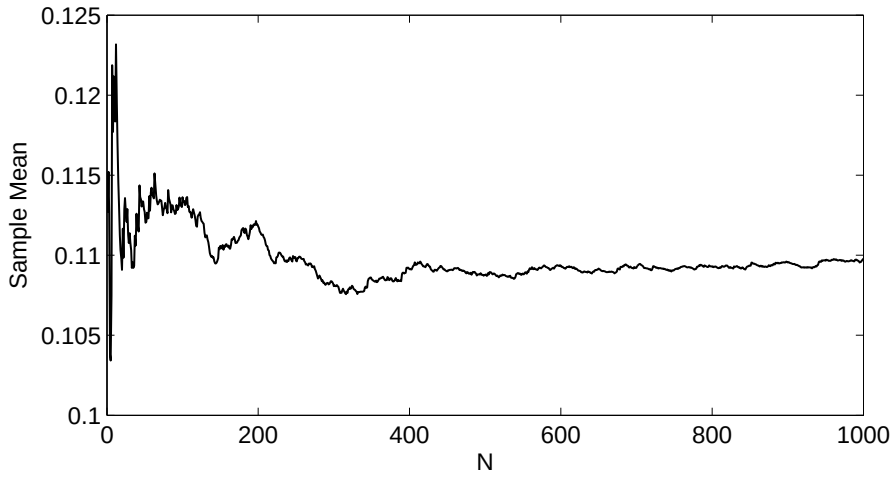

Figure 2.5: Sample mean of Y as a function of N (CP model)

Consider our motivating example. Suppose we are interested in the mean for the controller performance index. Using the data generated for the UA, we can produce Figure 2.5. Note that for $N \geq 500$ the sample mean remains relatively constant. At this point, we say the simulation has *converged* and the sample mean is taken to be the population mean [23]. The population variance found in the denominator of (2.2) can be calculated using this same approach. We have just discussed convergence of a single-loop MC simulation but this logic also holds for the inner loop of a double-loop MC simulation. Once we know that N is sufficiently large, we must turn our attention to the number of outer loop executions, R. Convergence can be tested for using a plot similar to Figure 2.5 but with R on the horizontal axis and the "sample" η_i on the vertical axis.

The double-loop MC simulation is often referred to as the *brute-force method* in SA literature [72]. Under the assumption of independent inputs, alternative strategies have been developed for the computation of η_i. Said strategies also rely upon MC methods but far fewer samples are required. Recall that η_i comes from the variance

Table 2.1: Computational cost of η_i

Method	Output	Cost	Reference
Extended FAST	$\eta_i, \eta_{Ti} \ \ \forall i$	$N \times d$	[67]
Modified Sobol's Method	$\eta_i, \eta_{Ti} \ \ \forall i;\ \eta_{ij} \ \ \forall i, j$	$N \times (d+2)$	[62]
Brute-force	$\eta_i \ \ \forall i$	$N \times R \times d$	[72]

decomposition derived by Sobol'. In this same work, the author introduced an efficient strategy for computing η_i [71]. Three decades prior to this work, Cukier *et al.* developed the Fourier Amplitude Sensitivity Test (FAST), one of the earliest forms of global SA [7, 68]. While their derivations differ substantially, the equivalence Cukier's sensitivity and η_i has been proven [64].

Sobol's Method and FAST are both efficient methods for finding η_i. Recall that higher-order sensitivities are often too costly to compute. Can insight be gained into the magnitude of these higher-order sensitivities? Total sensitivity, which we will denote by η_{Ti}, has been invented for this purpose [27]. For the i^{th} input, η_{Ti} is defined as the sum of all sensitivities which involve said input. For example, take $d = 3$. The total sensitivity of the first input is $\eta_{T1} = \eta_1 + \eta_{12} + \eta_{13} + \eta_{123}$. By this definition $\eta_{Ti} > \eta_i$. If the difference between η_i and η_{Ti} is small, then we can conclude that the higher-order effects are trivial and can be neglected. Sobol's Method and FAST have been extended to support the computation of total sensitivities. Table 2.1 compares the these methods with the brute-force approach.

Clearly, the brute-force method is the least favourable option because only η_i can be computed and at a much higher cost. However, Extended FAST and Modified Sobol's Method require the assumption of independent inputs. If input codependencies exist, then the brute-force approach must be used.

2.3 Moment-Free Sensitivity

2.3.1 Preliminaries

We have established that VB sensitivities are the benchmark in SA literature as a result of their efficient computation when inputs are independent. Input codependencies necessitate the brute-force method which is substantially more costly to run. Given the samples which are generated from the brute-force method, could a "better" sensitivity be defined? When VB sensitivity is used, it is assumed that variance is sufficient to describe uncertainty [63]. This assumption may fail to detect changes such as tail behaviour which occur in the output distribution.

The core idea in SA is to change some aspect of the input distribution and then see how the uncertainty of the output changes. Several authors have offered alternative definitions of sensitivity. These definitions differ in their interpretation of output uncertainty and how the input distribution is changed.

Park and Ahn have proposed that this change in output uncertainty be defined in the context of Information Theory [53]. Two different cases are proposed. In the *base case*, the inputs follow some nominal distribution. In the *sensitivity case*, the inputs follow some other distribution of interest. The output distribution is computed for both cases. The difference in output uncertainty is measured using the Kullback-Liebler divergence, a measure of the change in information between two distributions.

The method of Chun *et al.* also uses the idea of comparing the base and sensitivity cases [5]. The Cumulative Density Function (CDF) of the output will be different for the two cases. The change in the input distribution induces a shift in the output CDF. The authors associate this shift with the change in output uncertainty. Their

definition of sensitivity uses the area between the CDFs of the base case and the sensitivity case.

The drawback to both of these definitions is use of a sensitivity case. It may not always be clear to practitioners as to how to go about formulating a sensitivity case. The benefit of VB sensitivity is that only what amounts to the base case needs to be specified. The changes to the input distribution are made via conditioning.

Borgonovo used this idea of input conditioning along with measuring the shift in output distribution to define Moment-Free (MF) sensitivity [1]. As with VB sensitivity, we will first outline the idea of MF sensitivity using data from our motivating example, the CP model. Its formal definition will follow.

Let us begin by characterizing the unconditioned output distribution. From Figure 2.3, the output appears to follow a log-normal distribution. Maximum-Likelihood Estimation (MLE) was used to calculate the parameters of the distribution. The estimation of the unconditioned PDF is displayed as a solid line in Figure 2.6. We will take the low value of K_d to be the conditioning input. From its histogram in Figure 2.4, we see that it too can be represented by a log-normal distribution. The estimation of the conditioned PDF is displayed as a dashed line in Figure 2.6.

Note that the output PDF shifts when it is conditioned on an input. The extent of this shift can be indicated through the area between the conditioned and unconditioned PDFs (the shaded region of Figure 2.6). Formally, this quantity can be defined by

$$s(x_i^*) = \int |f_Y(y) - f_{Y|X_i=x_i^*}(y)| \, dy \tag{2.5}$$

where f_Y and $f_{Y|X_i=x_i^*}$ are the unconditioned and conditioned PDFs, respectively.

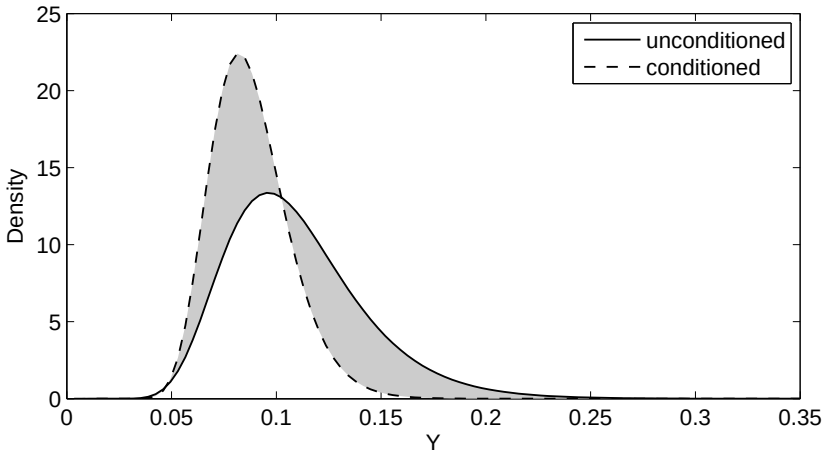

Figure 2.6: The shift in the PDF of Y when conditioned on an input

What value of x_i^* should be used? As was the case in our presentation of VB sensitivity, basing analysis on one x_i^* is ill-advised. The same approach can be used to avoid this problem. We define a random variable X_i^* which follows the same distribution as the i^{th} component of X. Using X_i^* in place of x_i^* in (2.5), $s(X_i^*)$ becomes a random quantity. MF sensitivity is then defined by

$$\delta_i = \frac{1}{2}\mathrm{E}\left[s(X_i^*)\right]; \quad \delta_i \in [0,1] \tag{2.6}$$

From this definition, it is clear where MF sensitivity derives its name. When moments are included in a definition of sensitivity, there is implicit assumption that any changes made to the inputs only influence the output moments which have been included. In the case of VB sensitivity, the changes which arise from conditioning on an input are assumed to only influence $\mathrm{E}\left[Y|X_i = x_i^*\right]$. Naturally, conditioning on an input will give rise to changes in additional moments. The definition of MF sensitivity avoids this issue by considering the output distribution as a whole. An alternative

form of MF sensitivity which uses CDFs in place of PDFs has also been proposed [42]. However, (2.6) will be taken as our definition of MF sensitivity because of its more prominent use.

2.3.2 Computation of δ_i [1]

A single-loop MC simulation is run to generate a vector of unconditioned output samples. We must now characterize the distribution of these samples. Unless otherwise specified, the data set will be fit to a parametric distribution. The protocol which was used is as follows.

1. A parametric distribution was proposed based upon the histogram of the data.
2. MLE was used to estimate the parameters.
3. The Kolmogorov-Smirnov Test [46] was used to confirm that the parametric distribution used was appropriate.

Once these parameters have been estimated, the unconditioned output PDF, f_Y, can be defined.

A double-loop MC simulation must also be run. The samples generated are then processed as follows to compute MF sensitivity. The same indexing strategy as on page 12 is used here.

1. Fit a distribution to $Y^{(j,\Diamond)}$ and call its PDF $\hat{f}_{Y|X_i=X_i^{*(j)}}$.
2. Evaluate $\hat{s}^{(j)} = \int |\hat{f}_Y(y) - \hat{f}_{Y|X_i=X_i^{*(j)}}(y)|\, dy$ using a Riemann sum.
3. Repeat Steps 1 and 2 for $j = 1, \ldots, R$.
4. δ_i is estimated by one half of the sample mean of $\hat{s}^{(\Diamond)}$.

Note that the cost of computing a VB or MF sensitivity take on the same form, $N \times R$. One must not neglect the role convergence plays in MC methods. There exists the potential for one type of sensitivity to require a larger value of N or R to achieve convergence. Plots such as Figure 2.5 should always be constructed to ensure that N and R are sufficiently large.

2.4 A Novel Sensitivity

2.4.1 Motivation

We have discussed two sensitivity measures thus far. The decomposition which VB sensitivity was derived from is based upon the assumption of independent inputs. This decomposition can be utilized to compute VB sensitivities from a relatively small number of model evaluations. If input codependencies exist, the definition of VB sensitivity holds but its computation is much more costly. In particular, a complete analysis would require $N \times R \times d$ model evaluations. MF sensitivity requires the same number of model evaluations. However, because output uncertainty is recognized in terms of PDFs instead of moments, some argue it is a better definition of sensitivity [1].

Recall that the idea of SA is to induce a change in the input distribution and then evaluate the change which manifests in the output uncertainty. VB and MF sensitivities differ in their definitions of output uncertainty but use the same strategy for changing the input distribution. Consider looking at sensitivity in a different light. Begin by ignoring the relationship $Y = f(X)$ and simply think of Y and X_i as two random variables. We proposed that a measure of statistical dependence between

Y and X_i be used for sensitivity. If Y and X_i have a high probability of occurring together, then the model is very sensitivity to its i^{th} input.

This proposition appears to be somewhat arbitrary but an intrinsic relationship between dependence and sensitivity exist. In the seminal work of Kolmogorov [37], a dependence measure known as *correlation ratio* was defined as

$$\Theta_A(B) = \frac{\text{Var}\left[\text{E}\left[B|A = A^*\right]\right]}{\text{Var}\left[B\right]}; \quad \Theta_A(B) \in [0, 1] \tag{2.7}$$

where A and B are random variables. If $\Theta_A(B) = 0$, then A and B are independent. If $\Theta_A(B) = 1$, then $B = g(A)$ where g is a Borel-measurable function. The upper bound on the correlation ratio is stated through the following theorem.

Theorem 1. *[61]* **(Relationship between correlation and correlation ratio)** *If A and B be random variables and the mean and variance of B exist, we have*

$$\Theta_A(B) = \sup_{(g)} \left|\text{Corr}\left[B, g(A)\right]\right|$$

where g runs over all Borel-measurable real functions such that the mean and variance of $g(A)$ exist. Equality stands in the above expression if and only if $g(a) = \beta_1 \text{E}\left[B|A = a\right] + \beta_2$ where $\beta_1 \neq 0$ and β_2 are constants.

When we recall our definition of VB sensitivity shown in (2.2), it is clear that $\eta_i = \Theta_{X_i}(Y)$. Loosely speaking, VB sensitivity can be regarded as how closely Y and X_i resemble a functional relationship.

Remark 2. *Let us now discuss how Theorem 1 relates to SA. Consider the linear model, $Y = \ell^T X$ where $\ell \in \mathbb{R}^d$ and Y and X follow their standard definitions.*

Without loss of generality, we take the conditional expectation with respect to the i^{th} input.

$$\mathrm{E}\left[Y|X_i = x_i^*\right] = \ell_i x_i^* + \sum_{j=1}^{d} \ell_j \mathrm{E}\left[X_j|X_i = x_i^*\right]; \quad i \neq j \tag{2.8}$$

We can rearrange to obtain

$$x_i^* = \frac{1}{\ell_i}\mathrm{E}\left[Y|X_i = x_i^*\right] + \epsilon; \quad \epsilon = \frac{1}{\ell_i}\sum_{i\neq j} \ell_j \mathrm{E}\left[X_j|X_i = x_i^*\right] \tag{2.9}$$

Consider the condition for attaining the upper bound in Theorem 1. If ϵ is not a function of x_i^*, then g is the identity function and $\eta_i = |\mathrm{Corr}\left[Y, X_i\right]|$. Indeed, this is the case when X is independent. This observation has also been made by McKay but its derivation differs from the one presented here [48].

In sum, $|\mathrm{Corr}\left(Y, X_i\right)|$ can be computed in place of η_i for a linear model with independent inputs. Suppose that it is not feasible to analytically derive sensitivity for a linear model and we must rely upon MC methods. How does the cost of η_i and $|\mathrm{Corr}\left[Y, X_i\right]|$ differ? The correlation between an input and the output only requires a single-loop MC simulation (N runs). From Table 2.1, it is clear that the cost of η_i will always be a function of the number of inputs. Hence, using $|\mathrm{Corr}\left[Y, X_i\right]|$ is a much more economical option when d is large.

Recall that correlation measures the degree of linear dependence between two random variables. Clearly, in the case of linear models with independent inputs, Y and X_i are scattered around a line and correlation is sufficient for measuring this relationship. Consider the case where the model contains nonlinearities. Potentially, Y and X_i is no longer scattered around a line. Could correlation be modified to measure nonlinear relationships? A natural modification would be to use Spearman's

correlation in place of Pearson's correlation [22]. Note that Spearman's correlation is a more generalized form of correlation. Its use would assumes that Y and X_i are scattered around a monotonic function and not a line [12]. However, we feel that this assumption is too strong for Spearman's correlation to be used as a sensitivity measure.

The notion of correlation can be generalized further. Instead of using a line or a monotonic function, dependence can be measured with respect to an arbitrary curve in $\mathbb{R}^2$. We will refer to this form of correlation as nonlinear correlation.

2.4.2 Theory of Nonlinear Correlation

The theory of nonlinear correlation was established by Delicado and Smrekar [9]. An overview of the theory will be presented here. Define the curve $c : \mathcal{I} \subseteq \mathbb{R} \mapsto \mathbb{R}^2$ which is at least continuous first-order differentiable and does not intersect with itself. Let c be parametrized by arc length, that is, $||c'(s)|| = 1 \; \forall s \in \mathcal{I}$. This curve will be referred to as a *generating curve*. Define $v(s)$ as a unitary vector field which is orthogonal to c, that is, $c'(s)^T v(s) = 0 \; \forall s \in \mathcal{I}$. The generating curve and this vector field are joined through χ_c where $\chi_c : \mathcal{I} \times \mathbb{R} \mapsto \mathbb{R}^2$ and is of the form $\chi_c(s,t) = c(s) + tv(s)$.

A point in $\mathbb{R}^2$ can be represented using an ordered pair, (s_o,t_o), and χ_c. Now consider this ordered pair to be a bivariate random vector, $[S,T]$, which will be called the *generating random vector*. The PDF of the generating random vector is $h(S,T)$ and has a support of $\mathcal{A}$ where $\mathcal{A} \subseteq \mathcal{I} \times \mathbb{R}$. h is subject to two conditions; $\mathrm{E}[T|S = s] = 0$ and $\mathrm{Var}[S] > \mathrm{Var}[T|S = s] \; \forall \; s \in \mathcal{I}$. We stipulate that $\chi_c : \mathcal{A} \mapsto \chi_c(\mathcal{A})$ is a bijective mapping. The machinery is in place to state the following definition.

Definition 1. *[9] Consider A and B to be random variables. Define this vector*

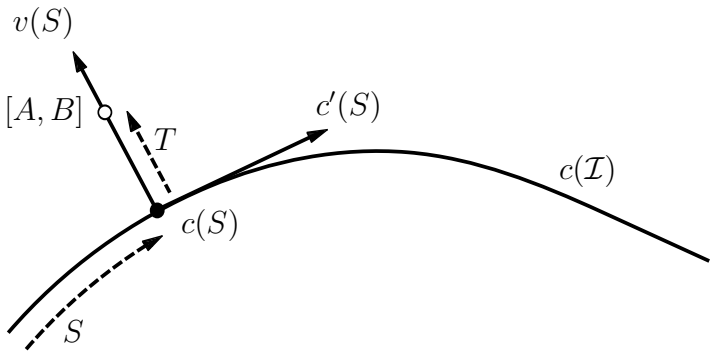

Figure 2.7: A bivariate random vector relative to a generating curve

by $[A, B] = \chi_c(S, T) : \mathcal{A} \mapsto \mathbb{R}^2$. *We say that* $[A, B]$ *is distributed along* $c(\mathcal{I})$. *A visualization of this definition is shown in Figure 2.7.*

We now review some results from Principal Components Analysis (PCA) which will lay the framework for defining nonlinear correlation. The covariance matrix of the random vector, $[A, B]$, will be denoted by Σ. Its spectral decomposition is as follows

$$\Sigma = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}^T \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix} \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}; \quad \xi_1 \geq \xi_2 \tag{2.10}$$

$[\cos\alpha \ \ \sin\alpha]^T$ and $[-\sin\alpha \ \ \cos\alpha]^T$ are the first and second principal components, respectively. α is the angle between the first principal component and the horizontal axis. Without loss of generality, we will assume that the first principal component exists in the first quadrant, that is, $\alpha \in [0, \pi/2]$. ξ_i is the variance of the i^{th} principal component. The variances and covariance of $[A, B]$ can be represented in terms of its

principal components.

$$\begin{aligned} \mathrm{Var}\left[A\right] &= \xi_1 \cos^2\alpha + \xi_2 \sin^2\alpha \\ \mathrm{Var}\left[B\right] &= \xi_1 \sin^2\alpha + \xi_2 \cos^2\alpha \\ \mathrm{Cov}\left[A,B\right] &= (\xi_1 - \xi_2)\cos\alpha \sin\alpha \end{aligned} \tag{2.11}$$

In traditional PCA, the principal components are centred at the mean of the data. Now consider performing PCA around a point on the generating curve, $c(s^*)$ where $s^* \in \mathcal{I}$. At this point, the principal components are scalar multiples of $c'(s^*)$ and $v(s^*)$. When recalling our assumption regarding the generating random vector, $\mathrm{Var}\left[S\right] > \mathrm{Var}\left[T|S=s\right]$, and that $\xi_1 \geq \xi_2$ in traditional PCA, the following definitions should be natural.

