Global Sensitivity Analysis for Covering Reservoir Geological and Flow Uncertainty

Karami Moghadam, Ali

Karami Moghadam, A. (2017). Global Sensitivity Analysis for Covering Reservoir Geological and Flow Uncertainty (Unpublished doctoral thesis). University of Calgary, Calgary, AB.
doi:10.11575/PRISM/5254
http://hdl.handle.net/1880/106256
doctoral thesis

UNIVERSITY OF CALGARY

Global Sensitivity Analysis for Covering Reservoir Geological and Flow Uncertainty

by

Ali Karami Moghadam

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN CHEMICAL AND PETROLEUM ENGINEERING

CALGARY, ALBERTA

DECEMBER, 2017

Abstract

Understanding sources of uncertainty has a major impact in the reservoir management design and significantly influences the operations decision-making. Integrating all sources of flow and geological uncertainty is very important to quantify the production uncertainty and to make optimal decisions in reservoir development. However, this task is computationally very intensive and the current methods used by the industry are not robust enough to capture the full complexity of the problem. To address this, this research focuses on identifying sources of input uncertainty that significantly influence reservoir response and decision making. Some sources of input uncertainty are significant by themselves. Others are significant through their interactions. Yet others are not significant at all. This information offers great insight as well as computational gains that reservoir engineers can exploit towards better utilizing their knowledge when making reservoir decisions for the company assets. For this purpose, this research provides promising sensitivity analysis frameworks that are suitable for tackling complex multi-dimensional models within the reservoir modelling workflow and overcome the drawbacks of the commonly used approaches.

The first section of this research study introduces a screening method that can successfully categorize the uncertain parameters in terms of their significance to the reservoir response with a low computational cost. This method is followed by a more sophisticated approach that is able to quantify the contribution of each input parameter to the variability of the model responses as well as the existing interactions among the parameters. The relation

between the accuracy of the results and the choice of experiment design is discussed in this section. In order to overcome the high cost of computation intrinsic to this method, Single-Layer and Multi-Layer Neural Network surrogate models are successfully employed and integrated with the method. In the next chapter, another approach based on classifying the response/decision variables into a limited set of discrete classes is discussed. This approach quantifies the sensitivity to parameters and parameter interactions, and incorporates the possibility that the interactions can be asymmetric for complex reservoir modeling. The discussed approaches are demonstrated and validated with multiple well known sensitivity analysis test functions and real field case studies.

Acknowledgements

This thesis would not have been possible without the inspiration and support of a number of wonderful individuals — my thanks and appreciation to all of them for being part of this journey and making this thesis possible. I would like to express my gratitude to my supervisor, Dr. Zhangxing (John) Chen for his continuous support, great leadership, and constant cooperation. I would also like to thank my co-supervisor Dr. Mario Costa Sousa for all his help and kindness. I'd also like to appreciate the contributions of the members of my examining committee. I am also thankful to Dr. Long Nghiem, Dr. Chaodong Yang and my colleagues at Computer Modelling Group Ltd. for providing an environment to gain valuable experience. Finally, my deep and sincere gratitude to my family, my partner, and my friends for their continuous and unparalleled love, help and encouragement. Without their enthusiasm, support and continuous optimism this thesis would hardly have been completed.

Dedication

To the precious love of my family

&

In beloved memory of my late grandparents

Table of Contents

List of Tables

List of Figures and Illustrations

List of Symbols, Abbreviations and Nomenclature

Symbol	Definition
Ω	Input factor space
Δ	Step size
$\hat{d}_{k,i\|j,m}$	Standardized CDF distance for interaction of parameter p_i on p_j in class k
$\hat{d}_{k,i}^{S}$	Standardized CDF distance for class k and parameter p_i
${w_{lj}}^{(K)}$	Weight matrix
$\delta_{a,b}$	Dissimilarity distance between models a and b
B'	Modified sampling matrix
B^*	Trajectory matrix
D^*	Diagonal matrix
D_{woc}	Water-oil contact depth
$\hat{J}$	Jacobian
$J_{m,k}$	Unit matrix
Kr_{gcg}	Relative permeability value of oil at the minimum gas saturation in the gas-liquid table, *fraction*
Kr_{gcl}	Relative permeability value of gas at the connate liquid saturation in the gas-liquid table, *fraction*
Kr_{ocw}	Relative permeability value of oil at the connate water saturation in the oil-water table, *fraction*
Kr_{wiro}	Relative permeability value of water at the irreducible oil saturation in the oil-water table, *fraction*
N/G	Net to Gross
P^*	Permutation matrix
P_b	Bubble point pressure
R^2	Coefficient of determination
S_{Ti}	Total sensitivity index
Sg_{con}	Connate gas saturation, *fraction*
Sg_{crit}	Critical gas saturation, *fraction*

S_i	First order sensitivity index
So_{irg}	Irreducible oil saturation in the two phase oil-gas system, *fraction*
So_{rg}	Residual oil saturation in the two phase gas liquid system, *fraction*
Sw_{con}	Connate water saturation, *fraction*
Sw_{crit}	Critical water saturation, *fraction*
$V_{i_1 i_2 \ldots i_s}$	Partial variances of the model
$X_{\sim i}$	Other input factors
X_i	Input factor
$\tilde{Y}$	Proxy output
a_i	Coefficients of Sobol G function
c_k	Class of response
d_{ij}	Euclidian distance between a pair of models x_i and x_j
d_{ml}	Distance between trajectories
n_g	Corey exponent for gas
n_{og}	Corey exponent for oil
$s_k(p_i \mid p_j)$	Conditional interaction sensitivity per class
w_i	Neuron weight
x^*	Randomly selected base value
x_p	Input parameter vector
$\delta_{\alpha,\beta}$	Kronecker symbol
I^S	S-dimensional unit cube
M	Total number of neurons
T	Time horizon
μ^*	Absolute mean elementary effect
$E(\vec{w})$	Error between the forecasted value and the exact simulation output
N	Number of sample points (training points for proxy)
R	Euclidian space

$T(z)$	Transformation Function
$V(Y)$	Total variance of the output
Y	Output
k	Number of model's input factors
p	Number of levels of the Morris design
r	Number of trajectories
$s(p_i)$	Standardized sensitivity measure for parameter p_i
$s(p_i \mid p_j)$	Average measure of sensitivity based on conditional interaction over all classes
$\theta(x)$	Heaviside step-function
μ	Mean elementary effect
σ	Standard deviation (non-linear effects and interactions)
τ	Intermediate time
$\varphi(x_i, x_p)$	Neuron (radial basis) function

Abbreviation	**Definition**
CDF	Cumulative Distribution Function
CSS	Cyclic Steam Stimulation
DGSA	Distance-based Global Sensitivity Analysis
EE	Elementary Effect
FAST	Fourier Amplitude Sensitivity Test
FF	Factors' Fixing
GOR	Gas Oil Ratio
GSA	Global Sensitivity Analysis
HL	Hidden Layer
HM	History Matching
LHC	Latin Hyper Cube
LHD	Latin Hypercube Design
MC	Monte Carlo

MDS	Multidimensional Scaling
ML NN	Multi Layered Neural Network
NN	Neural Network
NPV	Net Present Value
OAT	One At a Time
OF	Objective Function
OL	Output Layer
OLHD	Orthogonal Latin Hypercube Design
OP	Optimization
PCC	Partial Correlation Coefficients
PRCC	Partial Rank Correlation
QC	Quality Control
RBF	Radial Basis Function
SA	Sensitivity Analysis
SOR	Steam Oil Ratio
SRC	Standardized Regression Coefficients
SRRC	Standardized Rank Regression
UA	Uncertainty Assessment

Epigraph

"Sensitivity Analysis for Modelers:

Would you go to an orthopedist who did not use X-ray?"

(Fuerbringer, 1996)

Chapter One: Introduction

1.1 Reservoir Modelling

In the oil and gas industry, reservoir modeling includes creating a computer model of a petroleum reservoir that could be used for different purposes. Examples include improving estimation of reserves and making decisions regarding field development, predicting future reservoir production, placing additional wells, and evaluating different scenarios for reservoir management.

A reservoir model is the representation of the physical space of the reservoir as well as its fluids behavior and is created by an array of discrete cells, defined by a grid. Values for attributes of the reservoir, such as permeability, porosity, etc. are then associated with each cell.

Reservoir models are generally categorized as:

- Geological models that are created by geophysicists and geologists which provide a static description of the reservoir, prior to production.
- Reservoir simulation models which are created by reservoir engineers to simulate the flow of fluids within the reservoir, over its production lifetime.

It is common to construct the geological model at a fine resolution, and a coarser grid for the reservoir simulation model. Geological model is used to derive the effective values for the parameters of simulation model through what is known as upscaling process.

Through years of practice, reservoir modeling has become the go-to approach in decision making and risk assessment related to a reservoir during different stages of the reservoir's life. One of the established common workflows for analyzing a reservoir model is to conduct Sensitivity Analysis (SA), History Matching (HM), Optimization (OP), and Uncertainty Assessment (UA) studies as shown in Figure 1.1. (Mirzabozorg, 2015)

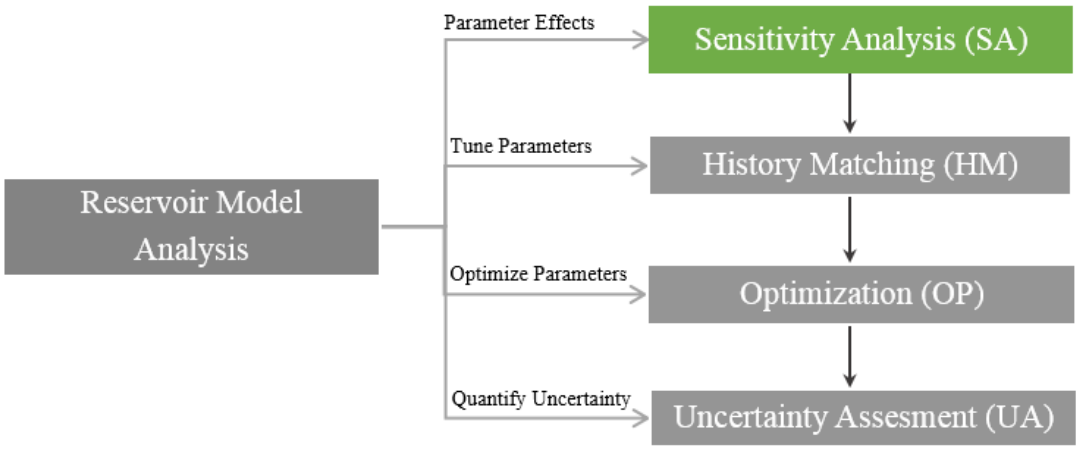


Figure 1.1: Reservoir model analysis workflow

The reservoir modeling workflow allows engineers to make informed decisions for reservoir management. However, model uncertainties pertaining to the geological and flow simulation, in addition to the inability of the currently used conventional methods to accurately quantify these uncertainties combined with time-consuming workflows, are some major bottlenecks which are impacting the quality of the decision making process for oil and gas companies.

This thesis concentrates on the first step of the reservoir modeling workflow in order to provide promising frameworks to improve the aforementioned drawbacks that exist in current approaches for sensitivity analysis.

1.2 Sensitivity Analysis

Sensitivity analysis (SA) is defined as the study of attributing the uncertainty in the output of a model to different sources of uncertainty in the model input (Saltelli et.al, 2008). Through SA, the effect of uncertainty or variability in the values of input parameters can be explored. In other words, sensitivity analysis aims to quantify the relative importance of each input parameter to the uncertainty of the outputs in a particular model. Methods of sensitivity analysis are usually classified into two categories (Sudret, 2008):

- Local sensitivity analysis methods that focus on the local impact of input parameters on the model. They are usually based on the computation of the gradient of the model response with respect to its parameters around a nominal value.

- Global sensitivity analysis methods that target to quantify the output uncertainty due to the uncertainty in the input parameters, which are taken independently or in combination with others.

1.2.1 Local sensitivity analysis

A very common approach to SA is known as the local approach. In this approach, the impact of small input perturbations, which occur around nominal values (the mean of a

random variable for instance), on the model output is studied (Sarma et al., 2015). This includes calculating or estimating the partial derivatives of the model at a particular point. Currently, such SA methods that are mostly used in the reservoir engineering workflows to understand the uncertainty of input parameters in reservoir models are unable to capture the full effects and are mostly based on simplistic assumptions. (Saltelli et al., 2008). The mostly common approaches to do parameter uncertainty evaluation are through local sensitivity analysis techniques such as tornado charts. As mentioned, these techniques provide the sensitivity of an output objective function with respect to an uncertain input parameter at a given point in the input uncertainty space; therefore, they are called "local" sensitivity. Due to this local nature, the importance ranking obtained from tornado charts can be very different depending on the point at which it is calculated, e.g. mean, P10, P90, etc. (Sarma et al., 2015). This is problematic in uncertainty management and field development planning, because different selection of this point could lead to identification of different most influential parameters regarding the output objective function and thereby to different uncertainty management/surveillance plans. Further, selecting what parameters are influential or not based on a single point (e.g. mean, median) can be inaccurate if the actual outcome is entirely different from this point (Sarma et al., 2015). Additionally, correct identification of the most significant parameters is also useful for efficient solution of inverse problems such as history matching, as it can help in reducing the dimension of the input space significantly. However, the local SA methods often fail to reliably capture all the influential inputs comprehensively due to their limitation. (Sarma et al., 2015)

1.2.2 Global sensitivity analysis

Another group of methods that have been developed in a statistical framework to overcome the limitations of local methods is Global Sensitivity analysis methods. In contrast to local sensitivity analysis, they are referred to as “Global Sensitivity analysis" (GSA) because they consider the whole uncertainty range of the model inputs (Iooss et al., 2015). Numerical model users and engineers have shown large interests in these tools which take full advantages of the advances in computing resources and numerical methods. Using GSA methods have many practical advantages for reservoir simulation including:

- identifying and ranking the most influential inputs,
- classifying non-influential inputs in order to fix them to nominal values,
- mapping the output behavior in function of the inputs by focusing on a specific domain of inputs if necessary,
- calibrating model inputs using some available information (historical model data, output behavior observations, constraints, etc.)

The GSA methods aim at determining the model input parameters which contribute the most to a quantity of interest based on model output. This quantity can be the variance of an output variable for instance (Iooss et al., 2015). These methods are mainly characterized as:

- the screening (crude sorting of the most influential inputs among a large number of parameters),

- the measures of importance (quantitative sensitivity indices) and exploration of the model behavior (measuring the effects of inputs on their all variation range).

1.3 Research Objectives

A very important contribution to the reservoir engineering workflow is providing reliable Sensitivity Analysis (SA) methods to be used while building and using numerical reservoir simulation models. They are instrumental in studying how the uncertainty in the output of a model can be assigned to different sources of uncertainty in the model input (Saltelli et al., 2008). They can be used to recognize the most contributing input parameters to the output variability as the influential inputs, or determine some interaction effects within the model. The objectives of SA are abundant; from model understanding and verification, to model simplifying and factor prioritization. SA is also an aid in the validation of a modelling code, guidance on research efforts, and justification in terms of field production design safety (Saltelli et al., 2008).