Definition 2. *[9] Let $[A,B]$ be distributed along the curve $c(\mathcal{I})$ with $[S,T]$ as the generating random vector. Take $\alpha(s)$ to be the angle between $c'(s)$ and the horizontal axis. Define the local variance of A and B by*

$$\begin{aligned} {\sigma_A^2}^*(s) &= \mathrm{Var}\left[S\right]\cos^2\alpha(s) + \mathrm{Var}\left[T|S=s\right]\sin^2\alpha(s) \\ {\sigma_B^2}^*(s) &= \mathrm{Var}\left[S\right]\sin^2\alpha(s) + \mathrm{Var}\left[T|S=s\right]\cos^2\alpha(s) \end{aligned}$$

Local covariance is defined by

$$\sigma_{AB}^*(s) = \left(\mathrm{Var}\left[S\right] - \mathrm{Var}\left[T|S=s\right]\right)\cos\alpha(s)\sin\alpha(s)$$

Local correlation is defined by

$$\rho^*_{AB}(s) = \frac{\sigma^*_{AB}(s)}{\left(\sigma_A^{2\,*}(s)\sigma_B^{2\,*}(s)\right)^{\frac{1}{2}}}$$

assuming that $\sigma_A^{2\,*}(s)\sigma_B^{2\,*}(s) \neq 0$.

We now define global correlation, which aggregates local correlation through the expectation operator.

Definition 3. *[9] In the context of Definition 2, nonlinear correlation is defined by*

$$\text{NLCorr}[A,B] = \left(\text{E}\left[\left(\rho^*_{AB}(S)\right)^2\right]\right)^{\frac{1}{2}}; \quad \text{NLCorr}[A,B] \in [0,1]$$

provided that Prob $\left\{\sigma_A^{2\,*}(S)\sigma_B^{2\,*}(S) \neq 0\right\} = 1$.

Given its definition, $\rho^*_{AB}(s^*)$ and $c'(s^*)$ will have the same sign. Squaring $\rho^*_{AB}(S)$ circumvents any cancellation that may occur when the expectation is taken. The condition Prob $\left\{\sigma_A^{2\,*}(S)\sigma_B^{2\,*}(S) \neq 0\right\} = 1$ voids any $c(\mathcal{I})$ with horizontal or vertical line segments.

2.4.3 Nonlinear Correlation as a Measure of Sensitivity

Now that the theory of nonlinear correlation has been established, let us return to our original goal. Recall that Theorem 1 shows how η_i is a measure of dependence between Y and X_i. Furthermore, we established the conditions, namely linearity, where $\eta_i = |\text{Corr}[Y, X_i]|$. Often SA is performed on models which are far from linear. Clearly, $|\text{Corr}[Y, X_i]|$ is not appropriate in these situations. We propose measuring the dependence between Y and X_i using nonlinear correlation. The value which is

computed is used to signify sensitivity. Formally, we will define this sensitivity as $\psi_i = \text{NLCorr}[Y, X_i]$ and we will name it Correlation-Based (CB) Sensitivity. Note that like the other forms of sensitivity which we have discussed, ψ_i is also contained on $[0, 1]$.

This new form of sensitivity is advantageous on several fronts. We are not required to condition the input vector and so the brute-force approach is no longer necessary. Furthermore, in light of Remark 3, we are not required to assume the inputs are normally distributed. This point motivates the third chapter of this thesis. Finally, we will soon see that the computation time of CB sensitivity is significantly less than that of VB and MF sensitivity.

2.4.4 Computation of ψ_i

The necessary data can be generated using a single-loop MC simulation. The output samples generated forms one data set. The other data set is the samples of the i^{th} input used to generate said output data. Recall in this new definition of sensitivity, we are simply measuring the dependence between these two data sets.

After reviewing the theory, one will note that the most pressing issue is finding the generating curve. A principal curve is a curve which passes through the middle of a data cloud. Delicado and Smrekar recommend that a principal curve used as the generating curve [9]. In the original work, two methods were demonstrated for fitting principal curves. Of these methods, we have opted to use the algorithm developed by Hastie and Struetzle [24]. Figure 2.8 illustrates the notion of principal curves using data from our motivating example. Delicado has made the code (written in R) for estimating ψ_i publicly available [8]. It should be consulted for supplementary

information. As with the other two forms of sensitivity, a plot similar to Figure 2.5 should be used to ensure that N is sufficiently large.

2.5 Input Codependencies and SA

2.5.1 Past Work

If an analysis is to be based upon VB sensitivities, then the brute-force approach is required when input codependencies are present [72]. Several strategies have been proposed to avoid this path. The same idea underlies all of these strategies. Consider the input vector, X, to have codependencies. Suppose that a transformation called T exists where $T : \mathbb{R}^d \mapsto \mathbb{R}^d$ and for $Z = T(X)$, Z is an independent random vector. Assuming that T is invertible, we can compute the sensitivity with respect to Z where $Y = f(T^{-1}(Z))$. Because Z is independent, the brute-force approach is unnecessary.

Strategies of this nature differ in their construction of T. Jacques *et al.* suggested that inputs be grouped such that the resulting groupings are independent [32]. SA is then performed on said groupings. This approach has limited applicability because a relatively sparse correlation matrix is required for these groupings to be realized. Reilly proposed that PCA be applied to the inputs and the SA be performed on the principal components [60]. In other words, T is the inverse of the eigenbasis of the correlation matrix of X. Mara and Tarantola presented a similar strategy but the Gram-Schmidt process acts as T [45].

The PCA-based and the Gram-Schmidt process-based strategies appear to be viable options for circumventing the brute-force approach. However, the logic behind these strategies is flawed. The goal of SA is to understand how the uncertainty of

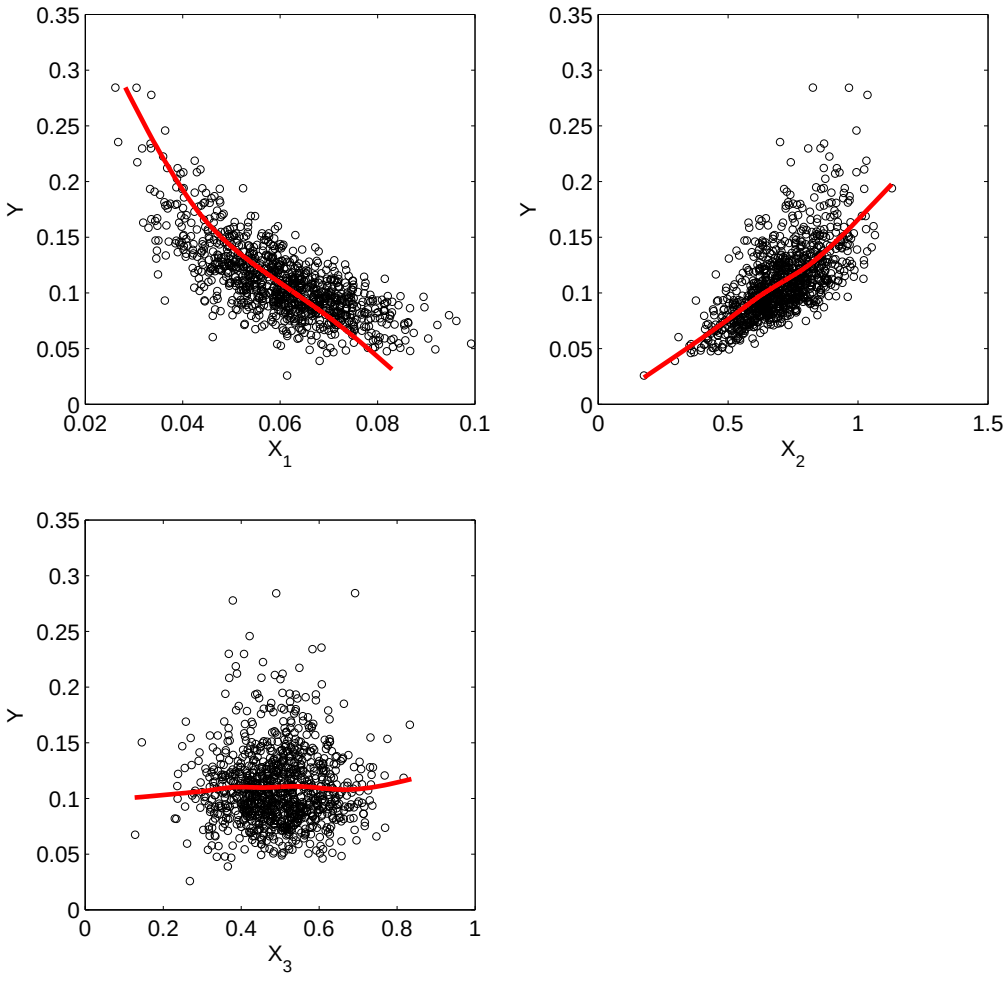

Figure 2.8: Principal curves (marked in red) fitted to data from the CP model

X_i (not Z_i) manifests as uncertainty in Y. Computing the sensitivity with respect to Z_i is simply mathematically convenient. Because no assumptions can be made about the model, it is impossible to relate the sensitivity of X_i to the sensitivity of Z_i. Interpretations of the sensitivity of Z_i accompany the above-mentioned strategies but they are too convoluted to be of much practical use.

An alternative approach is to decompose the sensitivity of a given input into independent and correlated contributions. Xu and Gertner used linear regression to achieve this decomposition [74]. The decomposition proposed by Li *et al.* also relies upon regression but splines are used to give added flexibility [40]. The fundamental problem is that the success of this decomposition is tied to the appropriateness of the regression model that is used. For example, Xu and Gertner's method is valid as long as the relationship between the output and inputs is approximately linear.

We can achieve the same effect as Xu and Gertner and Li *et al.* but in a more straightforward fashion. We draw our inspiration from the very small computational cost of CB sensitivities. Recall that an analysis based on VB/MF sensitivity costs $N \times R \times d$ where as CB sensitivities can be computed from only N runs. Consider computing two sets of sensitivities; one where the inputs are independent and another where input codependencies exist. If the SA were to be based upon CB sensitivities, then $2N$ runs would be required. Note that this computational cost is very small relative to the cost of one VB/MF sensitivity. The following definition will facilitate analysis.

Definition 4. *For a given input, let M be the sensitivity which has been calculated assuming input codependencies exist. For the same input, let M_o be the sensitivity which was calculated under the assumption of independent inputs. We can define*

Relative Change in Sensitivity by

$$\text{RCS} = \frac{M - M_o}{M_o}$$

The type of sensitivity used for M and M_o must be the same. A large RCS *corresponds to an input where codependencies have a significant influence on its sensitivity.*

RCS will be used to assess the CP model at the end of this chapter and then again on the Williams-Otto Plant in Chapter 4. Before addressing these more complex models, let us consider a simple model to gain some intuition into the effects of codependency on SA.

2.5.2 SA of a Linear-Quadratic Model

Consider the Linear-Quadratic (LQ) model

$$Y = \ell^T X + X^T Q X \tag{2.12}$$

where $\ell \in \mathbb{R}^d$, $Q \in \mathbb{R}^{d \times d}$, $Q = Q^T$ and $X \sim \mathcal{N}_d(0, \Sigma)$. We will begin by deriving the VB sensitivity of this model. Recall that the definition of VB sensitivity requires that the output be conditioned on a specific input. If the inputs are independent, then conditioning on one input will not influence any of the other input distributions. However, if the inputs are dependent, the covariance matrix, Σ, ties the inputs together and conditioning must be propagated through a multivariate distribution. The following theorem formalizes this notion.

Theorem 2. *[11, p. 117]* **(Conditioning the Normal Distribution)** *Let $X \sim \mathcal{N}_d(\mu, \Sigma)$. The mean vector is partitioned as $\mu = \begin{bmatrix}\mu_1^T & \mu_2\end{bmatrix}^T$ where $\mu_1 \in \mathbb{R}^{d-1}$ and*

$\mu_2 \in \mathbb{R}$. The covariance matrix is partitioned as

$$\Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{12}^T & \Sigma_{22} \end{bmatrix}$$

where $\Sigma_{11} \in \mathbb{R}^{d-1 \times d-1}$, $\Sigma_{12} \in \mathbb{R}^{d-1}$ and $\Sigma_{22} \in \mathbb{R}$. When X is conditioned on its d^{th} element, the resulting distribution is

$$X|X_d = x_d^* \sim \mathcal{N}_{d-1}\left(\mu_1 + \frac{\Sigma_{12}(x_d^* - \mu_2)}{\Sigma_{22}},\ \Sigma_{11} - \frac{\Sigma_{12}\Sigma_{12}^T}{\Sigma_{22}}\right)$$

Remark 3. *Step 2 of the double-loop MC simulation on page 12 can be facilitated through Theorem 2. The use of this theorem requires that the inputs are assumed to be normally distributed. There exist very few other multivariate distributions which can be conditioned in a similar fashion [18]. For the sake of practicality, we will assume that if VB or MF sensitivities are to be computed in the presence of input codependencies, then the inputs must be assumed to be normally distributed.*

In light of Theorem 2, the VB sensitivity with respect to the d^{th} input can be found. The complete derivation can be found in Appendix A. Recalling that $\eta_i = V_i/\text{Var}[Y]$, the expression derived for η_d can be generalized to the i^{th} input.

$$\text{Var}[Y] = 2\text{trace}\left((Q\Sigma)^2\right) + \ell^T\Sigma\ell \tag{2.13}$$

$$V_i = \left\{\ell_i\sigma_{ii} + \sum_{j=1}^{d}\ell_j\frac{\sigma_{ij}}{\sigma_{ii}}\right\}^2 + \left\{q_{ii}\sigma_{ii}^2 + 2\sum_{j=1}^{d}q_{ij}\sigma_{ij} + \sum_{j=1}^{d}\sum_{k=1}^{d}q_{jk}\frac{\sigma_{ij}\sigma_{ik}}{\sigma_{ii}^2}\right\}^2 \tag{2.14}$$

where q_{ij} is the (i,j) element of Q, $\sigma_{ii}^2 = \text{Var}[X_i]$, $\sigma_{ij} = \text{Cov}[X_i, X_j]$ and $i \neq j \neq k$. Yu and Harris have presented an expression for V_i when the inputs are independent and normally distributed [76]. Note that (2.14) matches the Yu-Harris expression if we let $\sigma_{ij} = 0 \ \forall i, j$ and $i \neq j$.

We will use two case studies to gain insight into the role which input codependencies influence SA. In the first case, we will consider a linear model ($Q = 0$) and compare the three forms of sensitivity we have presented. Analytical expressions for MF and CB sensitivities cannot be derived when the quadratic term is present. Hence, in our second case study, we will investigate the role nonlinearities play in SA using only VB sensitivities.

Case 1: Linear Model

Recall that from Remark 2, $\eta_i = |\text{Corr}[Y, X_i]|$ provided that the model is linear in the inputs and the inputs are independent. If a linear model is subject to input codependencies, by Theorem 1, $\eta_i \leq |\text{Corr}[Y, X_i]|$. Is $|\text{Corr}[Y, X_i]|$ a good approximation of η_i in this case? From (2.13) and (2.14), we know that

$$\eta_i = \frac{1}{\ell^T \Sigma \ell} K^2; \quad K = \ell_i \sigma_{ii} + \sum_{j=1}^{d} \ell_j \frac{\sigma_{ij}}{\sigma_{ii}} \tag{2.15}$$

It is trivial to show that

$$|\text{Corr}[Y, X_i]| = \frac{1}{\ell^T \Sigma \ell} |K| \tag{2.16}$$

We must now identify where K can exist. If $K \in [-1, 1]$, then $\eta_i \leq |\text{Corr}[Y, X_i]|$ and Theorem 1 is satisfied. If $K \notin [-1, 1]$, then $\eta_i > |\text{Corr}[Y, X_i]|$ and Theorem 1 is not satisfied. Therefore, K must exist on $[-1, 1]$. Because $|K|$ and K^2 are relatively close

to one another on this interval, we can consider $\eta_i \approx |\text{Corr}\,[Y, X_i]|$.

What is the CB sensitivity of a linear model? The random vector $[Y, X_i]$ follows a bivariate normal distribution which has a straight line as a generating curve. By Proposition 1 of [9], it was established that under these conditions $\text{NLCorr}\,[Y, X_i] = |\text{Corr}\,[Y, X_i]|$. Hence, CB sensitivity is an approximation of VB sensitivity for linear models.

Let us now investigate how MF sensitivity relates to the other two forms of sensitivity. Suppose that $\ell = [1\ \beta]^T$ where $\beta \in \mathbb{R}$ and that each input follows a standard normal distribution where $\rho = \text{Corr}\,[X_1, X_2]$. Borgonovo *et al.* have presented a closed-form expression for δ_i which applies here [2]. A sensitivity with respect to a given input can be viewed as a function of β and ρ. Figure 2.9 displays the contours of these functions. Note that one set of contour plots has been shown for η_i and ψ_i. This set of plots is actually for η_i. In light of the observation in the previous paragraph, we know that the plots for η_i and ψ_i are approximately the same.

At first glance, the contour plots for η_i/ψ_i and δ_i show a strong resemblance. It follows that a SA based upon any one of these three metrics would likely arrive at the same conclusions. If we were to assume that the inputs are uncorrelated, then the sensitivity of X_2 will always be greater the sensitivity of X_1 provided that $|\beta| > 1$. The contour plots confirm this statement. What happens to the sensitivities as the inputs become more correlated while assuming $|\beta| > 1$? The sensitivities of X_2 remain relatively invariant to changes in ρ. On the other hand, the sensitivities of X_1 increase substantially as the inputs become more correlated. This observation will be further discussed in Case 2.

Upon closer inspection, we see that shape of the two sets of contours are slightly

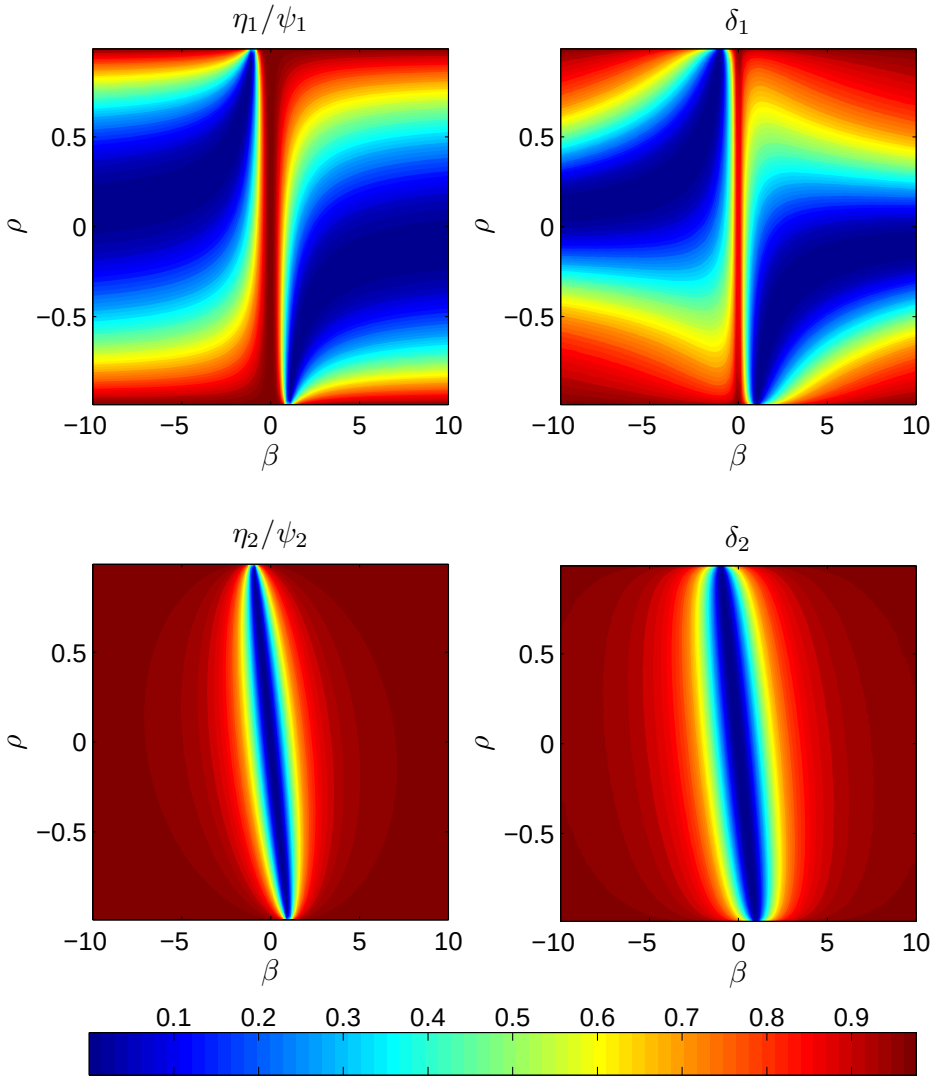

Figure 2.9: Sensitivity contours of a simple linear form

different. Consider the contour plots for the sensitivity of X_1. For a sufficiently large $|\beta|$, we see that as $|\rho|$ approaches one, δ_1 changes more rapidly than η_1/ψ_1. This implies that δ_i is more sensitive to changes in input correlation than η_i/ψ_i. If this observation extends to the all types of models, then we should see a larger RCS (from Definition 4) with respect to δ_i than η_i or ψ_i. The validity of this statement will be addressed in the analyses of the CP model (at the end of this chapter) and the Williams-Otto Plant (in Chapter 4).