Many studies have been devoted to Sensitivity Analysis in the last twenty years as shown in Figure 1.2. A good comprehensive review of the techniques is available in Saltelli et al. (2000).

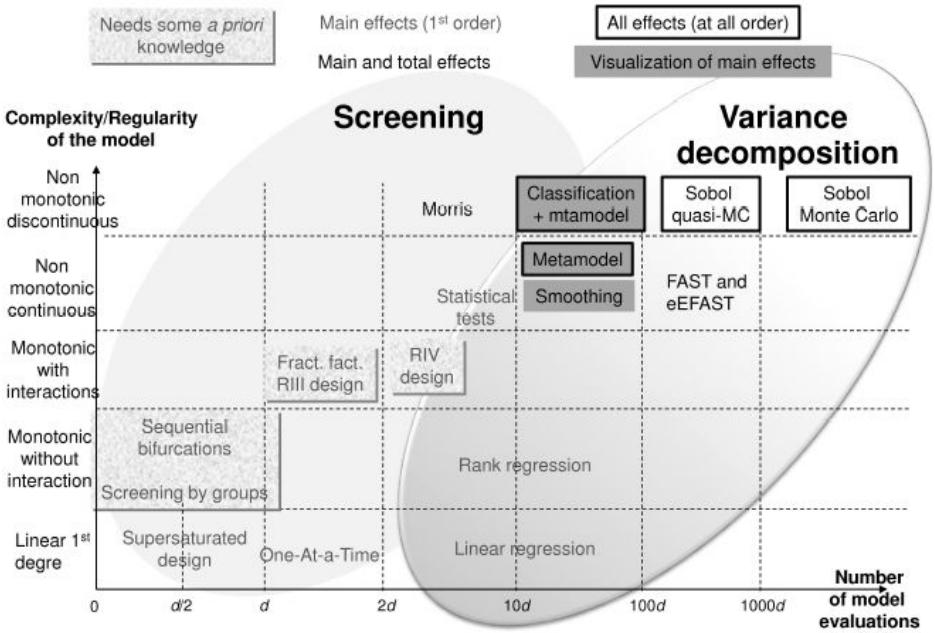


Figure 1.2: Spectrum of Sensitivity Analysis methods for different model types and levels of complexity (Iooss et al., 2015)

An important factor to note is the choice of an applicable sensitivity analysis method usually depends on factors such as the complexity of the model, the linearity/non-linearity of the response, computational cost of analysis, etc. This makes the choice of SA completely domain dependent in many cases. An example decision making process for choice of SA is demonstrated in Figure 1.3

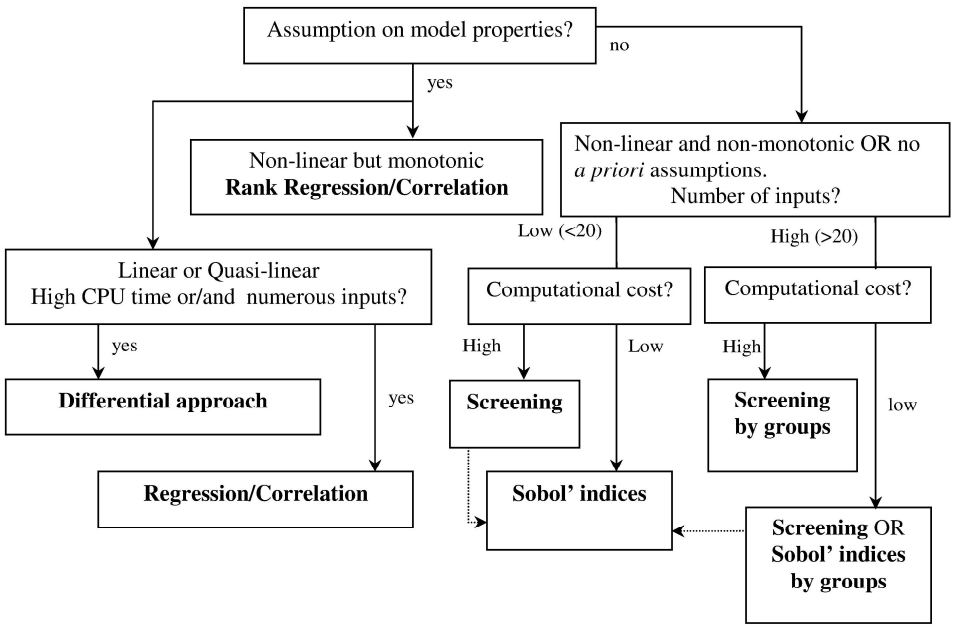


Figure 1.3: Decision making for the proper choice of Sensitivity Analysis approach (Iooss et al., 2015)

For example, while Local SA methods that we discussed in the previous section are valid choices in many scenarios, within the context of reservoir modeling, these methods are mostly too simplistic and based on assumptions like linearity and normality, local variations, etc. that do not hold true for many more complex cases. Some of the other reliable methods of performing SA are regression-based methods. The standardized regression coefficients (SRC) are based on a linear regression of the output on the input vector. The Pearson correlation coefficients measure the effect of each input variable by their correlation with the model output (Sudret, 2008). The partial correlation coefficients (PCC) are based on results of regressions of the model on all input variables except one.

These coefficients are useful to measure the effect of the input variables if the model is linear, however, as we encounter nonlinearity in many reservoir modeling cases, they fail to properly represent the sensitivities of model response (Sarma et al., 2015). Although, in case of monotonicity of the model with respect to the input parameters, the rank transform methods such as SRRC (standardized rank regression) and PRCC (partial rank correlation) coefficients might be used, but where we are dealing with general nonlinear non monotonic models as a whole, these approaches fail to provide reliable sensitivity measures (Saltelli and Sobol, 1995).

On the other hand, there are available SA methods that do not rely on requirements like linearity and normality so they could potentially be a good candidate for reservoir modeling scenarios. One category is variance-based methods: these approaches generally decompose the variance of the output as a sum of contributions of each input variable or their combinations. The McKay (1995) correlation ratios belong to this category for example. They are formulated as conditional variances and usually evaluated by crude Monte Carlo simulation or Latin Hypercube Sampling. The Fourier amplitude sensitivity test (FAST) indices (Cukier et al., 1978; Saltelli et al., 1999) and the Sobol' indices (Sobol', 1993; Saltelli and Sobol', 1995; Archer et al., 1997), are intended to represent the sensitivities for models that meet the criteria for reservoir models. But the caveat is the Sobol' indices are essentially computed using Monte Carlo simulation, which makes their direct application difficult for computationally demanding models, such as finite difference models in reservoir engineering.

As it can be observed, the choice and application of a robust SA method that could result in accurate and reliable analysis to be incorporated in the reservoir modeling workflow is no trivial task. The objective of this research is to put the emphasis on Global Sensitivity Analysis (GSA) measures that could be reliably used for different scenarios of reservoir modeling and present encouraging workflows that would help reservoir engineers benefit from these promising methods through Sensitivity Analysis studies.

1.4 Thesis Organization

In this thesis, we discuss three robust global sensitivity analysis methods aimed for handling complex reservoir models.

Chapter 2 introduces a 'Screening Method' that is based on a discretization of the model parameter inputs in levels, allowing a quick exploration of the model behavior. This method is adapted to a large number of inputs; since practice has often shown that in many cases only a small number of input parameters are primarily influential. The aim of this type of method is to identify the non-influential inputs with a small number of simulations while making realistic hypotheses on the model complexity.

Chapter 3 includes a variance based sensitivity analysis method which is proposed to get detailed analysis with regards to the sensitivities of the model outputs to input parameters, as well as capturing the interactions among parameters. It is used for complex models with non-linear and non-monotonic behavior, and is based on decomposition of the model output variance. Moreover, to overcome the big computational cost inherent to this method,

this chapter discusses the application of two Neural Network based surrogate models in the context of sensitivity analysis. Chapter 3 also includes discussion of the importance of experiment design on the accuracy of the results of the SA and implements a low-discrepancy sampling sequence for this purpose.

Chapter 4 takes a completely different approach to sensitivity analysis and incorporates the application of classification and clustering of ensemble reservoir models within the sensitivity analysis framework to quantify the influential parameters as well as asymmetric parameter interactions.

Multiple case studies and test functions are introduced and discussed in detail in all chapters and the results are evaluated to test the validity of the methods for models of different size, complexity and property.

Finally, chapter 5 provides conclusions of this study, discusses the major accomplishments and contributions and makes recommendations for future work.

Chapter Two: An Efficient Screening Design for Reservoir Models with Large Number of Parameters

Reservoir models are often very complex, computationally expensive to evaluate, and involve a large number of input factors. In many cases, it is important to understand a short-list of important factors. The question to address is: 'Which factors – among the many potentially important ones – are really important?'

Answering this question is important for a number of reasons. When a few important factors are identified, the reservoir engineer may choose to simplify the model structure by eliminating inputs that appear to be irrelevant or he/she may decide to proceed with a simpler model from the complex one. The identification of the input factors driving most of the variation in the output is also a way of quality assurance. If the model shows strong dependencies on factors that are supposed not to be greatly influential, or the other way around, one may rethink the model and eventually decide to revise. Furthermore, additional studies may be devoted to improving the estimates of the most influential factors, to increase the accuracy of model predictions.

To identify the most important factors from among a large number, the choice of a well-designed method is essential. Also, the method must be designed to be computationally cheap, i.e. requiring a relatively small number of model evaluations. Screening designs fulfil this requirement. These designs are conceived to deal with models containing tens or hundreds of input factors, efficiently. On the other hand, their main disadvantage is that

these methods only provide sensitivity measures qualitatively, i.e. they are able to sort and rank the inputs in terms of their importance, but they do not quantify how much a given factor is more important than another.

Screening methods are useful tools for Factors' Fixing (FF) in the models. In the FF setting the objective is to identify a group of input parameters that can be fixed at a nominal value over their range of uncertainty without significantly reducing the output variance. The screening methods provide a list of factors ranked in order of decreasing importance, which allows the user to identify the subset of less influential ones and possibly fix their values.

Screening techniques have been applied to several practical simulation studies in different domains, providing good results. In general, screening designs are a good choice when the number of influential factors in the model is small compared with the total number of factors. In other words, they perform better under the assumption that the influence of factors in the model is distributed based on Pareto's law and follows a few very influential factors and a majority of non-influential ones. In practice this is often verified and the results of screening exercises are generally rather satisfactory. In this chapter, we provide a reliable sensitivity analysis screening method suited for reservoir modeling applications and follow up with different test functions and case studies to validate the robustness of the method.

2.1 Morris Elementary Effects Method

The Morris method for global sensitivity analysis (also called elementary effect method) is a screening method to identify a subset of inputs that have the greatest influence on the outputs. It is an effective method of screening a few influential input factors within the many which may exist in a model (Campolongo et al., 2007). It is based on replicated and randomized "one-at-a-time" design of experiment (OAT), meaning that in each run only one input parameter is given a new value. Morris method provides global sensitivity analysis of the parameters through making local changes at different points x_i of the uncertainty range of input values and repeating the process r number of times. This is designed to calculate the changes in the output due to changes in a specific input factor in the OAT design. These differences are called "Elementary Effects" (EE). The method is global due to the fact that it varies over the whole range of uncertainty of the input parameters.

Similar to other screening approaches, the EE method provides qualitative sensitivity analysis measures. This basically means Morris method allows ranking the input factors in their order of importance and hence identifying influential and non-influential parameters, but it does not quantify the relative importance of the inputs exactly.

With its sensitivity measures, the Morris method determines if the effect of the input factor X_i on the output Y is

- negligible,
- linear and additive,

- or nonlinear\involved in interactions with other input factors $X_{\sim i}$.

2.2 Morris Sensitivity Measures

The Morris method provides two sensitivity measures per each input factor:

- the measure μ, which provides the overall importance of an input parameter on the model output;
- the measure σ, which describes interactions and\or non-linear effects.

The measures μ and σ are obtained according to the construction of a series of trajectories in the discretized input parameter uncertainty space, where inputs are randomly moved One-At-a-Time (OAT).

2.3 Experiment Design

For this purpose, the input factor space Ω is first discretized into p "levels" and the possible input factor values will be contained inside this k-dimensional p-level grid, where k is number of model's input parameters and p is the number of levels of the design (discretized level) as mentioned.

We assume that the k-dimensional vector X of the model input has components X_i each of which can assume integer values in the set $\left\{0, \frac{1}{p-1}, \frac{2}{p-1}, \dots, 1\right\}$.

The design of experiments starts by sampling a set of randomly selected start values, i.e. random seed, within the defined ranges of possible values from above and calculating the subsequent model outcome Y. This start value x^* is randomly chosen for x with each

component being sampled from the set $\left\{0, \frac{1}{p-1}, \ldots, 1-\Delta\right\}$. The next step involves changing the value for one variable by Δ while all the other inputs remain at their start values, and calculating the change in model outcome in comparison to the first steps. This means one or more of the components of x^* are increased\decreased by the step size, Δ, so that the new vector $x^{(1)}$ still has values belonging to the set of possible input values. Then, the values for another variable are changed by Δ (the previous variable is kept at its changed value and all other ones kept at their start values) and the resulting change in model outcome compared to the second run is calculated. This would continue until all input variables are modified, and one trajectory is created. An example of a trajectory for $k = 3$ is illustrated in Figure 2.1.

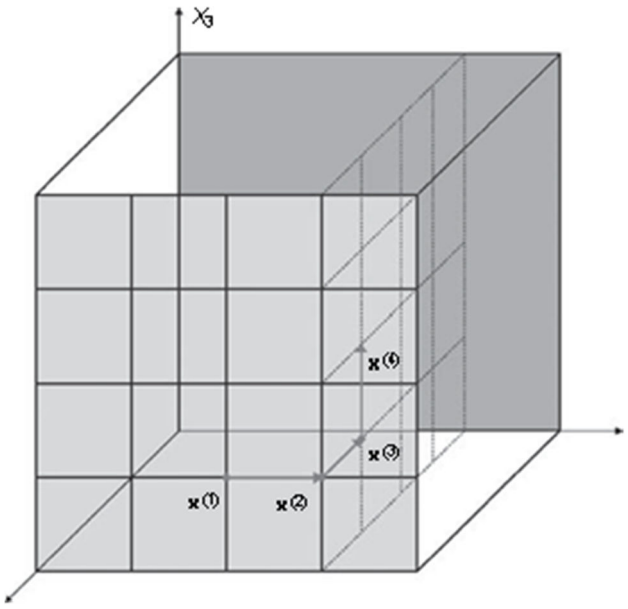


Figure 2.1: example of a trajectory in the input space when $k = 3$ (Saltelli et al., 2008)

In order to estimate the sensitivity measures, the experiment design focuses on the problem of sampling r trajectories. Since the computation of each elementary effect requires two sample points, the most basic design would need $2r$ sample points for each input, for a total of $2rk$, where k is the number of input factors (Campolongo et al., 2003). Morris (1991) suggested a design with better efficiency which is able to build *r* trajectories of $(k + 1)$ points in the input space. In this approach, each trajectory provides *k* elementary effects, one per input factor, for a total of $r(k + 1)$ sample points.