Case 2: Bivariate LQ Model

From the previous case study, it is clear that several nice properties arise from a linear model. In this case study, we will investigate if any of these properties are preserved in the presence of nonlinearities. Again, we will assume that the inputs follow a standard normal distribution and $\rho = \text{Corr}\,[X_1, X_2]$. From (2.14), we can show that

$$V_1 = (\ell_1 + \ell_2\rho)^2 + 2(q_{11} + 2q_{12}\rho + q_{22}\rho^2)^2 \tag{2.17}$$

$$V_2 = (\ell_2 + \ell_1\rho)^2 + 2(q_{22} + 2q_{12}\rho + q_{11}\rho^2)^2 \tag{2.18}$$

Note that both V_1 and V_2 are continuous with respect to ρ and

$$\lim_{\rho\to\pm1} V_1 = \lim_{\rho\to\pm1} V_2 \implies \lim_{\rho\to\pm1} \eta_1 = \lim_{\rho\to\pm1} \eta_2 \tag{2.19}$$

Therefore, when the inputs are perfectly correlated $|\eta_1 - \eta_2| = 0$. This statement should be intuitive. Recall that perfect correlation implies $X_2 = \beta_1 X_1 + \beta_o$ where β_o and $\beta_1 \neq 0$ are scalars and $\rho = \text{sgn}\,(\beta_1)$. At this point, the model has only one

stochastic input and both η_1 and η_2 are measuring the sensitivity with respect to this input. By virtue of the continuity of η_i as a function or ρ, we can state that $|\eta_1 - \eta_2|$ will be small when the inputs are highly correlated. Note that this logic can also be applied to the linear model in the previous case study.

We will visualize this concept using numerical examples. Let $\ell = [1\ 3/2]^T$ and

$$Q_1 = \begin{bmatrix} 1 & 1/2 \\ 1/2 & 2/3 \end{bmatrix} \quad \text{and} \quad Q_2 = \begin{bmatrix} 1 & 1/2 \\ 1/2 & 3/2 \end{bmatrix}$$

that is, we are considering two LQ models which differ by the Q matrix which is used. Figure 2.10 displays the VB sensitivities for each model as a function of input correlation. Observe that the loci of η_1 and η_2 are attracted to one another as $|\rho|$ increases and at $\rho = \pm 1$ the loci converge to the same value. One might presume that the attraction between η_1 and η_2 will weaken as the inputs become less correlated. This logic implies that the graphs in Figure 2.10 should be symmetric about $\rho = 0$. Clearly, this is not the case, thus the loci of η_1 and η_2 can experience repulsion as well. It follows that we can not make any generalizations about the sensitivities of two inputs for low and intermediate levels of correlation. However, there is evidence to suggest that the difference between the sensitivities of two inputs will be small if they are highly correlated.

Up to this point, we have thought of sensitivity as a function of input correlation. A more important issue is how does input correlation influence the interpretation of sensitivity? Using Theorem 1, we have offered one interpretation of VB sensitivity which led us to introduce a new sensitivity measure. Here, we offer a second interpretation which facilitates the comparison of two VB sensitivities. We begin by

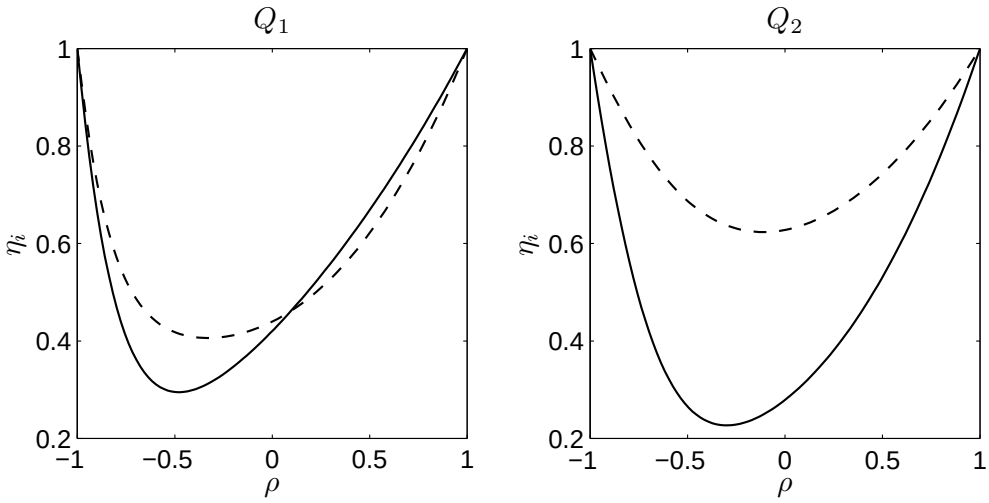

Figure 2.10: η_i as a function of ρ for the two LQ models. The solid line corresponds to η_1 and the dashed line corresponds to η_2.

defining

$$D = \eta_1 - \eta_2 = \frac{V_1 - V_2}{\mathrm{Var}\left[Y\right]} \tag{2.20}$$

By the Law of Total Variance [73, p. 385], $\mathrm{Var}\left[Y\right] = \mathrm{E}\left[\mathrm{Var}\left[Y|X_i = X_i^*\right]\right] - V_i$, we can rearrange to obtain

$$D = \frac{\mathrm{E}\left[\mathrm{Var}\left[Y|X_1 = X_1^*\right] - \mathrm{E}\left[\mathrm{Var}\left[Y|X_2 = X_2^*\right]\right]\right.}{\mathrm{Var}\left[Y\right]} \tag{2.21}$$

Suppose that $D > 0$. This implies that if we were to know the true value of X_1, we would see a greater reduction in the output variance than if we were know the true value of X_2. In other words, X_1 is more sensitive than X_2. $|D|$ indicates the extent of this difference in variance reduction. From Figure 2.10, it is clear that $|D|$ can change substantially with input correlation. For the Q_2 model, the sign of D indicates that X_2 will always be more sensitive than X_1. For the Q_1 model, this statement is not always true because X_1 is more sensitive then X_2 approximately when $\rho \in [0.1, 1]$.

This example demonstrates that accounting for input codependencies is critical.

We will now examine the interplay between input correlation and the nonlinear term in the model. Let us define the ratio of sensitivities by

$$\Gamma = \frac{\eta_1}{\eta_2} = \frac{V_1}{V_2} \tag{2.22}$$

If we were to only consider the linear component of the model, Γ_ℓ, then

$$\Gamma_\ell = \left(\frac{\ell_1 + \ell_2 \rho}{\ell_2 + \ell_1 \rho} \right)^2 \tag{2.23}$$

We can define a similar term for the quadratic component of the model.

$$\Gamma_Q = \left(\frac{q_{11} + 2q_{12}\rho + q_{22}\rho^2}{q_{22} + 2q_{12}\rho + q_{11}\rho^2} \right)^2 \tag{2.24}$$

We can now express Γ as a convex combination of Γ_ℓ and Γ_Q.

$$\Gamma = \alpha\Gamma_\ell + (1-\alpha)\Gamma_Q; \quad \alpha \in [0,1] \tag{2.25}$$

If the mixing parameter, α, is close to zero, then the quadratic term has significant influence on the SA. If α is close to one, then the linear term dominates. Figure 2.11 displays α as a function of ρ for the two LQ models. Two points can be made regarding this figure.

First, we see that the mixing parameter for the Q_1 model is always greater than the Q_2 model. Note that the only difference between the Q_1 and Q_2 matrices is that the $(2,2)$ element has been inverted. Because this element is larger in the Q_2 matrix, the quadratic term has a greater effect on the model output relative to the linear

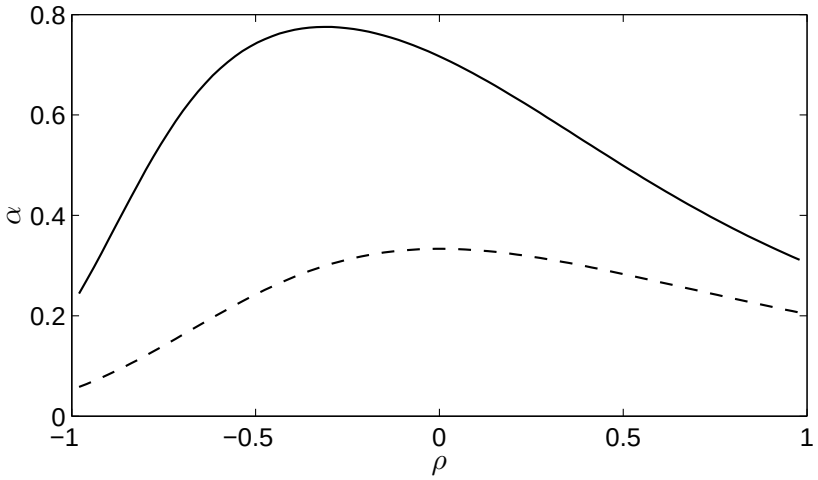

Figure 2.11: The role of the quadratic term in the LQ model. The solid line corresponds to the Q_1 model. The dashed line corresponds to the Q_2 model.

term. Hence, the mixing parameter for the Q_2 model will always remain relatively small.

Second, we see a substantial departure from linearity when the inputs are highly correlated. In the previous case study, we showed that $\eta_i \approx |\text{Corr}\,[Y, X_i]|$ for linear models. This approximation can no longer be considered to be valid. This discrepancy between η_i and $|\text{Corr}\,[Y, X_i]|$ is a result of $[Y, X_i]$ no longer following a bivariate normal distribution. We believe that the generality of nonlinear correlation should allow for CB sensitivity to perform well when inputs are highly correlated. By perform well, we mean that the conclusions drawn from the SA based upon ψ_i should mirror those of η_i and δ_i. We will now test this statement by performing a formal SA on our motivating example.

2.5.3 SA of the Controller Performance Model

At the beginning of this chapter, we introduced the idea of SA using the CP model. We performed a preliminary SA analysis which was simply based around our intuition. Assuming independent inputs, we concluded that the plant gains (K_p and K_d) largely dictate the controller performance while dead time (θ) has minimal influence. Let us see if these initial findings concur with the more reformed definitions of sensitivity.

A double-loop MC simulation was run to generate the samples required for VB and MF sensitivities ($N = 1000$ and $R = 400$). The same N was used for the single-loop MC simulation which generated the samples required for CB sensitivity. The output was deemed to be log-normally distributed for the computation of MF sensitivities. Table 2.2 displays the computed sensitivities. For each form of sensitivity, there is an Independence Assumed (IA) column and a RCS (from Definition 4) column.

We will begin by considering the case of independent inputs. It is incorrect to compare η_i, δ_i and ψ_i. Each form of sensitivity is scaled differently. By Remark 1, we know that the sum of the IA column for VB sensitivities must be less than or equal to one. Furthermore, the sum of all η_i is 0.9776 which suggests that the higher-order effects are small. The definitions of MF and CB sensitivities means that the sum of their IA columns must be less than or equal to d, which is three in this case. All three forms of sensitivity assign large values for K_p and K_d and small values for θ. The same conclusions are reached, regardless of the type of sensitivity used. And so the three types of sensitivity are equivalent from a diagnostic standpoint. Note that these findings are in agreement with the preliminary analysis from the beginning of this chapter.

Table 2.2: Sensitivities of the CP model

	η_i		δ_i		ψ_i	
Input	IA	RCS	IA	RCS	IA	RCS
K_p	0.5751	-0.0136	0.3023	0.0458	0.7486	-0.0527
K_d	0.4015	-0.1623	0.2928	0.0351	0.6771	-0.0622
θ	0.0010	238.8164	0.0215	9.6644	0.0463	9.7297

Are the sensitivities we have computed consistent from a control systems perspective? We will use a frequency domain approach to gain insight into the stability of the system. In short, we will examine what happens to the output signal when a sinusoidal signal is applied to the nominal system. Again, for the sake of brevity, we will not go into depth explaining the underlying theory here. A supplemental reference, such as [52], can be consulted for details. Our analysis will hinge upon two quantities; phase margin (PM) and gain margin(GM). A small phase (gain) margin indicates that only a small amount of dead time (gain) can be added to the process before it is destabilized. A rule of thumb is that systems where $30^o \geq \text{PM} \geq 60^o$ and GM ≥ 6 dB yield satisfactory performance [52, p.565].

Referring back to Figure 2.1, suppose that the sinusoidal signal is applied at $R(s)$ and $D(s) = 0$. We find that PM $= 61.4^o$ and GM $= 9.94$ dB and correspond to θ and K_p, respectively. Hence, we deem the nominal performance of the system to be satisfactory. We see that K_p can be increased only a small amount before the system will perform poorly while θ has more leeway. Now consider this system in a stochastic sense; that is, as it was posed in Chapter 2. We could obtain a GM which results in poor system performance. A significant departure from the nominal dead time (which is unlikely) would be required to result in poor performance. Therefore, controller performance is more sensitive to K_p than θ which concurs with the findings

of our SA.

Making a statement about K_d is more challenging. The use of PM and GM is no longer appropriate. Instead of viewing K_d as a gain, view it as the magnitude of the disturbance entering the system. A large disturbance will result in a large performance index; that is, K_d directly influences the output of the model. Therefore, controller performance is very sensitive to K_d which is reflected in the sensitivities of Table 2.2.

We will now consider the effects of a codependency in the inputs on SA. Again, we will assume that the inputs are normally distributed but we also specify that $\text{Corr}\,[K_d, \theta] = 0.8$. The same set of MC simulations was run again but using the new input distribution. The RCS for each input and type of sensitivity is displayed in Table 2.2. Rerunning the simulation for when input correlation exists is impractical in the case of VB and MF sensitivities. However, here the simulations have been run so that we can compare the RCSs for the three types of sensitivity.

Recall that RCS measures the extent to which input codependencies induce a change in sensitivity. We see that the RCS of K_p and K_d is relatively small for all forms of sensitivity. On the other hand, θ has a large RCS in all three cases. In the case studies, we demonstrated that when two inputs are highly correlated, the difference in sensitivity between the two inputs will be small. Because the model is already very sensitive to K_d, input correlation caused the sensitivity with respect to θ to be large. Notice that the RCS for the VB sensitivity is substantially larger than the other two forms of sensitivity. This is evidence that VB sensitivity is more sensitive to changes in correlation. In the case study of the linear model, we stated that the MF sensitivity was the most sensitive to changes in correlation. This contradiction illustrates that one type of sensitivity can behave differently depending on the model

being analysed.

As with the independent case, we must consider RCS from a control systems perspective. In the independent case, we showed that there is an inherent relationship between dead time and controller performance. Specifically, as dead time is increased, the system performance deteriorates. Moreover, we showed that K_d dictates the size of the disturbance. There is a strong positive relationship between θ and K_d. In other words, the system will frequently encounter large disturbances with large dead times and give rise to poor performance. The converse of this statement is also true. Dead time plays a more important role in dictating controller performance than it did in the independent case. Hence, it should receive a larger sensitivity which is reflected in the RCS.

Let us now give some perspective into the computational cost of the analysis. The simulations were constructed in Matlab® and ran on a Toshiba Tecra® laptop computer with 4.00 GB of RAM and 2.70 GHz dual-core Intel processor. No parallelization schemes were used. The double-loop MC simulation required for VB and MF sensitivities took approximately 22 hours of computation time. The single-loop simulation required for CB sensitivity took only 1 minute and 10 seconds. The staggering difference in computation time makes CB sensitivity a much more favourable option for SA.

Summary

Two ideas are at the core of this chapter. First, the interpretation of sensitivity becomes more difficult in the presence of input codependencies. Dependence causes the

sensitivity of a given input to be influenced by other components of the input vector. Second, the notion of sensitivity can be realized through a variety of definitions. How one chooses to define sensitivity directly influences how the simulation must be structured. VB and MF sensitivities requires the execution of multiple double-loop MC simulations. By Remark 3, we must also assume that the inputs are normally distributed when these sensitivity measures are used. The measure which we have proposed (CB sensitivity) is not subject to this limitation because it only requires the use of one single-loop MC simulation. Hence, we have much more flexibility with how the inputs can be modelled. In the following chapter, we present a flexible approach to modelling variates with codependencies. When we couple this approach with CB sensitivity, a general methodology with a small computational burden is obtained.

Chapter 3

Generating Codependent Samples

In the previous chapter, we introduced three types of sensitivity and contrasted them using a linear model and the CP model. We showed that a SA would arrive at the same conclusions, regardless of the type of sensitivity used. However, there is a substantial difference in computational cost when input codependencies are present. An analysis based upon VB/MF sensitivities requires that d double-loop MC simulations be executed where d is the number of inputs. If CB sensitivities are used instead, then all sensitivities can be calculated from one single-loop MC simulation.

The single-loop simulation has a decisive computational advantage over the brute-force method. There is one more advantage to using a single-loop MC simulation for SA; namely, that conditional sampling is not required. A double-loop MC simulation makes use of samples from the distribution, $X|X_i = x_i^*$. In light of Remark 3, a practitioner is inclined to assume that the inputs follow a multivariate normal distribution. For each input, its mean and variance must be specified in addition to a correlation matrix which encodes the joint behaviour of the random vector.

Suppose that a SA is to be based upon CB sensitivities. Is assuming normality

in the inputs the best course of action? The examples in Chapter 2 utilized the multivariate normal distribution for convenience. In some cases, we could find closed-form expressions through simple algebraic manipulations. These manipulations are not possible on most models and so MC methods facilitate analysis. Now there is no longer any benefit to assuming that the inputs are normally distributed. In sum, provided that we can generate input samples to feed into the model, then a single-loop MC simulation can be executed.

In this chapter, we will address the sampling of variates with codependencies. First, we will discuss the traditional method where samples are drawn from the multivariate normal distribution. An inherent limitation with this method will be highlighted at this time. Second, alternative methods which are more flexible that the former are discussed. Finally, we present a contemporary sampling method which is based upon *copulas*. The flexibility of this method makes it ideal for SA based upon CB sensitivities.

3.1 The Assumption of Normality

We will begin by presenting the machinery needed to generate samples from the multivariate normal distribution. It will be assumed that the computing environment we are working in can generate a random sample which is uniformly distributed on $[0, 1]$. Draw d of these samples and arrange them into a vector. Call this vector, T. We now state the following theorem.

Theorem 3. *[15]* **(Probability Integral Transform)** *Suppose that W_i is a continuous univariate random variable with a CDF denoted by F_i. The variate defined by $T_i = F(W_i)$ is uniformly distributed on $[0, 1]$, that is $T_i \sim \mathcal{U}[0, 1]$.*

By this theorem, we can transform T into a random vector, W, through $W_i = F_i^{-1}(T_i)$, where W_i follows an arbitrary continuous univariate distribution. Note that the elements of W are independent. Suppose we wish to generate samples from a multivariate normal distribution. Without loss of generality, we will assume that we are working with standardized variates. Let us denote the CDF of the standard normal distribution by Φ. The vector Z' where $Z'_i = \Phi^{-1}(V_i)$ is independent and normally distributed. The following theorem can be used to produce a correlated random vector.

Theorem 4. *[35]* **(Generating Correlated Samples)** *Take* Π *to be a correlation matrix. Denote its Cholesky decomposition by* $\Pi = L^T L$ *where* L *is an upper triangular matrix. If* $Z' \sim \mathcal{N}_d(0, I)$*, then* $Z \sim \mathcal{N}_d(0, \Pi)$ *where* $Z = LZ'$*.*

In sum, the above strategy can be used to generate d-dimensional samples from a multivariate normal distribution. As an illustration, we have generated 1000 samples from a standard bivariate distribution which we have specified $\text{Corr}\,[Z_1, Z_2] = -0.7$. Figure 3.1 displays these samples in a scatter plot. Histograms flank each axis to illustrate the marginal behaviour of each variate. In the lower left-hand corner the sample Pearson's correlation and sample Spearman's correlation are displayed.

Let us clarify one point regarding nomenclature. Previously, all mentions of correlation are referring to Pearson's correlation. Suppose that we have two sample sets. Pearson's correlation measures the correlation of the raw scores of the data. If we were to first rank each sample set and then calculate the correlation of the ranks, the value which we calculate is Spearman's correlation. We recommend [12] for a thorough examination of these two dependence measures. The dependence measures which are presented with each set of samples will be addressed in Section 3.4. From

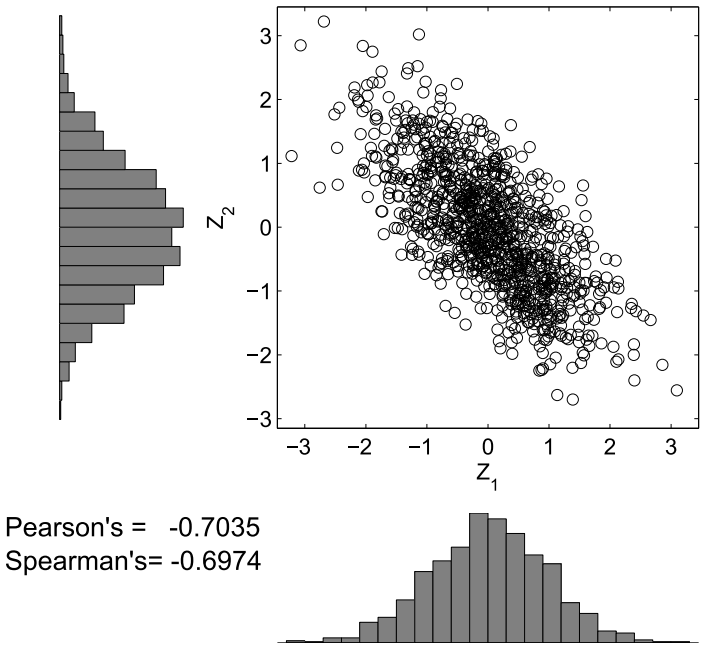

Figure 3.1: 1000 samples from a standard bivariate normal distribution, correlation=-0.7

this point forward, if we do not specify the type of correlation, then it is assumed that we are referencing Pearson's correlation.

Symmetry is a notable feature of Figure 3.1. Specifically, we see that the marginals are symmetric about their means. A well-known property of the normal distribution is that its skewness is zero. Suppose we wish to model a random vector which has some elements with skewness far from zero. Clearly, the multivariate normal distribution is inappropriate for this task. And so the traditional multivariate sampling protocol gives a practitioner very little latitude. This shortcoming has driven the development of other sampling protocols.

3.2 Generating Non-normal Samples

In the previous section, we showed that by applying a set of transformations to an initial sample, we can obtain samples from the multivariate normal distribution. In 1978, Johnson and Ramberg proposed the further transformation of normally distributed samples [33]. Specifically, the normal random vector which we have called Z is transformed such that the transformed vector follows a non-normal distribution. This distribution must come from a class of distributions defined in [34]. The challenge with strategy is choosing the mean vector and the covariance matrix of Z such that the non-normal sample exhibits the desired behaviour. The novelty of the 1978 work is that relationships between the parameters of the normal distribution and the parameters of the non-normal distribution were derived. Hence, a practitioner can control the behaviour of the non-normal sample. However, the major drawback of this algorithm is its lack of generality. Only two non-normal distributions were presented in this work. This is due to the difficulty surrounding the derivation of the parameter

relationships.

In 1982, Iman and Conover presented an algorithm which avoids using transformation to induce dependence in the sample set [30]. Two types of inputs are required from the user. Each element of the random vector is assigned a univariate distribution which is the desired marginal distribution. A Spearman's correlation for each pair of variates must also be supplied. N samples are drawn from each marginal distribution and arranged into a vector. Theorem 3 can be used to facilitate this step. Note that this set of sample vectors are uncorrelated. The algorithm reorders the sample vectors such that the generated data approximates the user-specified Spearman's correlations. While this algorithm is simple to implement, it has been criticized for its *ad hoc* nature [13]. However, because of its novelty at the time, the Iman-Conover algorithm has largely been embraced by the simulations community [49].

Copulas provide a flexible framework for specifying multivariate distributions. In the following section, we present a contemporary sampling algorithm which uses a copula. By contemporary, we mean that most publications presenting algorithms of this nature date back to the late 1990's. Preliminary copula theory had been established by the 1950's. Why was there such a delay in the simulations community adopting copulas? Prior to this point, practitioners were largely content using multivariate distributions, in particular, the multivariate normal distribution. In the 1980's, financial markets were in the middle of a revolution. In short, markets were becoming increasingly complex and mathematicians were being employed to model said markets. Mandelbrot had established that prices in general do not follow a normal distribution [44]. This trend extends to many other financial variables. Researchers now had a large amount of incentive to develop copula technologies. As a result,

there was a large spike in the number of publications related to this topic around this time [20]. Copula-based sampling arose from this movement but the Iman-Conover algorithm already had already achieved a foothold in simulation at this time.

3.3 Copula-based Sampling

This section has been divided into two parts. In the first part, we illustrate the flexibility of copula-based sampling through an example. This example will form the basis of the discussion of the theory in the second part of this section.

3.3.1 An Example

The method which we are about to implement has been presented in several works [4, 6, 41, 43]. An interesting note is that only the first two references describe this algorithm in terms of copulas. The first two references have more *ad hoc* presentations of the algorithm. Copula-based sampling bears a closer resemblance to the Johnson-Ramberg algorithm than the Iman-Conover algorithm. The Z samples (as described in Section 3.1) are being further transformed to have arbitrary marginal behaviours.

We begin by defining a random vector by

$$U_i = \Phi^{-1}(Z_i) \text{ for } i = 1, 2 \tag{3.1}$$

This transformation is then applied to the samples used to construct Figure 3.1. The new scatter plot is displayed in Figure 3.2. By Theorem 3, we know that U_i should be uniformly distributed. Indeed, this statement is confirmed when we inspect the accompanying histograms.

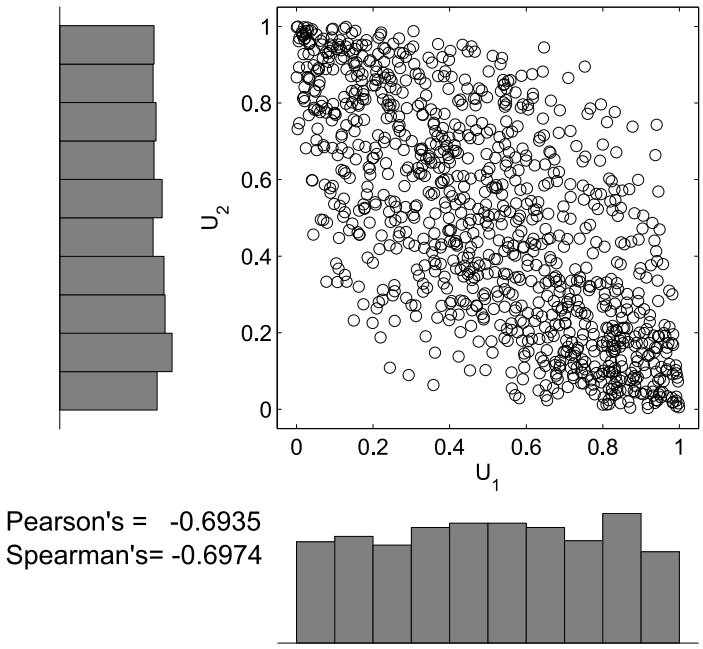

Figure 3.2: 1000 samples from a normal copula

We apply Theorem 3 yet again in the definition of the following random vector.

$$A_i = F_i^{-1}(U_i) \text{ for } i = 1, 2 \tag{3.2}$$

The marginal behaviour of the i^{th} component is dictated by the choice of F_i. For example, let F_1 be CDF of a beta distribution ($\alpha = \beta = 0.5$) and F_2 be the CDF of a normal distribution ($\mu = 2,\ \sigma^2 = 2$). Note that all parametrizations in this section are the same as in their respective Matlab® functions. Figure 3.3 displays the samples of A which were generated through the transformation of the samples in Figure 3.2. Observe that the marginal behaviour, as indicated by the histograms, behaves as expected.

For the purpose of comparison, we will define another bivariate random vector called B. It will be defined in the same fashion as in (3.2) but where F_1 is the CDF of an exponential distribution ($\lambda = 1$) and F_2 is the CDF of a chi-squared distribution ($\nu = 5$). The samples are displayed in Figure 3.4. Again, we see that the marginal behaviours of each variate is as expected.

In sum, this series of transformations (Z to U to A or B) has allowed us to generate two different sample sets with codependencies. The marginal behaviour of these sample sets was dictated through the choice of F_i. Recall that when assuming normality, we are forcing the samples to be symmetric about their mean. In the case of A, there is also symmetry about the mean but the marginals are clearly non-normal. In the case of B, the marginals are clearly asymmetric. This illustrates how copula-based sampling is more flexible than traditional sampling. In the remainder of this chapter, we will present the framework which underpins this method.

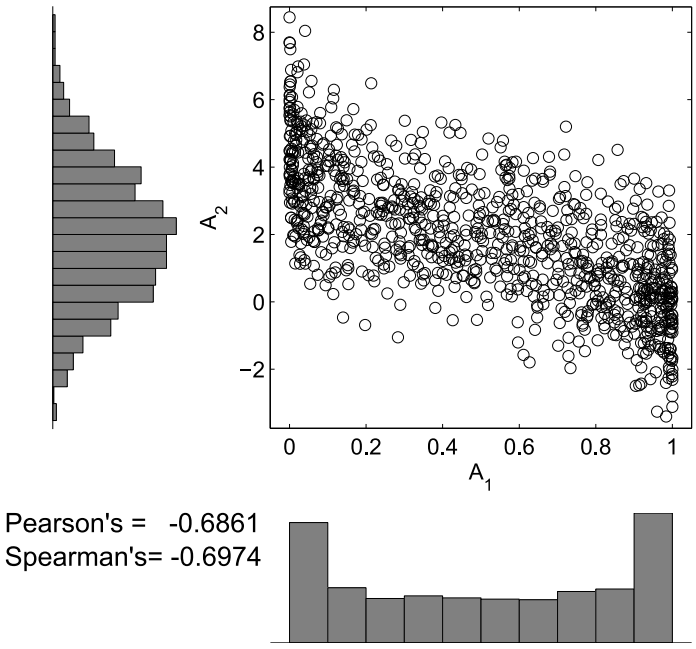

Figure 3.3: 1000 samples (of A) generated using a normal copula and symmetric marginals

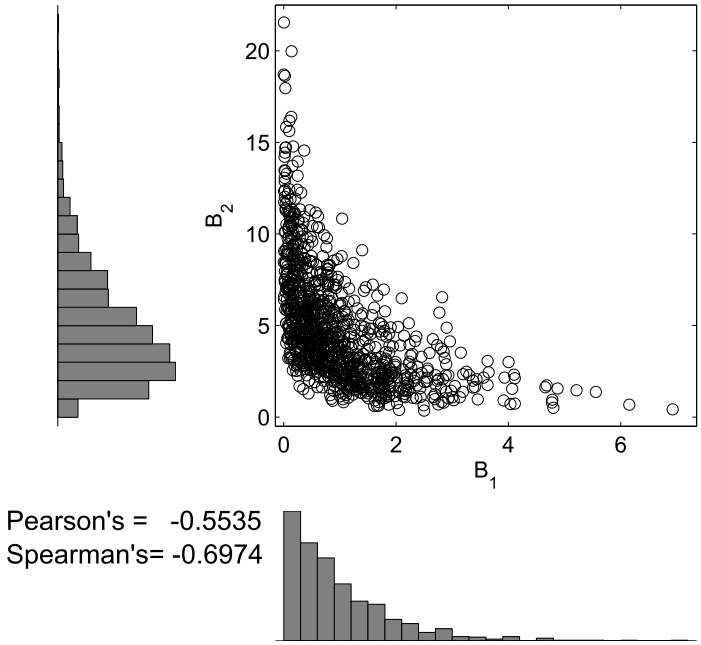

Figure 3.4: 1000 samples (of B) generated using a normal copula and asymmetric marginals

3.3.2 Theory of Copulas

In our example, we generated samples from two random vectors, A and B. Both of these vectors were defined through the transformation of U. How does one define the distribution of U? We considered the bivariate case in our example but let us consider the general case of d variates here. Recall that U arises from the transformation of Z, specifically, $U_i = \Phi(Z_i)$. Let us denote the d-dimensional standard normal CDF with correlation matrix, Π, by Φ_Π. It is obvious that the CDF of U is

$$C(u_1, \ldots, u_d) = \Phi_\Pi(\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_d)) \tag{3.3}$$

The above equation is the definition of the *normal copula* [10]. Obviously, "normal" comes from the multivariate distribution which it is based upon. This is one example of a broad class of functions. We will formalize the notion of a copula shortly but let us first offer an interpretation of this function. We begin by defining the random vector X where $X_i = F_i^{-1}(U_i)$. Only the marginal behaviours of X are specified in this definition. Clearly, the joint behaviour of X is dictated by the joint behaviour of U. We know that U must exist within the unit cube, $[0, 1]^d$. The copula function characterizes how U is distributed within this cube. Therefore, copulas provide the means for a practitioner to treat the marginal and joint behaviours of X separately.

Let us now define this notion in a more rigorous sense by posing two questions.

1. What is a copula?
2. How can a copula be used to define X?

The concept of a copula was first expressed in Hoeffding's 1940 Ph.D. dissertation. The fundamentals of bivariate copula theory were established and three papers arose

from this dissertation [26]. All three articles were written in German and two were published in little-known journals [14]. Therefore, the content was largely inaccessible to the statistics community. The idea of a copula was separately developed by Fréchet in 1951 [16]. The following theorem, which bears the name of both authors, answers Question 1.

Theorem 5. *[16, 26]* **(Fréchet-Hoeffding Bounds)** *Let* $M^d(u_1, \ldots, u_d) = \min\{u_1, \ldots, u_d\}$ *and* $W^d(u_1, \ldots, u_d) = \max\left\{1 - d + \sum_{i=1}^{d} u_i, 0\right\}$. *Any function,* $C : [0,1]^d \to [0,1]$ *which satisfies the bounds*

$$W^d(u_1, \ldots, u_d) \leq C(u_1, \ldots, u_d) \leq M^d(u_1, \ldots, u_d)$$

is a copula.

Note that there are some technicalities surrounding the lower bound, W^d. These issues are beyond the scope of this work but they have been addressed by Nelsen [51, p.42].

Now consider Question 2. In Sklar's 1959 paper, the term *copula* is coined which is a Latin noun for "link". Moreover, a theorem is presented which shows how copulas can be used to construct/decompose multivariate distributions [70]. This theorem is as follows.

Theorem 6. *[70]* **(Sklar's Theorem)** *Take* $X_1, \ldots, X_d$ *to be continuous variates. Let* F_i *be the marginal CDF of the* i^{th} *variate. Let* H *be the CDF of* X. *There exists a unique copula,* C, *such that*

$$H(x_1, \ldots, x_d) = C(F_1(x_1), \ldots, F_d(x_d)) \quad \forall x$$

Conversely, given $F_1, \ldots, F_d$ and C, H is a d-dimensional CDF with $F_1, \ldots, F_d$ as marginal CDFs.

Remark 4. *Recall that in Remark 3, we stated that the theory does not exist for conditioning non-normal multivariate distributions. Patton has stated a theorem resembling Theorem 6 for conditional copulas [54]. However, Patton acknowledged that one cannot derive a conditional copula from an unconditioned copula. It follows that copulas can only be applied to modelling inputs when a SA is based upon CB sensitivities.*

We have briefly presented two theorems which can be used for modelling variates with codependencies. For a more thorough presentation of copulas, the standard reference is [51]. If the reader prefers a reference for the uninitiated, we recommend [19]. Copulas have been applied in a wide range of fields such as actuarial science [17], finance [39] and reliability engineering [75]. To our knowledge, no one has applied copulas in an SA setting.

Recall that we motivated this discussion by presenting a sampling algorithm which uses a normal copula. From the generality of Theorem 5, it is clear many different copulas can be derived. Indeed, probability theorists have generated a large library of copulas which can be found in [51]. A different copula will induce a different joint behaviour in X.

To illustrate this point, we will generate scatter plots similar to Figures 3.2-3.4. However, the Clayton copula (as defined in [51, p.115]) which is parametrized by θ, will be used instead. We will distinguish these random vectors from the random vectors based upon the normal copula by using the prime, $'$. Figures 3.5-3.7 are the resulting plots where $\theta = 5$. Notice that the marginal behaviours, as indicated

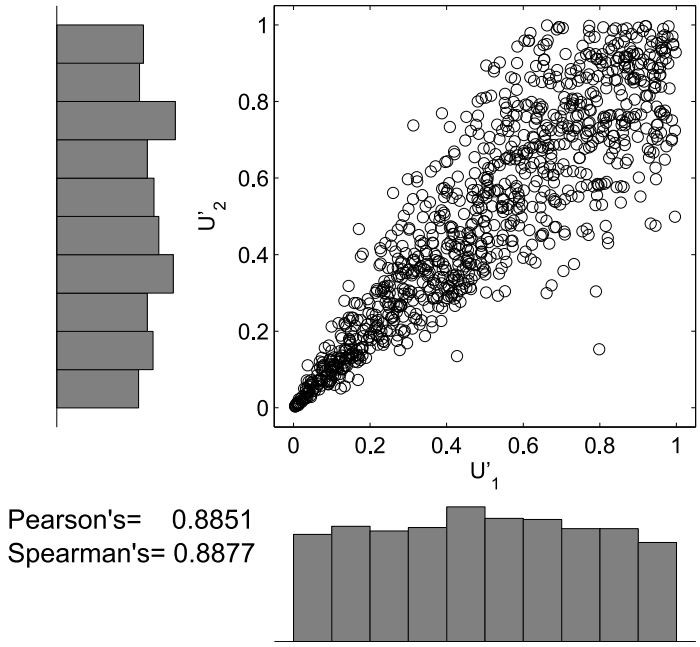

Figure 3.5: 1000 samples from a Clayton copula

through the histograms, are the same as in the case of the normal copula. The scatter plot is a realization of the joint behaviour. When we compare the respective plots for the two copulas we see contrasting joint behaviours.

We will restrict our attention to sampling based upon the normal copula in this work. In other words, the sampling protocol shown in our example will be used. Our reasoning is that the normal copula is well-suited for modelling X when d is large. Many copulas are not as flexible. For example, the Clayton copula has only been defined for when $d = 2$. Moreover, it can only accommodate positive dependence structures. In closing, we believe that the normal copula is the best candidate for SA.