The trajectories can be generated systematically as explained in the following manner:

A trajectory can be seen in the form of a matrix, B^*, with dimension $(k + 1) \times k$. In this matrix rows are the vectors $x^{(1)}, x^{(2)}, \dots, x^{(k+1)}$. To build B^*, the first step is the selection of a matrix B, whose dimensions are the same as B^*, and with elements of 0's and 1's. The main property here is that for every column index j, $j = 1, \dots, k$, there are two rows of B that differ only in the j^{th} entry. For example, a good choice for B could be a strictly lower triangular matrix of 1's (Campolongo et al., 2007):

$$B = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ 1 & 1 & 1 & 0 & \cdots \\ \dots & \dots & \dots & \cdots & \cdots \end{bmatrix} \quad (2.1)$$

Using B', the random directions of the trajectory are determined as defined in Eq. (2.2). B' specifies whether changes in the trajectory are positive or negative.

$$B' = \frac{1}{2}\left[\left(2B - J_{m,k}\right)D^* + J_{m,k}\right] \quad (2.2)$$

where $J_{m,k}$ is a ($m \times k$) matrix of 1's, where $m = k + 1$; B is a ($m \times k$) sampling matrix, explained above; D^* is a k-dimensional diagonal matrix which the diagonal elements have an equal probability of taking a value of +1 of −1.

B^* is now given in the following equation:

$$B^* = (J_{m,1}x^* + \Delta B')P^* \quad (2.3)$$

The final permutation matrix P^* is a randomly selected k-dimensional matrix where each column and row contains only single element equal to 1 and the rest 0's. P^* is not an essential element of B^*, but the random locations of the 1's changes the order that the variables are perturbed, and increases the number of trajectories possible (King et al., 2013).

$$P^* = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (2.4)$$

The final trajectory matrix, B^*, is given in the following equation:

$$B^* = (J_{m,1}x^* + \frac{\Delta}{2}[(2B - J_{m,k})D^* + J_{m,k}])P^* \quad (2.5)$$

2.3.1 Example for sampling

Consider a $k = 3$ variable model for which Ω has $p = 4$ levels, $x = \left\{0, \frac{1}{3}, \frac{2}{3}, 1\right\}$ and $\Delta = \frac{1}{3}$ has been selected for the experiment steps. Base value $x^* = \left\{\frac{1}{3}, 0, \frac{2}{3}\right\}$ have been randomly selected. Also the D^* matrix is chosen as

$$D^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (2.6)$$

The modified sampling matrix B' is calculated and then multiplied by the $\Delta = \frac{1}{3}$ defined earlier, to create the following matrices:

$$B' = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \quad (2.7)$$

Multiplied by $\Delta \Rightarrow$

$$\Delta B' = \begin{bmatrix} 0 & 0 & \frac{1}{3} \\ \frac{1}{3} & 0 & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & 0 \end{bmatrix} \quad (2.8)$$

The final permutation matrix P^* is a randomly selected as

$$P^* = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \quad (2.9)$$

Therefore, the final sampling matrix B^*would be calculated as:

$$B^* = \begin{bmatrix} 1 & \frac{1}{3} & 0 \\ 1 & \frac{2}{3} & 0 \\ 1 & \frac{2}{3} & \frac{1}{3} \\ \frac{2}{3} & \frac{2}{3} & \frac{1}{3} \end{bmatrix} \quad (2.10)$$

Here it can be seen that each column, which represents a variable, is changed one-at-a-time by a negative or positive $\Delta = \frac{1}{3}$.

The columns of B^* can be scaled to appropriate ranges of the input parameters and the model simulations performed.

After a sampling matrix B^* is created, this procedure is repeated r times, each time with a different set of start values, which leads to a total number of r trajectories, equivalent of $r \times (k+1)$ runs. This number of runs is very efficient compared to other methods for sensitivity analysis which are more demanding.

One issue with this type of sampling is the density of trajectories in the parameter space. The sampling strategy may not provide sufficient coverage of the variable space, especially when dealing with a large number of input variables. Due to the random starting point of the trajectories, there could be an uneven density of the r number of trajectories in the parameter space resulting in over-sampled/under-sampled areas that would affect the quality of sampling.

Campolongo et al. (2007) proposed an improvement of the sampling strategy just described that facilitates a better scanning of the input domain without increasing the number of model executions needed. The main idea is to select the r trajectories in such a way that their spread in the input space is maximized. This experiment design starts by generating a high number of different trajectories, and then selects the subsection of r with the highest spread (e.g. r 10, 20 out of all generated trajectories). Here the concept of spread is based on the following definition of distance, d_{ml}, between a pair of trajectories m and l:

$$d_{ml} = \sum_{i=1}^{k+1}\sum_{j=1}^{k+1}\sqrt{\sum_{z=1}^{k}\left[X_z^{(i)}(m) - X_z^{(j)}(l)\right]^2} \text{ for } m \neq l, \qquad (2.11)$$

otherwise $d_{ml} = 0$.

Here k is the number of input factors and $X_z^{(i)}(m)$ indicates the z^{th} coordinate of the i^{th} point of the m^{th} trajectory. In other words, d_{ml} is the sum of the geometric distances between all the pairs of points of the two trajectories under analysis (Campolongo et al., 2007). The best r trajectories from the total ensemble are selected by maximizing the distance d_{ml} among them. First we consider for each possible combination of r trajectories from the ensemble the quantity D^2, which is the sum of the squared distances between all possible pairs of trajectories belonging to the combination. This sampling strategy optimizes the scanning of the input space. An example of input space sampling with r =6 for trajectories created for a model with 3 input parameters is shown in Figure 2.2:

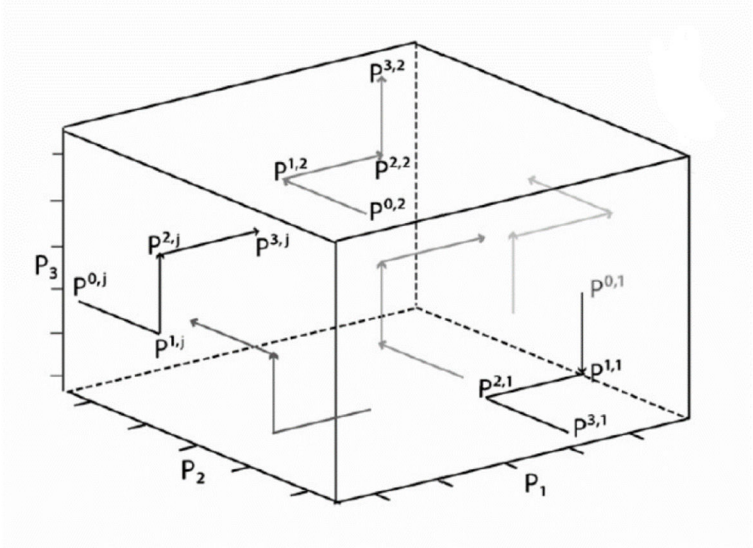


Figure 2.2: Sampling of input space with r =6 for trajectories created for a model with 3 input parameters

2.4 Elementary Effect

Each trajectory created with the sampling strategy explained in the previous section corresponds to *(k+1)* model executions and allows the computation of a derivative for each input parameter X_i.