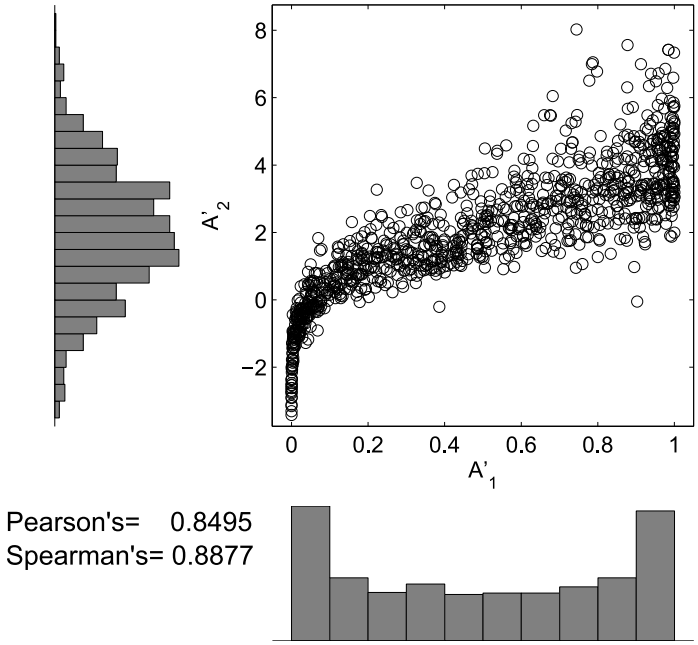

Figure 3.6: 1000 samples (of A') generated using a Clayton copula and symmetric marginals

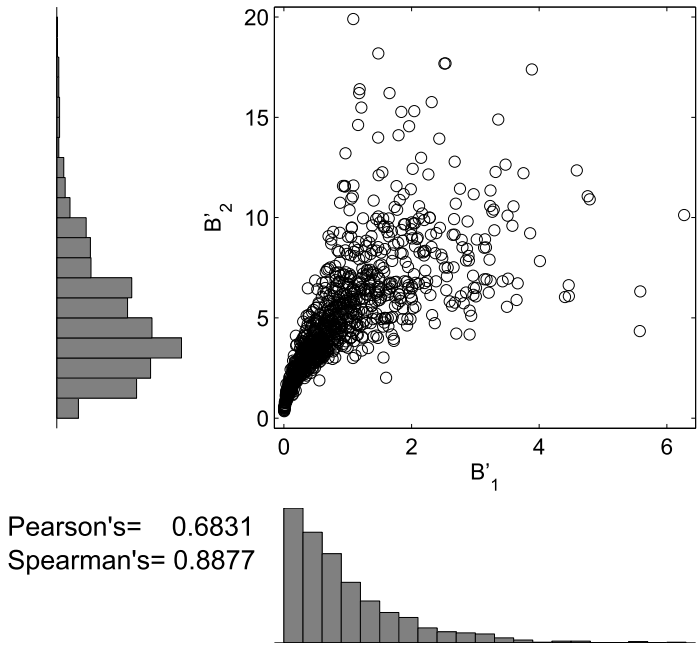

Figure 3.7: 1000 samples (of B') generated using a Clayton copula and asymmetric marginals

3.4 A Problem

There is an inherent problem with generating samples based on the normal copula. We will pose a solution in the following chapter but the problem will be formulated here. Consider our example from earlier in this chapter (Figures 3.1-3.4). Notice that the sample Spearman correlations remains the same and the sample Pearson correlation becomes distorted with every transformation. Recall that Spearman's correlation is based upon ranks and not raw scores. The ranks of a sample set are invariant to monotonically increasing transformations, which both F_i and F_i^{-1} are. Hence, Spearman's correlation is invariant to the marginal behaviours of a bivariate random vector. Copula theory formalizes this observation through the following identity [51, p.167]. For the random vector $[X_1, X_2]^T$ with C as a copula

$$\rho_S(X_1, X_2) = 12 \iint_{[0,1]^2} C(u_1, u_2)\ du_1 du_2 - 3 \tag{3.4}$$

where ρ_S is Spearman's correlation.

The problem is that sampling works in terms of Pearson's correlation and not Spearman's correlation. When we generate samples of Z, (recall $Z \sim \mathcal{N}_d(0, \Pi)$), we must specify the correlation matrix Π. After applying the necessary transformations, we arrive at X which has the correlation matrix Σ. Note that Σ is simply a distorted form of Π.

Let us discuss distortion in the context of our example. Table 3.1 summarizes the sample Pearson's correlations. Begin by considering the transformations of Z to U and Z to A. In both cases, we see only small changes in Pearson's correlation; that is, it only experiences a small amount of distortion. Upon examining the transformation

Table 3.1: Distortion of Pearson's correlation

Random Vector	Sample Pearson's correlation
Z	-0.7035
U	-0.6935
A	-0.6861
B	-0.5535

of Z to B, we see that the extent of distortion is more significant. What is different in this case? Recall that the histograms of Figures 3.1-3.4 are a realization of the marginal behaviours of the random vector. Notice that the marginals of U and A are symmetric and the marginals of B are clearly not. Cario and Nelson have identified that the symmetry of a PDF plays a large role in the extent of distortion [4]. We will now briefly state their findings. If the marginal PDFs of X are symmetric (ie. zero skewness), then Π will undergo little distortion and so $\Pi \approx \Sigma$. On the other hand, if the marginal PDFs of X have non-zero skewness, then Π will be distorted to a greater extent.

Li and Hammond have derived a relationship between the i, j element of Π and Σ [41]. Hence, it is "possible" to back-calculate Π from Σ but this requires substantial computational effort. We say "possible" because while a "Π" can be computed, there is the potential for it not being able to be implemented. In the following chapter, we characterize the correlation matrix through three requirements; the most challenging of which is the positive semi-definite requirement. This requirement means that the elements of Π cannot be arbitrarily specified. The Li-Hammond method individually calculates each element so it cannot ensure the positive semi-definite requirement. This strategy may produce a valid Π but Ghosh and Henderson have shown that the probability of this happening goes down drastically as dimensionality increases [21].

We propose the following strategy. A practitioner poses a correlation matrix in terms of X and then assumes that $\Pi = \Sigma$. Samples of X can then be generated using this assumption. To check that this assumption is appropriate, compute the sample correlation matrix. If the match between this matrix and Σ is satisfactory, then proceed with the SA based upon CB sensitivity.

One step is required prior to executing this strategy. We have stated that Σ must be positive semi-definite; that is, the elements of Σ cannot be arbitrarily chosen. Consider specifying a matrix, called $\widehat{\Sigma}$, which has all of the properties of a correlation matrix with the exception of positive semi-definiteness. A practitioner does not face restrictions when posing $\widehat{\Sigma}$. In the next chapter, we will present an algorithm which can compute a Σ which approximates a given $\widehat{\Sigma}$. Once this algorithm has been executed, a practitioner can proceed with procedure outlined in the previous paragraph.

Summary

In this chapter, we have shown how non-normal multivariate samples can be generated by the transformation of multivariate normal samples. This procedure gives practitioners added flexibility in the modelling of the inputs for SA based upon CB sensitivity. Copulas provide the mathematical framework which support this procedure.

Chapter 4

Specifying a Correlation Matrix

In the previous chapter, we showed how copulas could be used to separate the marginal and joint behaviours of a distribution. Furthermore, the normal copula was identified as a powerful tool for modelling X when d is large. The codependencies between variates is encoded in the correlation matrix. In the context of SA, a practitioner must specify the correlation matrix based upon their knowledge of the inputs. A correlation matrix, which we will denote by Σ, can be characterized by three points [59].

1. The matrix is square symmetric with a unit diagonal.
2. The (i, j) element corresponds to $\text{Corr}[X_i, X_j]$. As a result, all off-diagonal elements must be contained on $[-1, 1]$.
3. The matrix is positive semi-definite.

When $d = 2$, the off-diagonal element can be arbitrarily specified and all three points are met. However, honouring the third point when $d > 2$ is complicated. The elements cannot be arbitrarily chosen. To illustrate why this is the case, we perform the following thought experiment.

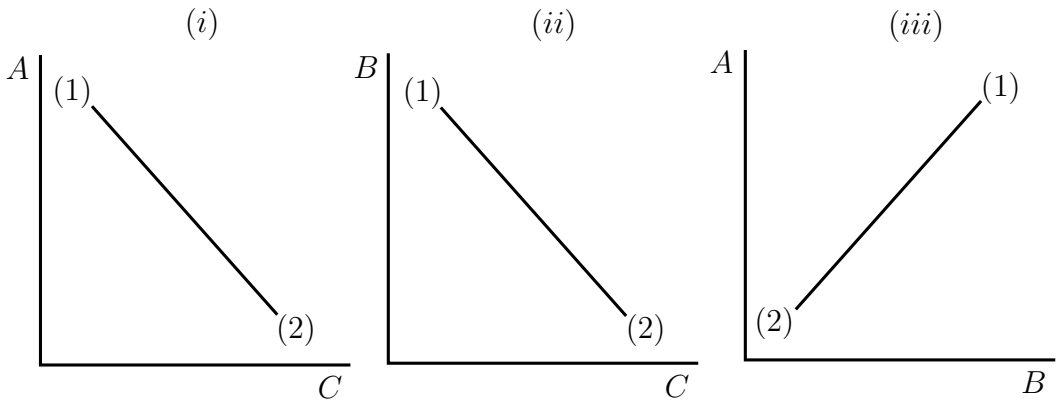

Figure 4.1: An illustration of how Σ cannot be arbitrarily specified

Consider a trivariate random vector with its elements labelled as A, B and C. Suppose that we make two assumptions. First, A and C are perfectly negatively correlated. Second, B and C are perfectly negatively correlated. Plots (i) and (ii) of Figure 4.1 illustrate our assumptions. It follows that the correlation between A and B must be perfect as well. What values of can the correlation of A and B take on? Our main concern is the endpoints. Let us label these points as (1) and (2) in Figure 4.1. We will use the following notation so that the extremal values of each component can be distinguished. Taking A as an example, we define $a^L = \min(A)$ and $a^U = \max(A)$. The same notation is used for B and C. From our assumptions, it is clear that (1) is $\left[a^U, b^U, c^L\right]$ and (2) is $\left[a^L, b^L, c^U\right]$. Now consider a plot of A and B which is shown in (iii). The endpoints have already been defined from our two assumptions and have been labelled in the plot. In sum, we cannot arbitrarily choose the correlation structure of a random vector. This idea is formalized through point 3.

A pseudo-correlation matrix, which we will denote by $\widehat{\Sigma}$, is a matrix which obeys points 1 and 2 but not 3. Inconsistent structures can be encountered in practice.

When estimating Σ, the off-diagonal elements are sample correlations. If the data is noisy or pieces are missing, then the estimator may fail to meet the positive semi-definite requirement. The challenge is to turn $\widehat{\Sigma}$ into a valid correlation matrix. We will refer to this operation as *salvaging a correlation matrix.* This problem has been addressed in [3, 25, 36, 38, 50, 56, 57, 59, 69]. With regards to SA, a practitioner is free to specify a pseudo-correlation matrix. This matrix can then be salvaged and used in MC simulations.

Salvaging requires that the following optimization problem be solved.

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & P(\widehat{\Sigma}, \Sigma), \quad \Sigma = q(\theta) \\ \text{subject to} \quad & \theta \leq r(\theta) \end{aligned} \tag{4.1}$$

where $\theta \in \mathbb{R}^m$, $P : \mathbb{R}^{d\times d} \times \mathbb{R}^{d\times d} \mapsto \mathbb{R}$, $q : \mathbb{R}^m \mapsto \mathbb{R}^{d\times d}$, and $r : \mathbb{R}^m \mapsto \mathbb{R}^m$. In classical optimization, P is the cost function and θ constitutes the decision variables. In this work, we will refer to P as the proximity function as it is a better indication of its meaning. The idea behind salvaging is to find the correlation matrix which is "closest" to a given pseudo-correlation matrix. We will denote the solution of (4.1) by Σ^*.

Vector norms can be extended to matrices. For an arbitrary matrix, A, where $A \in \mathbb{R}^{d\times d}$, the Frobenius norm is defined by

$$||A||_F = \left(\sum_{i=1}^{d} \sum_{j=1}^{d} a_{ij}^2 \right)^{\frac{1}{2}} \tag{4.2}$$

This quantity could be interpreted as the "length" of A. If we were to take $A = \Sigma - \widehat{\Sigma}$, then $||\Sigma - \widehat{\Sigma}||_F$ is the distance between Σ and $\widehat{\Sigma}$. Other matrix norms could be

used but the Frobenius norm has several properties which has made it popular within matrix-nearness literature. Suppose we desire to have a particular element of $\widehat{\Sigma}$ match up well to its corresponding element in Σ^*. Weights need to be introduced into the proximity function to realize this goal. Higham has established the Frobenius norm to be opportune for this purpose [25]. What is more, this norm is invariant to changes of basis. This property in particular facilitates the proximity function we will pose later in this chapter.

The generality of the problem formulation lends itself to numerous interpretations. A practitioner could state any proximity function they deem suitable. The challenge lies in structuring $q(\theta)$ and $r(\theta)$ to produce a valid correlation matrix for any θ in the feasible set.

The earliest attempt at salvaging can be traced back to the work of Devlin *et al.* in 1975. The idea starts by defining a target matrix, Γ, where $\Gamma \in \mathbb{R}^{d\times d}$ and a mixing parameter, α, where $\alpha \in [0, 1]$. The mixing parameter is the equivalent to θ in this situation. The correlation matrix can be cast as convex combination of the target matrix and the pseudo-correlation matrix, that is $q(\alpha) = \alpha\Gamma + (1 - \alpha)\widehat{\Sigma}$. Originally, a proximity function was posed and the optimization was carried out numerically. However, in 2003, Ledoit and Wolf derived an analytical expression for the optimal α when the Frobenius norm proximity function is used [38]. While this result looks promising, the minimizer is dictated by Γ. No concrete strategy for choosing Γ exists.

Modern salvaging strategies are based upon more sophisticated formulations. Larger θ vectors are used in conjunction with more complex mappings to realize the necessary Σ. For examples, we refer the reader to [3, 56, 77]. The ultimate goal is to design an optimization scheme which renders rapid convergence when d is large. Authors

reformulate (4.1) with two goals in mind. First, m must be kept small relative to d. In optimization, convergence rate is exponentially related to the dimensionality of the problem. Second, alternative formulations can provide easier access to objects such as gradients and Hessians. In general, the calculation/approximation of these objects comprises a large portion of the computational effort expended by the solver. The details of many of these strategies are beyond the scope of this work.

This chapter will proceed as follows. First, we present a straightforward formulation which can be coupled with a standard nonlinear optimization solver. Second, we discuss a problem with using the proximity function based upon the Frobenius norm. Third, we pose an modified proximity function which avoids this issue. MC simulation is used to validate this claim.

4.1 Rebonato and Jäckel's Method [59]

Rebonato and Jäckel have proposed two approaches for salvaging, namely, hypersphere decomposition and spectral decomposition. Hypersphere decomposition defines $q(\theta)$ as a mapping between a d-dimensional sphere and Σ. No $r(\theta)$ is required which means the optimization problem is unconstrained. The choice of proximity function is at the discretion of the practitioner. In spectral decomposition, the spectrum of Σ is manually chosen to resemble the spectrum of $\widehat{\Sigma}$. This approach has no element of optimization. The authors have suggested that these algorithms could be used in tandem. The nonlinear solver used on the hypersphere decomposition requires an initial guess. Spectral decomposition can provide this guess at a relatively low computational cost. This policy forms the foundation of our salvage algorithm.

4.1.1 Finding an initial guess *(Spectral Decomposition)*

In this approach, the underlying idea is to place the eigenvalues of Σ such that they match up well to those of $\widehat{\Sigma}$. Begin by performing an eigendecomposition on $\widehat{\Sigma}$.

$$\Lambda_{\widehat{\Sigma}} = V_{\widehat{\Sigma}}^T \widehat{\Sigma} V_{\widehat{\Sigma}} \tag{4.3}$$

where $V_{\widehat{\Sigma}}$ is the eigenbasis of $\widehat{\Sigma}$ and $\Lambda_{\widehat{\Sigma}}$ is a diagonal matrix where $\left(\Lambda_{\widehat{\Sigma}}\right)_i = \text{eig}(\widehat{\Sigma})$. We now define another spectrum which is encoded in the diagonal matrix defined by

$$\Lambda'_{\widehat{\Sigma}} : \left(\Lambda'_{\widehat{\Sigma}}\right)_i = \begin{cases} \left(\Lambda_{\widehat{\Sigma}}\right)_i & \text{if } \left(\Lambda_{\widehat{\Sigma}}\right)_i \geq 0 \\ 0 & \text{if } \left(\Lambda_{\widehat{\Sigma}}\right)_i < 0 \end{cases} \tag{4.4}$$

Recognize that all of the negative eigenvalues of the pseudo-correlation matrix have been set to zero.

The initial guess for the nonlinear solver, Σ_o, can be expressed through a Cholesky decomposition. Formally, this is stated by $\Sigma_o = B_o^T B_o$. For a diagonal matrix A, the Cholesky decomposition will be denoted by $A = \sqrt{A}\sqrt{A}$. For a moment, suppose we define the Cholesky factor of the initial guess to be $\sqrt{\Lambda'_{\widehat{\Sigma}}} V_{\widehat{\Sigma}}$. The resulting "correlation matrix" would be positive semi-definite but not possess a unit diagonal. It follows that the "correlation matrix" must be scaled up. We define

$$T : \quad T_i = \left[\sum_j \left(V_{\widehat{\Sigma}}\right)_{ij} \left(\Lambda'_{\widehat{\Sigma}}\right)_j \right]^{-1} \tag{4.5}$$

The Cholesky factor of the initial guess can now be defined by

$$B_o = \sqrt{\Lambda'_{\widehat{\Sigma}}} V_{\widehat{\Sigma}} \sqrt{T} \tag{4.6}$$

For this initial guess to be implemented, we must solve for θ_o where $\Sigma_o = B_o^T B_o$ and $B_o = q(\theta_o)$. Clearly, this mapping is nonlinear and numerical methods are required. There is no need to solve this system to a high degree of accuracy. The tolerances can be set relatively large so that an approximation of θ_o can be obtained in a reasonable amount of time.

4.1.2 Defining $q(\theta)$ *(Hypersphere Decomposition)*

The mapping, $q(\theta)$, can now be established. A correlation matrix can be expressed as $\Sigma = B^T B$. The idea is to relate a point on the unit hypersphere to B. The angles which define said point constitute θ. It follows that $B = q(\theta)$. Let us revamp our definitions of θ and $q(\theta)$ to better fit the problem at hand. Let $\theta \in \mathbb{R}^{d\times(d-1)}$ which forces $q : \mathbb{R}^{d\times(d-1)} \mapsto \mathbb{R}^{d\times d}$. We define

$$B: \quad B_{ij} = \begin{cases} \cos\theta_{ij} \ \prod_{k=1}^{j-1} \sin\theta_{ik} & \text{for } j = 1, \ldots, n-1 \\ \prod_{k=1}^{j-1} \sin\theta_{ik} & \text{for } j = n \end{cases} \tag{4.7}$$

Therefore, this approach uses $d \times (d-1)$ decision variables. Several authors have posed modifications to reduce this number [36, 57]. The transparency of the standard approach has prompted us to avoid these alternatives.

Note that this problem formulation is nonconvex and so we have two options. We

could expend a large amount of computational effort attempting to find a global minimum. Alternatively, we could find a local minimum at a much lower computational cost. We have elected to use local minimizers for the following reasoning. It is understood that Σ^* is not precisely the $\widehat{\Sigma}$ they specify. In short, the minimizer simply needs to be close to $\widehat{\Sigma}$. Expending a large amount of computational power to find a global minimizer is unnecessary. The spectral decomposition provides a very good initial guess. Hence, an algorithm which finds a local solution will converge in a fraction of the time a global solver would require. This strategy may be particularly beneficial when d is large. In this work, Matlab's Trust-Region Reflective Algorithm was used for both finding θ_o and solving the optimization problem. Its default settings were always used.

4.2 Issues with Salvaging

4.2.1 Not all $\widehat{\Sigma}$ are created equal

Notice that we have stated the proximity function, $P(\widehat{\Sigma}, \Sigma)$, such that it has two inputs. First, the practitioner specifies $\widehat{\Sigma}$ based upon their knowledge of the inputs. Second, the salvage algorithm is ran to adjust Σ such that it matches up well to $\widehat{\Sigma}$. Consider the impact of $\widehat{\Sigma}$ on this approach. It follows that some pseudo-correlation matrices are easier to salvage. Alternatively, some proximity functions obtain a lower optimal value than others.

What indicates the difficulty of a salvage? This question poses a dilemma. It is typical to consider one or two test problems in optimization literature. A specific $\widehat{\Sigma}$ translates to one proximity function and thus one problem. Here, a class of problems

must be investigated. Recall that MC methods explore a given space through random sampling. Suppose each off-diagonal element on one side of the diagonal is uniformly distributed on $[-1, 1]$. A sample pseudo-correlation matrix can be formed by drawing a sample for each element and arranging them into a matrix with the required structure. Hence, we can explore the space which $\widehat{\Sigma}$ lives in. A salvage is performed on each sample $\widehat{\Sigma}$ drawn. The output data can be used to understand the behaviour of the salvage algorithm. We will use the standard $P(\widehat{\Sigma}, \Sigma) = ||\Sigma - \widehat{\Sigma}||_F$ and let $d = 5$ as an example. The MC simulation used 1000 sample pseudo-correlation matrices.

The challenge is to identify a connection between some characteristic of $\widehat{\Sigma}$ and $P(\widehat{\Sigma}, \Sigma^*)$. Several characteristics were considered and the smallest eigenvalue, $\widehat{\lambda}_{min}$, of $\widehat{\Sigma}$ was found to show a very strong trend. Figure 4.2 displays this relationship. Other values of d were simulated to confirm the trend. To this end, we propose that $\widehat{\lambda}_{min}$ be used to gauge how successful the salvage will be. If the magnitude of $\widehat{\lambda}_{min}$ is sufficiently large, then the practitioner might consider reformulating $\widehat{\Sigma}$.

4.2.2 Ill-conditioned Minimizers

We must review some concepts from numerical linear algebra to understand the above-mentioned phenomenon. First, we recall the connection between positive semi-definiteness and the spectrum of a matrix. We will take A to be a real symmetric matrix where $A \in \mathbb{R}^{d \times d}$. Because A is Hermitian, all of its eigenvalues are real. Recall that if A is positive semi-definite, then all of its eigenvalues are non-negative. Likewise, if A is positive definite, then all of its eigenvalues are positive. It follows that all positive definite matrices are positive semi-definite, but the converse is not true.

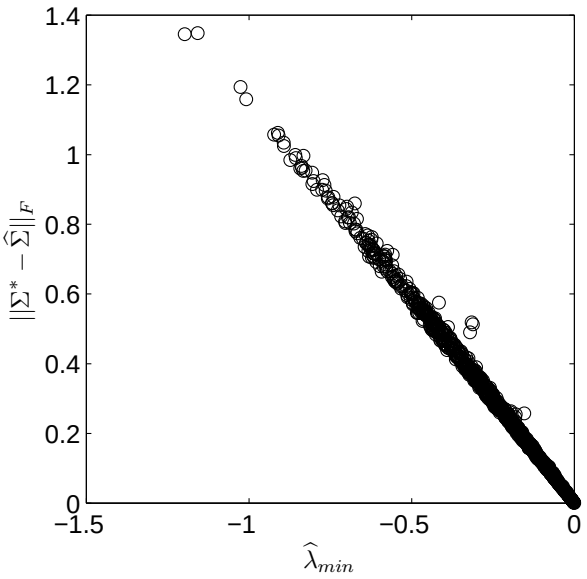

Figure 4.2: A plot of $||\Sigma^* - \widehat{\Sigma}||_F$ against $\widehat{\lambda}_{min}$

Second, we establish the relationship between the spectrum and rank of a matrix. Recall that $\text{rank}(A) + \text{nullity}(A) = d$. For a moment, take $\text{nullity}(A) = 1$. One of the eigenvectors occupies the null space and has a corresponding eigenvalue of zero. A is rank deficient and cannot be inverted. Alternatively, we say that A is singular. This logic extends to the case where multiple zero-valued eigenvalues are encountered.

Third, we introduce the notion of nearly-singular matrices. Recall that through inversion, we are solving for z where the system is defined by $Az = b$. Theoretically, provided that all eigenvalues are non-zero, A can be inverted. Numerically, it is a different story. Because computers have finite precision, one encounters round-off error when inverting a matrix. Matrices which are "far" from singular are insulated from the effects of round-off error. However, when a matrix is "close" to singular, small changes in b can produce very large changes in z. This is caused by the accumulation of round-off error.

The condition number of a matrix indicates the extent to which this phenomenon takes place. We will denote the condition number of A by $\kappa(A)$. By recalling that A is a normal matrix, we define $\kappa(A) = \max(\text{eig}(A))/\min(\text{eig}(A))$. More general definitions exist for $\kappa(A)$ but this definition is sufficient for our purposes. Large condition numbers correspond to matrices which require higher levels of precision to invert accurately. If $\kappa(A)$ is large, then A is said to be ill-conditioned and should not be inverted. As a general rule of thumb, "too large" translates to $\log(\kappa(A))$ being greater than the precision of the computer. The IEEE standard of double-precision floating-point yields 16 digits of precision. Therefore, we will consider a matrix to be ill-conditioned if $\log(\kappa(A)) \gtrsim 16$.

Let us return back to the idea of salvaging. Previously, we established that a large magnitude of $\widehat{\lambda}_{min}$ means a large $||\Sigma^* - \widehat{\Sigma}||_F$. With this in mind, it should be apparent that the minimizer of the given problem formulation is singular. It may be helpful to think about salvaging in terms of spectra. For convenience, let us take $\widehat{\lambda}_i = \text{eig}(\widehat{\Sigma})$ and $\lambda_i = \text{eig}(\Sigma)$. The eigenvalues of the pseudo-correlation matrix can exist to both the right and left of zero. The eigenvalues of the correlation matrix can only exist at or to the right of zero. In some sense, salvaging minimizes the difference between corresponding eigenvalues. This statement will be made more concrete shortly. If $\widehat{\lambda}_i > 0$, then λ_i will be driven to $\widehat{\lambda}_i$ by the salvage. Conversely, if $\widehat{\lambda}_i < 0$, then λ_i will driven to zero. In practice, λ_i will be close but not precisely at these values because the solver will terminate before Σ^* is truly singular.

Most matrix operations cannot be performed on singular matrices. Letting $C = \kappa(\Sigma)$, we state $C = \lambda_{max}/\lambda_{min}$. Suppose that we are dealing with a correlation matrix generated from a salvage. We have established that λ_{min} will be very close to zero.

A bound for λ_{max} can be instituted using a property of trace. Recall that for A, $\text{trace}\,(A) = \sum_i A_{ii} = \sum_i \text{eig}(A)$. For a correlation matrix, the unit diagonal results in $\text{trace}\,(\Sigma) = d = \sum_i \lambda_i$. With this in mind, we can infer $\lambda_{max} \leq d$. From the previous paragraph, we illustrated how λ_{min} is driven very close to zero. Because λ_{max} is bounded, it is clear that C will be very large and salvaged matrix will be nearly singular.

Some applications, such as data compression, make use of rank-deficient correlation matrices. However, this is not the case when generating samples for simulation. Recall that Theorem 4 facilitates the generation of correlated random samples. Chiefly, the Cholesky factor of the correlation matrix is required. The Cholesky decomposition requires that a matrix be well-conditioned. Therefore, the classical salvaging approach is problematic. Some authors have produced salvage algorithms which avoid singularity. Schöttle and Werner introduced an constraint on C in their algorithm [69]. Mishra claims that the same effect can be achieved through constraining $\det(\Sigma)$ [50]. Both of these approaches are indirectly constraining the spectrum. In the section to follow, we propose a modified cost function which penalizes nearly-singular correlation matrices and requires that no constraints be implemented.

4.3 A Modified Proximity Function

The classic proximity function is defined by

$$P(\widehat{\Sigma}, \Sigma) = ||\Sigma - \widehat{\Sigma}||_F = \left(\sum_{i=1}^{d} \sum_{j=1}^{d} \left(\Sigma_{ij} - \widehat{\Sigma}_{ij} \right)^2 \right)^{\frac{1}{2}} \tag{4.8}$$

This function must be recast to gain access to the spectrum of Σ. Let the spectral decomposition of the correlation matrix be defined by $\Lambda_\Sigma = V_\Sigma^T \Sigma V_\Sigma$. The Frobenius norm is invariant to changes of basis [29]. Taking V_Σ as a change of basis matrix, we obtain

$$P(\widehat{\Sigma}, \Sigma) = ||\Lambda_\Sigma - \Omega||_F \text{ where } \Omega = V_\Sigma^T \widehat{\Sigma} V_\Sigma \tag{4.9}$$

What does Ω look like? Suppose we let Σ be Σ_o which has been defined in Section 4.1.1. Remember that Σ_o is simply $\widehat{\Sigma}$ with the negative eigenvalues set to zero. The correlation matrix can be defined by

$$\Sigma = \sqrt{T} V_{\widehat{\Sigma}}^T \Lambda'_{\widehat{\Sigma}} V_{\widehat{\Sigma}} \sqrt{T} \tag{4.10}$$

Upon comparing this expression to the spectral decomposition of Σ, it is clear that $V_\Sigma = V_{\widehat{\Sigma}} \sqrt{T}$ and it becomes trivial to show that $\Omega = T\Lambda_{\widehat{\Sigma}}$. We can make two observations regarding Ω. First, Ω is diagonal if $\Sigma = \Sigma_o$. The optimization stage of the salvage prompts $\Sigma \approx \Sigma_o$. As a result, Ω will have the expected diagonal elements but with very small off-diagonal elements. Second, the corresponding elements of Ω and $\Lambda_{\widehat{\Sigma}}$ will have the same sign through noting $T_i > 0$. Consequently, we can determine if $\widehat{\lambda}_i$ lies to the right or left of zero based upon the diagonal of Ω. This observation motivates our modification to the proximity function.

We begin by expressing (4.9) in summation notation.

$$P(\widehat{\Sigma}, \Sigma) = \left(\sum_{i=1}^{d} (\lambda_i - \Omega_{ii})^2 + \sum_{i \neq j} (\Omega_{ij})^2 \right)^{\frac{1}{2}} \tag{4.11}$$

Let us agree that $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_d$ and likewise for $\widehat{\lambda}_i$. We will also say that $\widehat{\Sigma}$ has

M eigenvalues which are less than zero. The proximity function can be rexpressed as

$$P(\widehat{\Sigma}, \Sigma) = \left(\underbrace{\sum_{i=1}^{M} (\lambda_i - \Omega_{ii})^2}_{(\star)} + \sum_{i=M+1}^{d} (\lambda_i - \Omega_{ii})^2 + \sum_{i \neq j} (\Omega_{ij})^2 \right)^{\frac{1}{2}} \tag{4.12}$$

We now shift our focus to the $(\star)$ term. Specifically, we will examine one summand which will be denoted by $g(\lambda_i) = (\lambda_i - \Omega_{ii})^2$. Recognize that $g(\lambda_i)$ is an upward opening parabola with a single root at Ω_{ii}. Given our partitioning of summations, we know that $\Omega_{ii} < 0$. Recalling the "constraint" on the spectrum ($\lambda_i \geq 0$), we can conclude that $g(\lambda_i)$ will be minimized when $\lambda_i = 0$. The minimization of the proximity function constitutes simultaneously minimizing all of the summands. Hence, singularity arises from some of the λ_i's being driven to zero.

Let us pose a modified form of g which we will call $\widetilde{g}$. Suppose that $\widetilde{g}(\lambda_i) = w_i(\lambda_i - p_i)$ where $w_i > 0$ and $p_i > 0$. Following the same logic as before, we can conclude λ_i will be driven to p_i. Therefore, we can control how close Σ comes to singularity. The purpose of w_i is allow for more weight to be assigned to this term. If the magnitude of $(\lambda_i - p_i)$ were small, then the optimization may terminate before λ_i is sufficiently close to p_i. In sum, the modified cost function we propose is

$$\widetilde{P}(\widehat{\Sigma}, \Sigma) = \left(\sum_{i=1}^{M} w_i \left(\lambda_i - p_i\right)^2 + \sum_{i=M+1}^{d} (\lambda_i - \Omega_{ii})^2 + \sum_{i \neq j} (\Omega_{ij})^2 \right)^{\frac{1}{2}} \tag{4.13}$$

Choosing w_i and p_i is at the practitioner's discretion. Note that if these parameters are chosen carelessly, then $||\Sigma^* - \widehat{\Sigma}||_F$ may be exceedingly large or the solver may not converge. We have developed the following heuristics which have been used through

out this work.

Choice of p_i: For a moment, take $M = 1$. The idea is to choose p_1 such that a sufficiently small C is realized. A rough approximation of C is $C \approx d/p_1$. Using this relationship, p_1 can be computed for a desired C. Let us now consider when $M > 1$. We recommend that the p_i's be staggered ($p_1 < p_2 < \ldots < p_M$). By definition, Σ cannot be defective; that is, it must have d distinct λ_i. Driving multiple eigenvalues to the same value would be senseless.

Choice of w_i: The solver puts more emphasis on matching up the larger λ_i. These effects are not noticeable when d is small. When d is large however, the solver may terminate before λ_i is close to p_i. The weighting, w_i, can be used to cope with this problem. We recommend that $w_i = 10^{\gamma}/p_i$ where γ is the order of magnitude of $d/2$. In other words, γ is the exponent in normalized scientific notation. This policy will ensure that an average amount of emphasis is given to matching λ_i to p_i.

We will now demonstrate that this modified proximity yields a well-conditioned Σ^* with a minute sacrifice in distance. The MC method used at the end of Section 4.2.1 will also be used here. Again, we will take $d = 5$ and use 1000 samples. A small d helps to facilitate the convergence of solver in each iteration. First, the simulation was run using the classic proximity function, $P(\widehat{\Sigma}, \Sigma)$. The distributions of $\log(C)$ and $||\Sigma^* - \widehat{\Sigma}||_F$ are displayed in Figure 4.3. Second, the simulation was run for the modified proximity function. The parameters were set as $p_1 = 10^{-6}$ and $p_2 = 10^{-4}$ where p_2 was used when necessary. The results are shown in Figure 4.4.

In Figure 4.3 (i), we see that $\log(C)$ is centred around 16. Hence, the classic proximity function can be regarded as frequently producing nearly-singular correlation

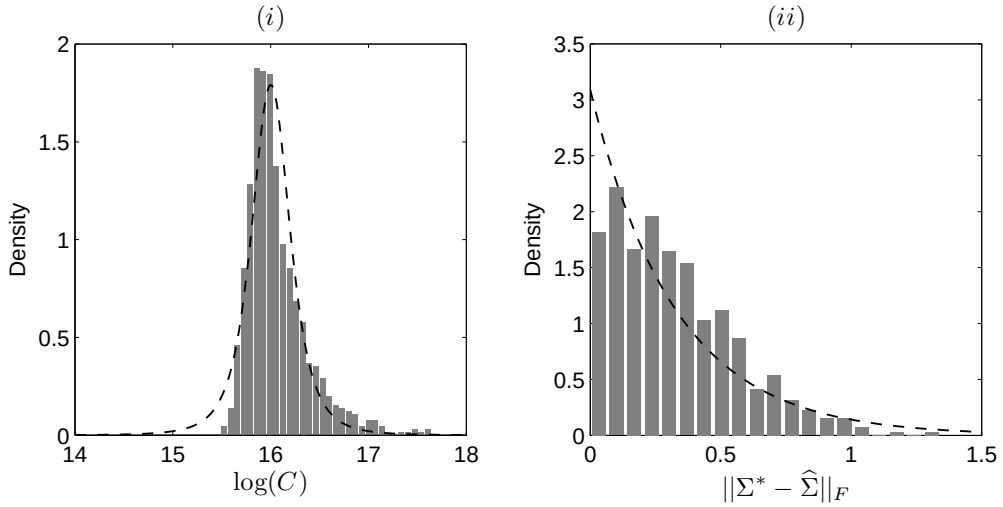

Figure 4.3: Distribution of $\log(C)$, shown in (i), and $||\Sigma^* - \widehat{\Sigma}||_F$, shown in (ii), using the classic proximity function

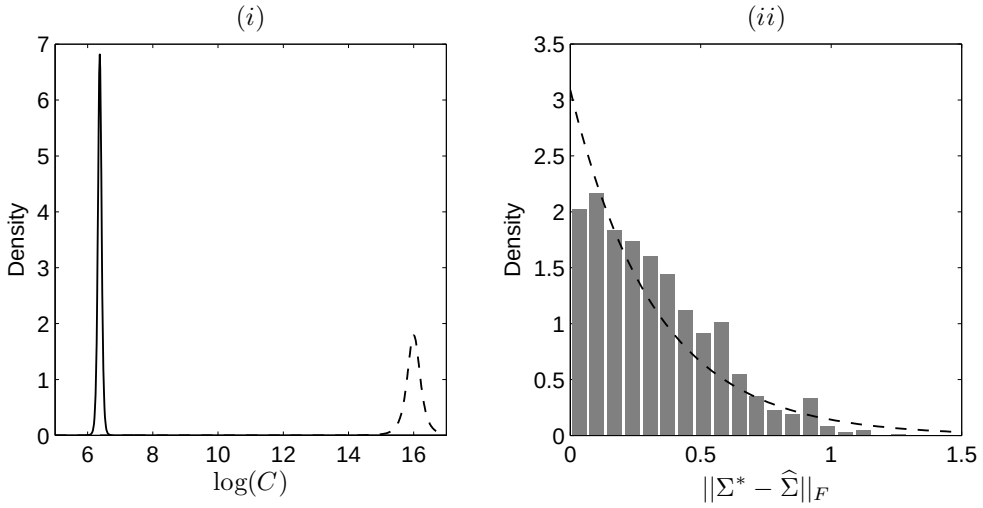

Figure 4.4: Distribution of $\log(C)$, shown in (i), and $||\Sigma^* - \widehat{\Sigma}||_F$, shown in (ii), using the modified proximity function

matrices. When we examine Figure 4.4 (i), it is clear that the modified proximity function produces a substantial shift away from singularity. Let us now consider the changes to $||\Sigma^* - \widehat{\Sigma}||_F$ caused by a different proximity function. Histogram (ii) in both Figures 4.3 and 4.4 bare a strong resemblance. Discrepancies between the two histograms indicate the distance which has been sacrificed to avoid invertibility. This effect appears to be minimal but we must confirm this statement. A two-sample Kolmogorov-Smirnov test [46] was performed on the data used to generate (ii) of Figures 4.3 and 4.4. The test rendered a p-value of 0.9125 which indicates that both of these sample sets likely came from the same distribution. Hence, there is a minimal sacrifice in cost when using the modified proximity function.

Example of Salvaging

Let us consider a small problem ($d = 3$) to gain insight into this algorithm. Begin by expressing a correlation matrix as follows

$$\Sigma = \begin{bmatrix} 1 & \rho_{12} & \rho_{13} \\ \rho_{12} & 1 & \rho_{23} \\ \rho_{13} & \rho_{23} & 1 \end{bmatrix} \tag{4.14}$$

Again, we will distinguish the elements of a pseudo-correlation matrix using a hat, $\widehat{\ }$. Suppose that $\widehat{\rho}_{12} = \widehat{\rho}_{23} = 0.9$. We will salvage $\widehat{\Sigma}$ using different values of $\widehat{\rho}_{13}$. Table 4.1 displays the elements of the salvaged correlation matrix along with $\widehat{\lambda}_{min}$. Note that in the case of $\widehat{\rho}_{13} = 0.9$, the resulting matrix is actually positive definite and so no salvaging is required.