The elementary effect (EE) of a given value of input factor X_i is defined as a finite difference derivative approximation:

$$EE_i(X) = \frac{[Y(X_1, X_2, \ldots, X_{i-1}, X_i + \Delta, X_{i+1}, \ldots, X_k) - Y(X)]}{\Delta} \qquad (2.12)$$

Based on the sampling matrix values, we can calculate one EE per input variable from each of the r trajectories from which the sensitivity indexes, or importance measures, can be computed. The steps are summarized in Figure 2.3:

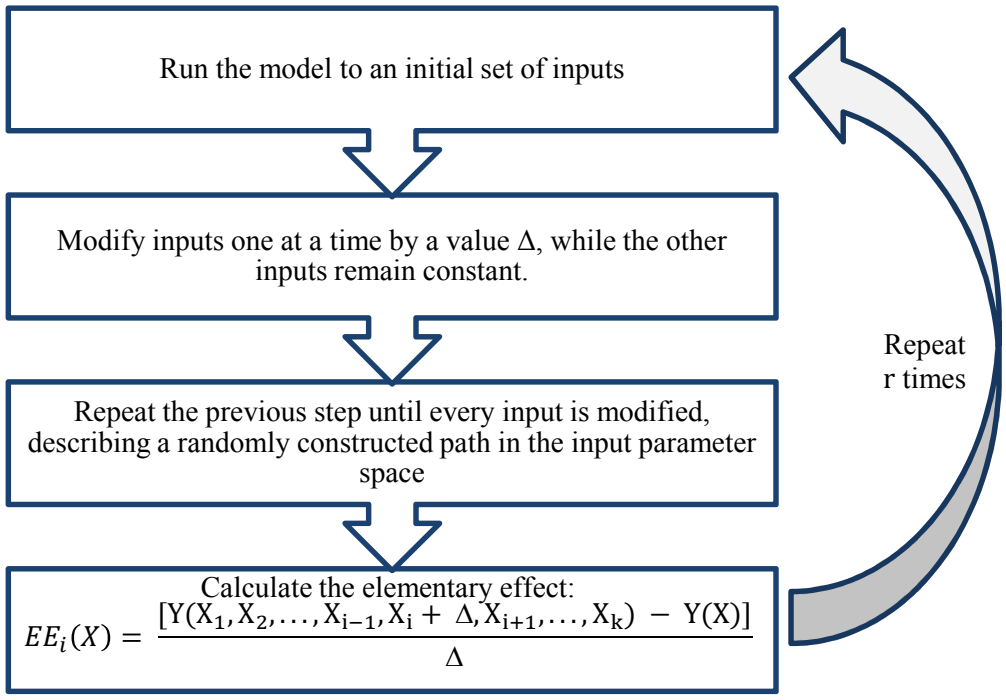


Figure 2.3: Morris screening method procedure

Morris (1991) proposed two measures, mean (μ) and standard deviation (σ) of the set of elementary effects for each input variable, while Campolongo et al. (2007) introduce a third index, μ^*, which gives extra information.

2.5 Morris Method Indices

As the set of EEs for each input variable is calculated, the two measures μ and σ are then defined as the mean and the standard deviation of the distribution of the elementary effects of each input:

$$\mu_i = \frac{1}{r}\sum_{j=1}^{r} EE(X_i^j) \qquad (2.13)$$

$$\sigma_i = \sqrt{\frac{1}{r-1}\sum_{j=1}^{r}(EE(X_i^j) - \mu_i)^2} \qquad (2.14)$$

Since with the utilized sampling strategy r number of trajectories are created, each input in the model will have a distribution of r elementary effects. Thus, the mean μ, and the standard deviation σ of these distributions could be calculated as per Eq. (2.13) and (2.14) as shown in Figure 2.4:

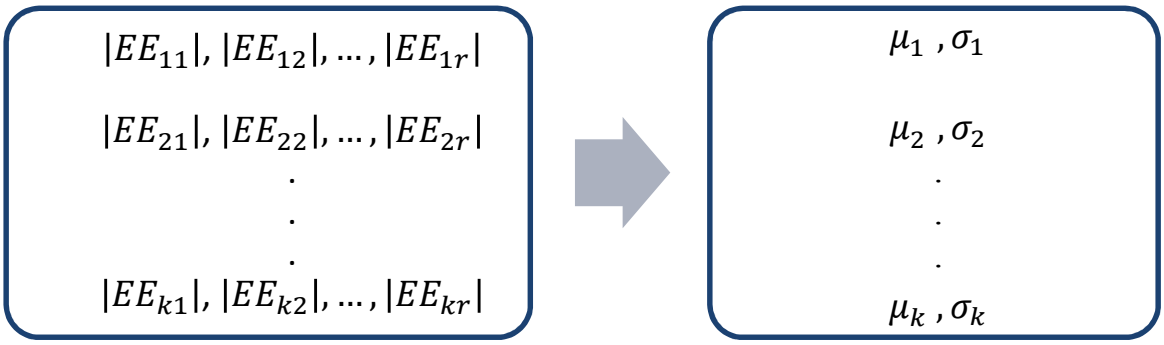


Figure 2.4: Calculation of sensitivity measures, mean μ, and the standard deviation σ

The sensitivity index μ_i, calculated using Eq. (2.13), evaluates the sensitivity of the output response to the i^{th} input variable due to all first and higher-order effects that are associated with that variable (Campolongo and Braddock, 1999). When μ_i is high for this parameter in comparison with other variables, the output has a high sensitivity to this input variable. On the other hand, a variable with a low μ_i value has small sensitivity associated to it. This can be concluded because the same Δ change for these parameters causes a relatively low change in output.

Equation (2.14) is used to determine the variance, or in other words spread of the finite distribution of the EE_i values, which is denoted by σ_i. Equation (2.14) indicates possible interactions of a parameter with other variables and/or the fact that the parameter has a non-linear effect on the output (Campolongo and Braddock, 1999). According to Campolongo et al. (2007), an intuitive meaning of σ can be explained as the following. If for parameter X_i we obtain a high value of σ, it means that the elementary effects relative

to this parameter differ significantly from one another, which implies that the elementary effect value is strongly affected by the choice of the sample point at which it is computed, i.e. by the choice of the other factors' values. By contrast, a low value of σ indicates very similar values among the elementary effects, implying that the effect of X_i is almost independent of the values taken by the other factors.

2.5.1 Avoiding Type II error

One important point about the sensitivity measure μ, is the problem of type II errors. Type II error refers to failing to identify an important parameter with significant influence on the output of the model. When the model is non-monotonic, this can happen if the finite distribution of elementary effects associated with the i^{th} input factor contains negative elements, and some effects may cancel each other out when computing the mean. Therefore, for an input parameter with elementary effects of different signs which can cancel each other out, we may possibly have a low value of μ but a considerable value of σ sometimes. This would result in underestimating the importance of factor or overestimating its involvement in interaction with other factors\the non-linearity of its effect.

In order to avoid this, Campolongo et al. (2007) proposed a revised measure μ^*. It is simply the mean of the distribution of the absolute values of the elementary effects of the input factors:

$$\mu_i^* = \frac{1}{r}\sum_{j=1}^{r}\left|EE(X_i^j)\right| \qquad (2.15)$$

The index μ^* provides an importance measure, which is free of any non-monotonic input-to-output behavior that could be present in μ. This means μ^* provides the overall sensitivity of the i^{th} input variable without any cancelling out effects that may be contained in μ_i. Similar to calculation procedure of μ, μ^* is calculated from the same distribution of r elementary effects and does not require any new model evaluations (King et al., 2013). The sensitivity measures, now including μ^*, can be seen as displayed in Figure 2.5.