This table illustrates how the algorithm works. The elements of the correlation

Table 4.1: A simple example of salvaging

$\widehat{\rho}_{13}$	$\widehat{\lambda}_{min}$	ρ^*_{12}	ρ^*_{13}	ρ^*_{23}
-0.9	-0.8000	0.5000	-0.5000	0.5000
-0.45	-0.5175	0.6194	-0.2327	0.6194
0	-0.2728	0.7424	0.1023	0.7424
0.45	-0.0675	0.8585	0.4739	0.8585
0.9	0.1000	0.9000	0.9000	0.9000

matrix are adjusted to be relatively close to the pseudo-correlation matrix we pose. In light of our discussion in Section 4.2.2, we see that some pseudo-correlation matrices do not salvage as well as other matrices. Again, we see that $\widehat{\lambda}_{min}$ is an indicator of the success of a salvage. Additionally, we see that $\rho^*_{12} = \rho^*_{23}$ which is in agreement with our specification, $\widehat{\rho}_{12} = \widehat{\rho}_{23}$.

4.4 SA of the Williams-Otto Plant

Model Description

We will now demonstrate how the salvaging strategy developed in this chapter fits into SA. The economic feasibility of a chemical plant is closely related to the kinetic parameters of the reactions which are involved. Faster reactions lead to smaller processing vessels and higher rates of productivity. Ultimately, the kinetic parameters play a critical role in determining a plant's feasibility. We will consider a hypothetical plant, known as the Williams-Otto (WO) Plant.

This model has been a benchmark in optimization literature. It is well-suited for our purposes because it has single output and a reasonable number of inputs ($d = 6$). Several variations of this problem exist. We have opted to use the model as posed

Table 4.2: Molecular weights of species (WO plant)

Species	MW (lbs mol^{-}1)
A	100
B	100
C	200
P	200
E	300
G	100

by Ray and Szekely [58]. Note that the model uses non-conventional units in their presentation. These conventions also will be used here.

The plant is designed such that the following reaction system is exploited to produce P.

$$A + B \xrightarrow{k_1} C \tag{4.15}$$

$$C + B \xrightarrow{k_2} P + E \tag{4.16}$$

$$P + C \xrightarrow{k_3} G \tag{4.17}$$

where

$$k_i = X_i \exp\left(\frac{-X_{i+3}}{T_{reactor}}\right) \quad \text{for} \;\; i = 1, 2, 3 \tag{4.18}$$

We will consider the kinetic parameters $(X_1, \ldots, X_6)$ to be the inputs of the model. The molecular weight of each species is displayed in Table 4.2 and the nominal values of the inputs are displayed in Table 4.3.

The process flow diagram of the WO plant is shown in Figure 4.5. Before we launch into our description of the plant, some notation will be declared so that we can distinguish between different mass flow rates. As an example, consider $F_E^{(1)}$.

Table 4.3: Nominal kinetic parameters (WO plant)

Input	Nominal Value
1	5.9755×10^9 L lbs_A^{-1} h^{-1}
2	2.5962×10^{12} L lbs_B^{-1} h^{-1}
3	9.6283×10^{15} L lbs_C^{-1} h^{-1}
4	12000^oR
5	15000^oR
6	20000^oR

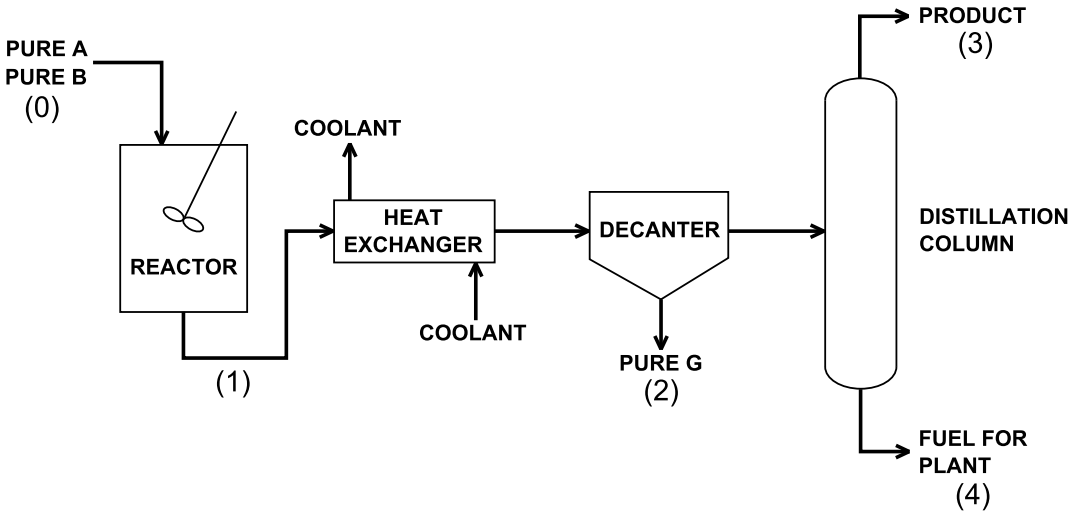

Figure 4.5: Process flow diagram of the WO plant

Notice that some of the streams have been assigned numbers in Figure 4.5. The superscript corresponds to the stream number. The subscript indicates the chemical species. The absence of a subscript indicates a total mass flow rate. In the case of our example, we are considering the mass flow rate of E in stream (1).

The process begins by feeding A into the reactor at a rate of 10^4 lbs/hr. B also enters the reactor at a mass flow rate twice that of A. The reactor is operated isothermally ($T_{reactor} = 650^oR$) and at a fixed volume ($\rho V_{reactor} = 3000$ lbs). Each reaction rate is k_i multiplied by the product of the two reactant concentrations.

The effluent passes through a heat exchanger which cools the stream. The drop in temperature partitions the flow into two phases. One phase is pure G while the other phase contains the remaining species. The decanter separates out the G phase ($F^{(2)} = F_G^{(1)}$) while the other phase enters a distillation column. The volatility of P results in the product exiting the overhead stream. A portion of the bottoms can be used as fuel for the plant. The separation efficiently can be summarized by the relationship $F_P^{(1)} - F_P^{(3)} = 0.1F_E^{(1)}$. Note that the non-zero right hand side of this expression indicates the presence of an azeotrope between P and E. The number of hours of operation per year, T, will be assumed to be 8400 hours.

The annual rate of return on fixed investment is the output of this model. In Ray and Szekely's presentation, it is defined as

$$Y = 100 \times \left[T\left(0.876\gamma_1 - \gamma_2\right) - 0.1\gamma_3 - 2.2F^{(2)}\right]/\gamma_3 \tag{4.19}$$

where

$$\begin{aligned}
\gamma_1 &= 0.3F^{(3)} + 0.0068F^{(4)} \\
\gamma_2 &= 0.02F_A^{(0)} + 0.03F_B^{(0)} + 0.01F^{(2)} \\
\gamma_3 &= 600\rho V_{reactor}
\end{aligned}$$

Assuming steady-state operation, we can derive a system of equations from the component balances. For the sake of brevity, this system of equations will not be presented here but the original work can be consulted for details [58, p.147]. The mass flow rates required for the computation of Y can be found by solving this system. The nonlinearities present in the system means that a numerical algorithm is required.

The composition of the reactor's effluent is dictated by the kinetic parameters. We will now investigate how these parameters effect the annual rate of return on fixed investment via SA. Two scenarios will be considered. First, the kinetic parameters will be assumed to be normally distributed. This will facilitate a comparison between the three forms of sensitivity we have addressed. Second, we will consider the case where the inputs have non-normal marginals. By recalling Remark 3, we see that only CB sensitivity can be used in this scenario. This scenario will illustrate how the assumption of normality can impact SA.

SA with Normal Inputs

As with the CP model in Chapter 2, we will consider an independent case and a correlated case. It will be assumed that the nominal value of an input constitutes its mean and that the standard deviation is fifteen percent of its nominal value. We must define a correlation matrix to be used for the correlated case. Suppose that the joint behaviour which we would like the inputs to follow is the "correlation matrix" shown in Table 4.4. While this construction appears to be somewhat arbitrary it is not without reason. The diagonals of 0.9 imply that for each reaction the pre-exponential factor and the activation energy are strongly correlated. This is frequently encountered in kinetic parameter estimation problems [55]. To necessitate the use of salvaging, other non-zero correlations have been specified so that the matrix is not positive semi-definite.

Upon inspection of its spectrum, we see that there is one negative eigenvalue. Hence, the matrix which we have specified is actually a pseudo-correlation matrix. This matrix was salvaged using the modified proximity function ($p_1 = 10^{-6},\ w_1 =$

Table 4.4: Pseudo-correlation matrix for the WO plant

Input	1	2	3	4	5	6
1	1	-0.6	0	0.9	0	0
2		1	0	0	0.9	0
3			1	0	0	0.9
4				1	0	0
5					1	0.4
6						1
$\widehat{\lambda}_i$	-0.2980	0.0388	0.4157	1.5843	1.9612	2.2980

Table 4.5: Salvaged correlation matrix for the WO plant

Input	1	2	3	4	5	6
1	1	-0.4863	-0.0295	0.8378	-0.0923	0.0485
2		1	-0.0353	-0.0739	0.7904	0.0545
3			1	0.0190	0.0284	0.8864
4				1	0.0599	-0.0296
5					1	0.3559
6						1
λ_i	9.0757×10^{-7}	0.0037	0.3469	1.5126	1.9270	2.2097

$1/p_1$) and the resulting correlation matrix is shown in Table 4.5. Indeed, the salvaged matrix is well conditioned ($\log(C) = 6.3865$). When we compare Tables 4.4 and 4.5, the workings of the salvage algorithm becomes apparent. While no element matches up perfectly, there are no large disparities either. Again, we see that small changes to each of the elements can give rise to a valid correlation matrix.

The same machine and computing environmental used for the CP model was also used here. The double-loop MC simulations required for the computation of VB and MF sensitivities were run taking $N = 1000$ and $R = 400$. These computations took approximately 5 hours to complete. We could not find a suitable parametric family

Table 4.6: Sensitivities of the WO plant with normal inputs

	η_i		δ_i		ψ_i	
Input	IA	RCS	IA	RCS	IA	RCS
1	0.0018	54.0913	0.0454	1.8744	0.0623	3.7406
2	0.0014	88.8929	0.0452	2.5932	0.0691	4.2197
3	0.0015	194.8327	0.0444	5.7404	0.0886	6.2915
4	0.2830	-0.1830	0.3281	-0.1483	0.5213	-0.0088
5	0.2047	-0.3751	0.2143	-0.2030	0.3927	-0.1988
6	0.2945	-0.2437	0.2889	-0.0538	0.6513	-0.0036

to use in the computation of MF sensitivity. As proposed by Borgonovo *et al.*, we opted to use kernel density estimation instead [2]. The single-loop MC simulation required for the computation of CB sensitivity used the same N and took only 8 seconds. Table 4.6 displays the sensitivities for each input.

We begin by considering the independent case. The first three inputs, which correspond to the pre-exponential factors, have been assessed to have small sensitivities. The output is very sensitive to the activation energies (inputs 4-6). In particular, the activation energies of (4.15) and (4.17) are slightly more sensitive than that of (4.16). These statements can be made using any one of the three types of sensitivity and so they can be considered to perform equally as well. The advantage of VB sensitivity is that we can gain insight into higher-order sensitivities. The sum of all η_i is equal to 0.7869 which means that they are more substantial here than in the CP example. However, higher-order sensitivities are infeasible to compute and so first-order sensitivities will form the basis of the analysis.

Let us now address the correlated case through looking at RCS. Looking at the pseudo-correlation matrix in Table 4.4, we see a band of 0.9. The position of this band means that for a given reaction, the pre-exponential factor and the activation

energy are strongly correlated. In Chapter 2, we showed that when two inputs are strongly correlated, the difference in sensitivities will be small. In the independent case, we found that the model is much more sensitive to activation energies than pre-exponential factors. It follows that correlation will drive up the sensitivities of the pre-exponential factors. The truth of this statement is indicated through the large RCSs for inputs 1-3.

Like in the CP model, we see that the VB sensitivity tends to have a much larger RCS than the other two types of sensitivity. This contradicts our findings from when we analyzed the linear model in Chapter 2. There, we saw that the MF sensitivity changed more with correlation than VB/CB sensitivities did. Clearly, the nonlinearities in the model make VB sensitivity very susceptible to changes in input correlation. Is this a desirable property for a sensitivity to have? At the end of Chapter 3, we showed that imposing a correlation structure on an input vector is not a precise process. Basing an analysis on a sensitivity, which can drastically change with input correlation, would be poor practise. The RCS of the MF and CB sensitivities are much smaller in both the CP model and the WO plant. Therefore, they are more suitable options. In light of its smaller computational cost, CB sensitivity is the most favourable option.

SA with Non-normal Inputs

We will now consider the scenario where the marginals of the inputs follow non-normal distributions. Suppose the pre-exponential factors (inputs 1-3) follow a symmetric triangular distribution. The vertices exist at the nominal value and the nominal value plus/minus three standard deviations. The activation energies (inputs 4-6)

Table 4.7: SA of the WO plant using non-normal inputs

	Normal		Non-normal	
Input	IA	RCS	IA	RCS
1	0.0623	3.7406	0.3854	0.9359
2	0.0691	4.2197	0.3564	0.3603
3	0.0886	6.2915	0.1103	3.4700
4	0.5213	-0.0088	0.8537	-0.0490
5	0.3927	-0.1988	0.6725	-0.3639
6	0.6513	-0.0036	0.7167	-0.3245

follow a uniform distribution with boundaries at the nominal value plus/minus three standard deviations. The joint behaviour of the input vector will be encoded through a normal copula which uses the same correlation matrix as in the previous SA.

A single-loop MC simulation was run for when the inputs are independent and when codependencies exist ($N = 1000$). The sample correlation matrix of the inputs was checked against Table 4.5. It was deemed to be an adequate match. The CB sensitivities are displayed in Table 4.7. Note that the Normal columns are the same as in Table 4.6. The sensitivities from the SA using normal marginals has also been displayed for comparison purposes.

Note that in the non-normal case, the marginals were designed to be distributed over approximately the same interval as the normal case. What is meant by approximately? A normal distribution has an infinite support but the triangular and uniform distributions used here have finite supports. For a normally distributed variate, we know that 99.7% of samples will lie within three standard deviations of the mean. We have set the bounds of both non-normal distributions to be at the mean plus/minus three standard deviations. Hence, the probability density is distributed over roughly the same interval and only the change in the shape of the distribution is causing a

change in the SA.

Clearly, this change in distribution shape can greatly impact the SA. Let us consider the independent case first. All inputs were assessed to be more sensitive here than in the SA with normal marginals. We see a particularly interesting feature with the pre-exponential factors. In the case of non-normal inputs, the model is much more sensitive to the first and second pre-exponential factors. In the case of normal inputs, the sensitivity of all three pre-exponential factors are relatively equal. Let us now turn our attention to the RCSs. For all inputs, the sign of the RCS is the same for the normal and non-normal cases. However, notice the large differences in the magnitude of RCS. This indicates that input codependencies influence sensitivity to a different extent based upon the marginals. Given the marginal behaviours used here, we see that input codependencies impact the analysis less so than in the case of normal marginals. The above observations illustrate the importance of not blindly assuming that the inputs follow a normal distribution.

Summary

Recall that in Chapter 3, we showed how one can arbitrarily choose the marginal behaviour of each input while still giving a practitioner the freedom to specify the joint behaviour. In this chapter, we have addressed the technicalities of establishing this joint behaviour. Specifically, we have posed an algorithm which facilitates a practitioner's choice of correlation matrix. Using the procedures established in Chapters 3 and 4, a practitioner has much more flexibility in the generation of input samples for MC simulations. Coupling this sampling protocol with CB sensitivities yields a general methodology to SA. The WO plant was used to illustrate this methodology.

Chapter 5

Conclusion

The main contribution of this work is assessing the consequences of recklessly assuming independent inputs in global SA. It has been shown that input codependencies complicate the interpretation of sensitivity. Of the few authors which have addressed this problem, most suggest that some form of input orthogonalization be used. While this approach is computationally lean, the sensitivities remain difficult to interpret.

In this work, we propose a different approach to the problem at hand. We recognize that input codependencies will inherently make the interpretation of sensitivities difficult. Our goal was to quantify the extent to which codependencies convolute the analysis. This was realized through defining RCS. To make this computationally feasible, it was necessary to define a new form of sensitivity (CB sensitivity). This methodology is formalized through the following steps.

Step 1: Begin by assigning a continuous univariate PDF to each input. These assignments can either be based upon data, expert judgement or a combination of the two.

STEP 2: Pose a correlation matrix for the inputs. Again data, expert judgement or a combination of the two can be used. Check the spectrum of the matrix for positive definiteness. If the correlation matrix is positive definite, then proceed to Step 3. If this is not the case, use the salvage algorithm discussed in Chapter 4 to find a valid correlation matrix.

STEP 3: Execute two single-loop MC simulations. Assume that the inputs are independent in the first simulation. In the second simulation, use the copula-based sampling algorithm described in Chapter 3 to generate codependent input samples.

STEP 4: For the simulation assuming independence, compute the CB sensitivity for each input using the procedure discussed in Section 2.4.4. Repeat this process for the simulation where codependent inputs are used.

STEP 5: Compute the RCS (as expressed in Definition 4) from the two sets of sensitivities. Conclusions can be drawn from the CB sensitivities for independent inputs and the RCSs. The CB sensitivities forms the basis of the analysis. The RCS indicates how much its corresponding sensitivity can be trusted. For example, a sensitivity with a small RCS should be trusted more than a sensitivity with a large RCS.

This work is novel in several regards. First, it was observed that the classical definition of sensitivity is a measure of dependence. This lead us to propose a new definition of sensitivity which is based upon a nonlinear dependence measure. Second, copula-based sampling can be integrated into the simulation required to compute this new sensitivity. Copulas provide a flexible framework for modelling codependent inputs. To our knowledge, this has never been done in the context of SA. The final

element of novelty involves the salvage algorithm which supports copula-based sampling. Few salvage algorithms have addressed the inherent numerical issue with the correlation matrix produced. Some authors have advocated constraining the optimization problem to avoid this issue. In this work, we achieve the same effect but using a simple modification to the cost function being optimized.

While this method shows promise, further work is necessary. Many avenues can be pursued but two will presented here. First, our methodology has been performed on only one model, the Williams-Otto Plant. The nature of global SA requires that no assumptions can be made about the model's structure. It follows that our method can only be validated by analysing an array of models and comparing conclusions reached with the conclusions of more traditional analyses. Second, only one type of copula, the normal copula, has been considered in this work. There exists a growing array of copulas which are available. Other copulas which are relevant to SA should be identified. Increased flexibility in the modelling of inputs will help to facilitate more accurate analyses.

Bibliography

[1] E. Borgonovo, "A new uncertainty importance measure," *Reliability Engineering and System Safety*, vol. 92, pp. 771–784, 2007.

[2] E. Borgonovo, W. Castaings, and S. Tarantola, "Moment Independent Importance Measures: New Results and Analytical Test Cases," *Risk Analysis*, vol. 31, no. 3, pp. 404–428, 2011.

[3] R. Borsdorf and N. Higham, "A preconditioned Newton algorithm for the nearest correlation matrix," *IMA Journal of Numerical Analysis*, vol. 30, pp. 94–107, 2010.

[4] M. Cario and B. Nelson, "Modeling and Generating Random Vectors with Arbitrary Marginal Distributions and Correlation Matrix," Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, Tech. Rep., 1997.

[5] M. Chun, S. Han, and N. Tak, "An uncertainty importance measure using a distance metric for the change in a cumulative distribution function," *Reliability Engineering and System Safety*, vol. 70, pp. 313–321, 2000.

[6] R. Clemen and T. Reilly, "Correlations and Copulas for Decision and Risk Analysis," *Management Science*, vol. 45, no. 2, pp. 208–224, 1999.