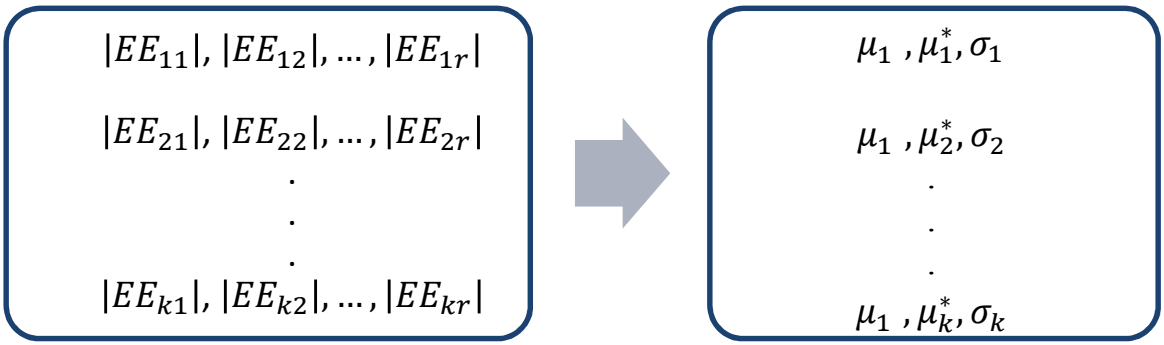


Figure 2.5: Calculating the absolute values of Elementary Effects to avoid Type II error

Using μ and μ^* together can provide insight into the nature of the non-linearity of the model and input variables. This will be explained using an example in the following section.

2.5.2 Choices of $P, \Delta,$ and r

An important choice related to Morris method is the choice of the parameters p and Δ. First factor to consider is that the choice of p is linked to the choice of r, and if we consider a high value of p, and create a high number of possible levels to be explored, we can potentially enhance the accuracy of the sampling. But we need to be aware of the fact that this must be simultaneous with a high value of r as well, otherwise many possible levels will remain not sampled.

The other important factor to consider is the choice for the parameters p and Δ so that sampling strategy guarantees equal-probability sampling from the finite distribution of elementary effects associated with the input factors. For example, one choice for the parameters p and Δ could be p even and Δ equal to $\frac{p}{2(p-1)}$.

For this choice, as shown in the top part of the Figure 2.6 below, we can see that when p = 4, the choice of $\Delta = \frac{2}{3}$ (top left) guarantees that the four levels have equal probability of being selected. On the other hand, a choice of $\Delta = \frac{1}{3}$ (top right) would imply that the levels $\frac{1}{3}$ and $\frac{2}{3}$ are sampled more often. The bottom part of the Figure 2.6 shows the case in which an odd number of levels is considered for p. In this case, regardless of the chosen value of Δ, equal probability for the elementary effects cannot be achieved. More details for the choices of p, Δ, and r can be found in Morris (1991) and Saltelli et al. (2008).

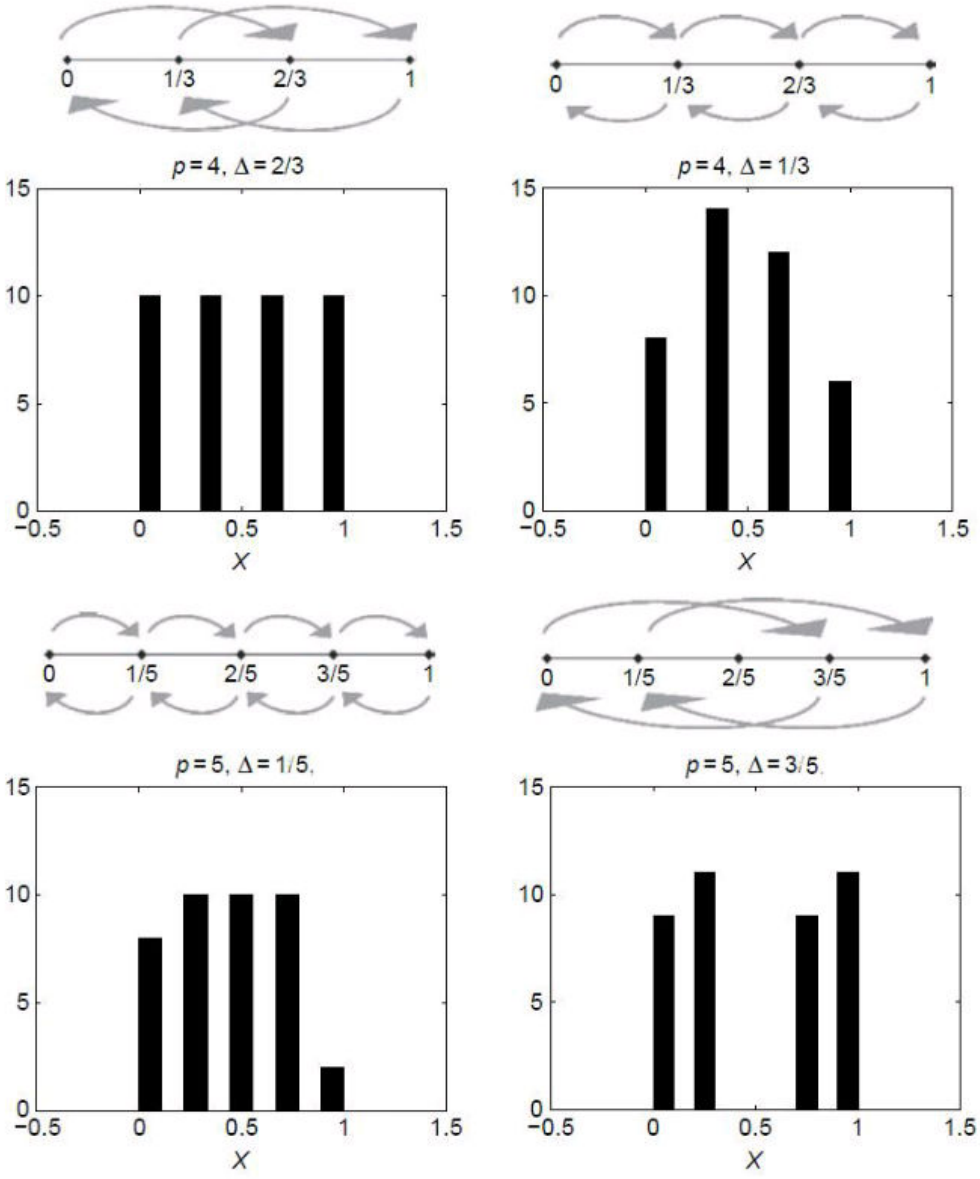


Figure 2.6: The effect of choice of p and Δ. (Saltelli et al., 2008)

2.6 Analysis of the Results

A very intuitive way of drawing conclusions from the two Morris measures, mean and standard deviation, is by analyzing them together on a two-dimensional graph so that we could rank input parameters in their order of importance for the output function and identify those inputs which do not influence the output variability.

We know that the μ^* assesses the overall influence of the factor on the output and shows the parameter importance. The standard deviation σ estimates the ensemble of the factor's effects, whether nonlinear and/or due to interactions with other factors. Based on these:

- If all samples of the elementary effect of the i^{th} input factor are zero, then X_i doesn't have any effect on the output Y, the sample mean and standard deviation will both be zero.

- If all elementary effects have the same value, then Y is a linear function of X_i. The standard deviation of the elementary effects will then of course be zero.

- For more complex situations, because of interactions between parameters as well as nonlinearity effects, if the mean of the elementary effects is relatively large and the standard deviation is relatively small, the effect of X_i on Y is "mildly nonlinear". If the opposite, the mean is relatively small and the standard deviation is relatively large, then the effect is supposed to be "strongly nonlinear" (Morris, 1991).