[7] R. Cukier, C. Fortuin, K. Shuler, A. Petschek, and J. Schaibly, "Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficents. I Theory," *The Journal of Chemical Physics*, vol. 59, pp. 3873–3878, 1973.

[8] P. Delicado. (2007, June) Principal curves of oriented points. [Online]. Available: http://www-eio.upc.es/~delicado/PCOP/

[9] P. Delicado and M. Smrekar, "Measuring non-linear dependence for two random variables distributed along a curve," *Statistics and Computing*, vol. 19, no. 3, pp. 255–269, 2009.

[10] D. Drouet Mari and S. Kotz, *Correlation and Dependence*. Imperial College Press, 2001.

[11] M. Eaton, *Multivariate Statistics: A Vector Space Approach*. Institute of Mathematical Statistics, 2007, ch. The Normal Distribution on a Vector Space, pp. 103–131.

[12] P. Embrechts, A. McNeil, and D. Straumann, *Risk Management: Value at Risk and Beyond*. Cambridge University Press, 2002, ch. Correlation and Dependence in Risk Management: Properties and Pitfalls, pp. 176–223.

[13] S. Ferson, R. Nelsen, J. Hajagos, D. Berleant, J. Zhang, W. Tucker, L. Ginzburg, and W. Oberkampf, "Dependence in probabilistic modeling, Dempster-Shafer theory, and probability bounds analysis," Sandria National Laboratories, Tech. Rep. SAND2004-3072, 2004.

[14] N. Fisher, *Encyclopedia of Statistical Sciences*. Wiley, 1997, ch. Copulas, pp. 159–163.

[15] R. Fisher, *Statistical Methods For Research Workers*, F. Crew and D. Cutler, Eds. Oliver and Boyd, 1932.

[16] M. Fréchet, "Sur les tableaux de corrélation dont les marges sont données," *Annales de 'lUniversite de Lyon Sciences*, vol. 4, pp. 53–84, 1951.

[17] E. Frees and E. Valdez, "Understanding relationships using copulas," *North American Acturarial Journal*, vol. 2, pp. 1–25, 1998.

[18] Y. Fujikoshi, V. Ulyanov, and R. Shimizu, *Mulitvariate Statistics: High-Dimensional and Large-Sample Approximations*. Wiley, 2010.

[19] C. Genest and A. Favre, "Everything You Always Wanted to Know about Copula Modeling but Were Afraid to Ask," *Jornal of Hydrologic Engineering*, vol. 12, no. 4, pp. 347–368, 2007.

[20] C. Genest, M. Gendron, and M. Bourdeau-Brein, "The Advent of Copulas in Finance," *The European Journal of Finance*, vol. 15, pp. 609–618, 2009.

[21] S. Ghosh and S. Henderson, "Behavior of the NORTA Method for Correlated Random Vector Generation as the Dimension Increases," *ACM Transactions on Modeling and Computer Simulation*, vol. 13, pp. 273–294, 2003.

[22] D. Hamby, "A review of techniques for parameter sensitivity analysis of enviornment models," *Enviornmental Monitoring and Assessment*, vol. 32, pp. 135–154, 1994.

[23] J. Hammersley and D. Handscomb, *Monte Carlo Methods*. Wiley, 1964.

[24] T. Hastie and W. Stuetzle, "Principal Curves," *Journal of the American Statistical Associtation*, vol. 84, no. 406, pp. 502–516, 1989.

[25] N. Higham, "Computing the nearest correlation matrix - a problem from finance," *IMA Journal of Numerical Analysis*, vol. 22, pp. 329–343, 2002.

[26] W. Hoeffding, *The Collected Works of Wassily Hoeffding*. Springer, 1940, ch. Scale-invariant correlation theory, pp. 57–107.

[27] T. Homma and A. Saltelli, "Importance measures in global sensitivity analysis of nonlinear models," *Reliability Engineering and System Safety*, vol. 52, pp. 1–17, 1996.

[28] S. Hora and R. Iman, "Comparison of Maximus/Bounding and Bayes/Monte Carlo for fault tree uncertainty analysis," Sandria National Laboratories, Tech. Rep. SAND85-2839, 1986.

[29] R. Horn and C. Johnson, *Matrix Anal*, 2nd ed. Cambridge University Press, 2012.

[30] R. Iman and W. Conover, "A distribution-free approach to inducing rank correlation among input variables," *Communications in Statistics - Simulation and Computation*, vol. 11, no. 3, pp. 311–334, 1982.

[31] R. Iman and S. Hora, "A Robust Measure of Uncertainty Importance for the Use in Fault Tree System Analysis," *Risk An*, vol. 10, no. 3, pp. 401–406, 1990.

[32] J. Jacques, C. Lavergne, and N. Devictor, "Sensitivity analysis in presence of model uncertainty and correlated inputs," *Reliability Engineering and System Safety*, vol. 91, pp. 1126–1134, 2006.

[33] M. Johnson and J. Ramberg, "Transformations of the multivariate normal distribution with applications to simulation," Los Alamos Scientific Laboratory, Tech. Rep. LA-UR-77-2595, 1978.

[34] N. Johnson, "Bivariate Distributions Based on Simple Translation Systems," *Biometrika*, vol. 36, no. 3/4, pp. 297–304, 1949.

[35] H. Kaiser and K. Dickman, "Sample and population score matrices and sample correlation matrices from an arbitrary population correlation matrix," *Psychometrika*, vol. 27, pp. 179–182, 1962.

[36] A. Kercheval, "On Rebonato and Jäkel's parameterization mathod for finding nearest correlation matrices," in *Proceedings of the International Conference on Risk Management and Quantitative Approaches to Finance*, 2005.

[37] A. Kolmogorov, *Grundbegriffe der Wahrscheinlichkeitsrechnung.* Berlin: Springer, 1933.

[38] O. Ledoit and M. Wolf, "Improved estimation of the covariance matrix of stock returns with an application to portfolio selection," *Journal of Empirical Finance*, vol. 10, pp. 603–621, 2003.

[39] D. X. Li, "On default correlation: A copula function approach," The RiskMetrics Group, Tech. Rep. 99-07, 2000.

[40] G. Li, H. Rabitz, P. Yelvington, O. Oluwole, F. Bacon, C. Kolb, and J. Schoendorf, "Global sensitivity analysis for systems with independent and/or correlated inputs," *Journal of Physical Chemistry A*, vol. 114, pp. 6022–6032, 2010.

[41] S. Li and J. Hammond, "Generation of Pseudorandom Numbers with Specified Univariate Distributions and Correlation Coefficents," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-5, no. 5, pp. 557–561, 1975.

[42] Q. Liu and T. Homma, "A New Importance Measure for Sensitivity Analysis," *Journal of Nuclear Science and Technology*, vol. 47, no. 1, pp. 53–61, 2010.

[43] P. Lurie and M. Goldberg, "An Approximate Method for Sampling Correlated Random Variables from Partially-Specified Distributions," *Management Science*, vol. 44, pp. 203–218, 1998.

[44] B. Mandelbrot, "The Variation of Certain Speculative Prices," *The Journal of Business*, vol. 36, no. 4, pp. 394–419, 1963.

[45] T. Mara and S. Tarantola, "Variance-based sensitivity indices for models with dependent inputs," *Reliability Engineering and System Safety*, vol. 107, pp. 115–121, 2012.

[46] F. Massey, "The Kolmogorov-Smirnov test for goodness of fit," *Journal of the American Statistical Association*, vol. 46, no. 253, pp. 68–78, 1951.

[47] A. Mathai and S. Provost, *Quadratic Forms in Random Variables*. Marcel Dekker, 1992, ch. Quadratic Forms in Random Variables, pp. 25–84.

[48] M. McKay, "Nonparametric Variance-based Methods of Assessing Uncertainty Importance," Los Alamos National Laboratory, New Mexico, Tech. Rep. 875450600, 1997.

[49] S. Mildenhall, "Correlation and Aggregate Loss Distributions With An Emphasis On The Iman-Conover Method," Casualty Actuarial Society, Tech. Rep., 2005.

[50] S. Mishra, "The nearest correlation matrix problem: Solution by differential evolution method of global optimization," Department of Economics, North-Eastern Hill University (India), Tech. Rep., 2007.

[51] R. Nelsen, *An Introduction to Copulas*. Springer, 1999.

[52] K. Ogata, *Modern Control Engineering*, 4th ed. Prentice Hall, 2002.

[53] C. Park and K. Ahn, "A new approach for measuring uncertainty importance and distributional sensitivity in probabilistic safety assessment," *Reliability Engineering and System Safety*, vol. 46, pp. 253–261, 1994.

[54] A. Patton, "Modelling asymmetric exchange rate dependence," *International Economic Review*, vol. 47, pp. 527–556, 2006.

[55] D. Pritchard and D. Bacon, "Prospects for reducing correlations among parameter estimates in kinetic models," *Chemical Engineering Science*, vol. 33, pp. 1539–1543, 1978.

[56] H. Qi and D. Sun, "A Quadratically Convergent Newton Method for Computing the Nearest Correlation Matrix," *SIAM Journal on Matrix Analysis and Applications*, vol. 28, no. 2, pp. 360–385, 2006.

[57] F. Rapisarda, D. Brigo, and F. Mercurio, "Parameterizing correlations: a geometric interpretation," *IMA Journal of Management Mathematics*, vol. 18, pp. 55–73, 2007.

[58] W. Ray and J. Szekely, *Process Optimization with Application to Metallurgy and Chemical Engineering*. Wiley, 1973, ch. Constrained Optimization Techniques, pp. 144–151.

[59] R. Rebonato and P. Jäkel, "The most general methodology to create a valid correlation matrix for risk management and option pricing purposes," Quantitative Research Centre of the NatWest Group, Tech. Rep., 1999.

[60] T. Reilly, "Sensitivity Analysis for Dependendent Variables," *Decision Sciences*, vol. 31, no. 3, pp. 551–572, 2000.

[61] A. Renyi, "New version of the probabilistic generalization of the large sieve," *Acta Mathematica Academiae Scientiarum Hungarica*, vol. 10, no. 1-2, pp. 217–226, 1959.

[62] A. Saltelli, "Making best use of model evaluations to compute sensitivity indicies," *Computer Physics Communications*, vol. 145, pp. 280–297, 2002.

[63] ——, "Sensitivity Analysis for Importance Assessment," *Risk Analysis*, vol. 22, pp. 579–590, 2002.

[64] A. Saltelli and R. Balado, "An alternative way to compute Fourier amplitude sensitivity test (FAST)," *Computation Statistics and Data Analysis*, vol. 26, pp. 445–460, 1998.

[65] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola, *Global Sensitivity Analysis: The Primer*. Wiley, 2008, ch. Variance-based Methods, pp. 155–176.

[66] ——, *Global Sensitivity Analysis: The Primer*. Wiley, 2008.

[67] A. Saltelli, S. Tarantola, and K.-S. Chan, "A Quantitative Model-Independent Method for Global Senstivity Analysis of Model Output," *Technometrics*, vol. 41, pp. 39–56, 1999.

[68] J. Schaibly and K. Shuler, "Study of the sensitivity of coupled reaction systems to uncertainties in rate coefficents. II Applications," *Journal of Chemical Physics*, vol. 59, pp. 3879–3888, 1973.

[69] K. Schottle and R. Werner, *Risk Analysis IV*. WIT Press, 2004, ch. Improving the "most general methodology to create a valid correlation matrix", pp. 701–712.

[70] A. Sklar, "Fonctions de répartition à n dimensions et leurs marges," *Paris Institute of Statistics*, vol. 8, pp. 229–231, 1959.

[71] I. Sobol', "Sensitivity estimation for nonlinear mathematical models," *Matematicheskoe Modelirovanie*, vol. 2, no. 1, pp. 112–118, 1990.

[72] S. Tarantola, "Variance-based methods," in *Third Summer School on Sensitivity Analysis on Model Output*, September 2004.

[73] N. Weiss, *A Course in Probability*. Addison-Wesley, 2005.

[74] C. Xu and G. Gertner, "Uncertainty and sensitivity analysis for models with correlated parameters," *Reliability Engineering and System Safety*, vol. 93, pp. 1563–1573, 2008.

[75] W. Yi and V. Bier, "An application of copulas to accident precursor analysis," *Management Science*, vol. 44, pp. S257–S270, 1998.

[76] W. Yu and T. Harris, "Parameter uncertainty effects on variance-based sensitivity analysis," *Reliability Engineering and System Safety*, vol. 94, pp. 596–603, 2009.

[77] Z. Zhang and L. Wu, "Optimal low-rank approximation to a correlation matrix," *Linear Algebra and its Applications*, vol. 364, pp. 161–187, 2003.

Appendix A

Derivation of η_i for LQ model

Derivation of V_i

From the definition of η_i in (2.2), it is natural to begin by working with $\mathrm{E}[Y|X_i = x_i^*]$. In light of Theorem 2, the sensitivity with respect to the d^{th} input can be derived without loss of generality. The use of this theorem implies that the inputs are normally distributed. Moreover, it will be assumed that X has a zero mean vector. Let the subscript $-d$ indicate a vector with the d^{th} component omitted. Let $v = [\Sigma_{12}^T \ \sigma_d^2]^T$ where $\sigma_d^2 = \mathrm{Var}[X_d]$.

The linear and quadratic components of (2.12) will be treated separately. We begin with the linear component which we will call Y_ℓ.

$$Y_\ell = \ell^T X \tag{A.1}$$

$$Y_\ell = \ell_d X_d + \ell_{-d}^T X_{-d} \tag{A.2}$$

$$\mathrm{E}[Y_\ell|X_d = x_d^*] = \ell_d x_d^* + \ell_{-d}^T \mathrm{E}[X_{-d}|X_d = x_d^*] \tag{A.3}$$

Theorem 2 can be used to evaluate $\mathrm{E}\left[X_{-d}|X_d = x_d^*\right]$. It follows that

$$\mathrm{E}\left[Y_\ell|X_d = x_d^*\right] = \frac{v^T\ell}{\sigma_d^2}x_d^* \tag{A.4}$$

The quadratic component, Y_Q, can be addressed in a similar fashion. The same notation used for the covariance matrix in Theorem 2 will be used to partition Q.

$$\begin{aligned} Y_Q &= X^TQX & \text{(A.5)}\\ Y_Q &= Q_{dd}{X_d}^2 + 2X_dQ_{12}^TX_{-d} + X_{-d}^TQ_{11}X_{-d} & \text{(A.6)}\\ \mathrm{E}\left[Y_Q|X_d = x_d^*\right] &= Q_{dd}{x_d^*}^2 + \underbrace{2Q_{12}^Tx_d^*\mathrm{E}\left[X_{-d}|X_d = x_d^*\right]}_{1} \\ &\quad + \underbrace{\mathrm{E}\left[X_{-d}^TQ_{11}X_{-d}|X_d = x_d^*\right]}_{2} & \text{(A.7)} \end{aligned}$$

The approach that was used to derive (A.4) can also be used to evaluate the expectation in Term 1. In Term 2, the expectation of a quadratic form is required. An expression for this expectation has been derived [47, p.50]. Using this result and what is known about the parameters of the conditioned distribution via Theorem 2, we obtain

$$\mathrm{E}\left[Y_Q|X_d = x_d^*\right] = \mathrm{trace}\left(Q_{11}\Sigma_{-d}\right) + \frac{1}{\sigma_d^4}\left\{Q_{dd}\sigma_d^4 + 2\sigma_d^2Q_{12}^T\Sigma_{12} + \Sigma_{12}^TQ_{11}\Sigma_{12}\right\}{x_d^*}^2 \tag{A.8}$$

where $\Sigma_{-d} = \Sigma_{11} - \Sigma_{12}\Sigma_{12}^T/\sigma_d^2$. The braced term can be condensed as follows

$$\mathrm{E}\left[Y_Q|X_d = x_d^*\right] = \mathrm{trace}\left(Q_{11}\Sigma_{-d}\right) + \frac{v^TQv}{\sigma_d^4}{x_d^*}^2 \tag{A.9}$$

Summing the conditional expectations of Y_ℓ and Y_Q, we obtain

$$\mathrm{E}\left[Y|X_d = x_d^*\right] = \mathrm{trace}\left(Q_{11}\Sigma_{-d}\right) + \frac{v^T\ell}{\sigma_d^2}x_d^* + \frac{v^TQv}{\sigma_d^4}x_d^{*2} \tag{A.10}$$

In its current form, $\mathrm{E}\left[Y|X_d = x_d^*\right]$ is not a random quantity. Define a random variable X_d^* which follows the same distribution as the d^{th} component of X. When we swap x_d^* for X_d^*, (A.10) becomes random and its variance can be taken.

$$\mathrm{Var}\left[\mathrm{E}\left[Y|X_d = X_d^*\right]\right] = \mathrm{Var}\left[\mathrm{trace}\left(Q_{11}\Sigma_{-d}\right) + \frac{v^T\ell}{\sigma_d^2}X_d^* + \frac{v^TQv}{\sigma_d^4}X_d^{*2}\right] \tag{A.11}$$

$$\begin{aligned}\mathrm{Var}\left[\mathrm{E}\left[Y|X_d = X_d^*\right]\right] &= \frac{(v^T\ell)^2}{\sigma_d^2} + \left(\frac{v^TQv}{\sigma_d^4}\right)^2\mathrm{Var}\left[X_d^{*2}\right] \\ &\quad +2\mathrm{Cov}\left[\frac{v^T\ell}{\sigma_d^2}X_d^*, \frac{v^TQv}{\sigma_d^4}X_d^{*2}\right]\end{aligned} \tag{A.12}$$

This expression can be simplified using properties of normally distributed random variables. First, we examine $\mathrm{Var}\left[X_d^2\right]$.

$$\begin{aligned}\mathrm{Var}\left[X_d^2\right] &= \mathrm{E}\left[X_d^4\right] - \left(\mathrm{E}\left[X_d^2\right]\right)^2; \quad \mathrm{E}\left[X_d^4\right] = 3\sigma_d^4 \text{ and } \mathrm{E}\left[X_d^2\right] = \sigma_d^2 \\ \mathrm{Var}\left[X_d^2\right] &= 2\sigma_d^4\end{aligned} \tag{A.13}$$

Second, we look at the covariance term which we will abbreviate as $\mathrm{Cov}\left[aX_d, bX_d^2\right]$.

$$\begin{aligned}\mathrm{Cov}\left[aX_d, bX_d^2\right] &= \mathrm{E}\left[abX_d^3\right] - \mathrm{E}\left[aX_d\right]\mathrm{E}\left[bX_d^2\right] = 0; \quad \mathrm{E}\left[X_d^3\right] = 0 \text{ and } \mathrm{E}\left[X_d\right] = 0 \\ \mathrm{Cov}\left[aX_d, bX_d^2\right] &= 0\end{aligned} \tag{A.14}$$

Using these above two statements, (A.12) reduces to our final expression for V_d.

$$V_d = \mathrm{Var}\left[\mathrm{E}\left[Y|X_d = X_d^*\right]\right] = \frac{1}{\sigma_d^4}\left\{\sigma_d^2(v^T\ell)^2 + 2(v^TQv)^2\right\} \tag{A.15}$$

Derivation of $\mathrm{Var}\left[Y\right]$

We begin with a standard definition of variance.

$$\mathrm{Var}\left[Y\right] = \mathrm{E}\left[Y^2\right] - \left(\mathrm{E}\left[Y\right]\right)^2 \tag{A.16}$$

The expectation of the model's output is found to be $\mathrm{E}\left[Y\right] = \mathrm{trace}\left(Q\Sigma\right)$ [47, p. 50]. The second moment of the LQ model can be expressed as follows [47, p. 54].

$$\mathrm{E}\left[Y^2\right] = 2\mathrm{trace}\left(\left(Q\Sigma\right)^2\right) + \ell^T\Sigma\ell + \left(\mathrm{trace}\left(Q\Sigma\right)\right)^2 \tag{A.17}$$

From (A.16) and the expressions for the moments displayed above, the variance of the LQ model is

$$\mathrm{Var}\left[Y\right] = 2\mathrm{trace}\left(\left(Q\Sigma\right)^2\right) + \ell^T\Sigma\ell \tag{A.18}$$