Looking at the parameters being distributed in a 2D space according to the 3 conclusions above, would result in a graph similar to what is shown in Figure 2.7:

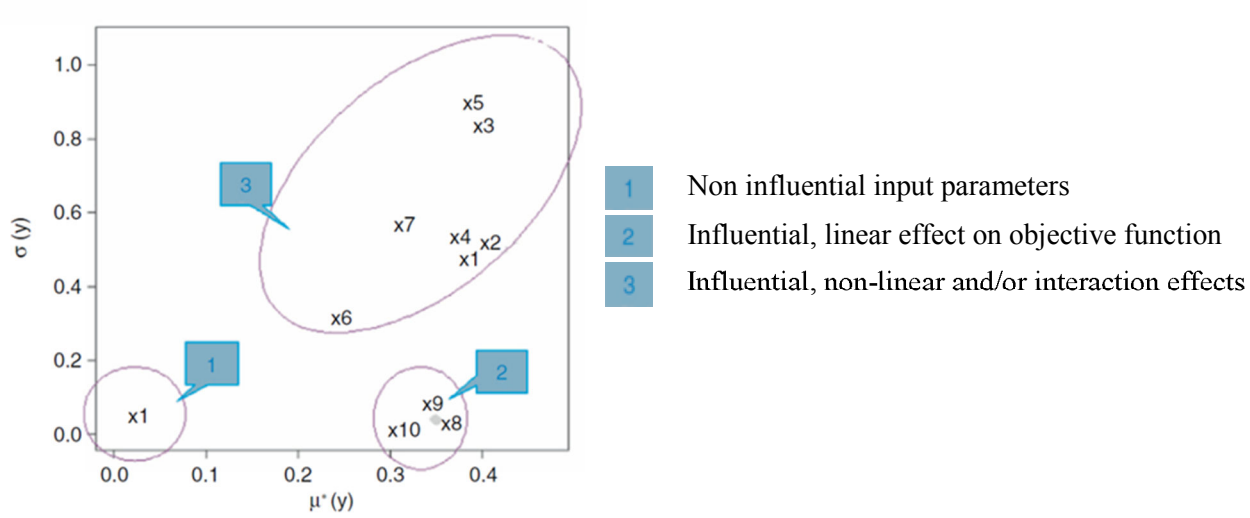


Figure 2.7: Interpretation of Morris analysis results

As a rule of thumb:

a) high mean indicates a factor with an important overall influence on the output, and

b) high standard deviation indicates that either the factor is interacting with other factors or the factor has strong nonlinear effects on the output.

2.7 Case Studies

2.7.1 Polynomial function

To demonstrate the mentioned methodology, we define a simple polynomial function with 3 parameters, with X_2 being the most important input parameter, and interactions existing between parameters X_1 & X_2:

$$Y = X_1 + 10X_2 + X_3 + 50X_1X_2 \quad (2.16)$$

Consider this model for which Ω has $p = 4$ levels, $x = \left\{0, \frac{1}{3}, \frac{2}{3}, 1\right\}$ and $\Delta = \frac{2}{3}$ has been selected for the experiment steps. Base value $x^* = \left\{\frac{2}{3}, \frac{2}{3}, 0\right\}$ have been randomly selected. Also the D^* matrix is chosen as

$$D^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.17)$$

Using a lower triangular matrix B and also matrix J,

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (2.18)$$

$$J = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (2.19)$$

The modified sampling matrix B' is calculated and then multiplied by the $\Delta = \frac{2}{3}$ defined earlier. Next, the final sampling matrix B^* would be calculated as:

$$B^* = \begin{bmatrix} \frac{1}{3} & 1 & 0 \\ 1 & 1 & 0 \\ 1 & \frac{1}{3} & 0 \\ 1 & \frac{1}{3} & \frac{2}{3} \end{bmatrix} \quad (2.20)$$

Here it can be seen that each column, which represents a variable, is changed one-at-a-time by a negative or positive $\Delta = \frac{2}{3}$.

All 3 parameters are uniformly distributed between $[0,1]$ in this case, but the columns of B^* can be scaled to appropriate ranges of the input variables and the model simulations performed.

We repeat this process and therefore create multiple B^* matrices from which we calculate the Elementary Effects (EE) of each parameter from each two consecutive rows. Therefore we can calculate the sensitivity measures μ^* and σ.

The results of Morris analysis are shown in the following Figure 2.8 and Table 2.1:

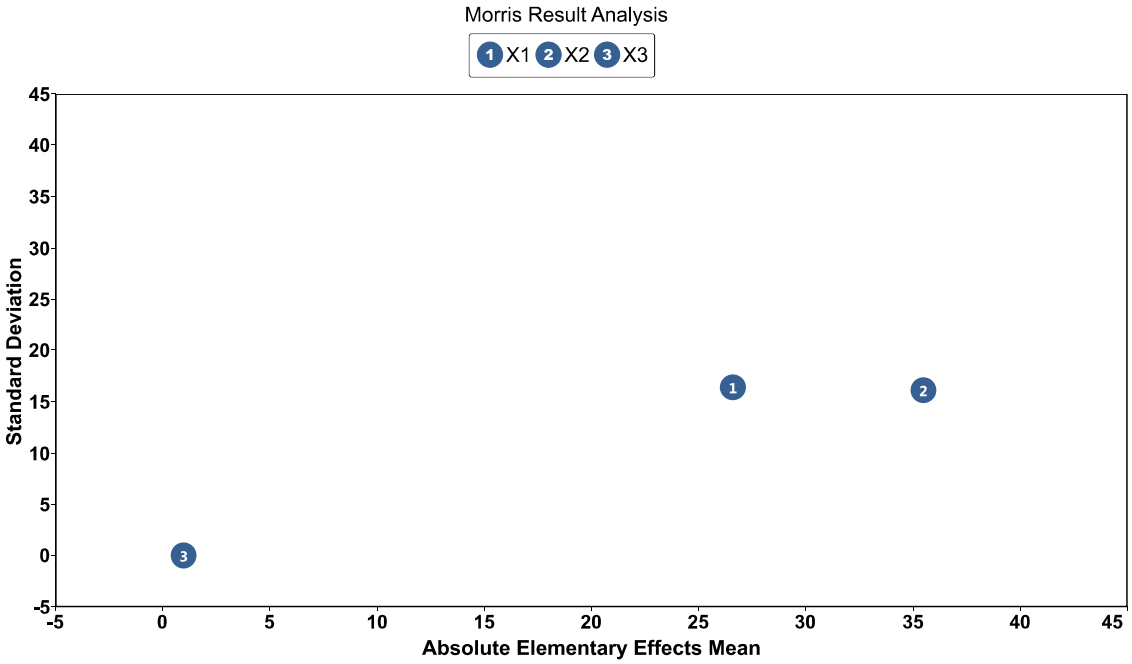


Figure 2.8: Morris chart for the polynomial objective function

Table 2.1: Absolute elementary effects mean and standard deviation values for polynomial objective function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
X_2	35.5	16.151202
X_1	26.625	16.237238
X_3	1	3.12E-15

Morris chart of Figure 2.8 shows parameter X_2 has the highest value of μ^* therefore being the most important, followed by parameters X_1 and X_3. In terms of the interactions, parameter X_3 has the value of zero on the Y axis which indicates no interaction effects for

this parameter, while parameters X_1 and X_2 are interacting similarly. This validates the result from the analysis, matching the polynomial model behavior that we can observe.

2.7.2 Ishigami function

The Ishigami function was first introduced by Ishigami and Homma (1990) and is given as follows:

$$f(x) = Sin(X_1) + a\, Sin^2(X_2) + bX_3^4\, Sin(X_1) \qquad (2.21)$$

It is commonly used to test sensitivity and uncertainty analysis techniques because it exhibits strong non-linearity and non-monotonicity. It also has a peculiar dependence on X_3, as described by Sobol & Levitan (1999). The values of a and b used by Crestaux et al. (2007) and Marrel et al. (2009) are: $a = 7$ and $b = 0.1$. We use the same values in this case study.

First, we perform a Monte Carlo simulation with 3200 experiments, compute the output of the Ishigami function, and plot the histogram of the Ishigami function output in Figure 2.9. This plot shows the distribution of values for Ishigami function with all 3 uncertain parameters sampled from the prior probability density functions.

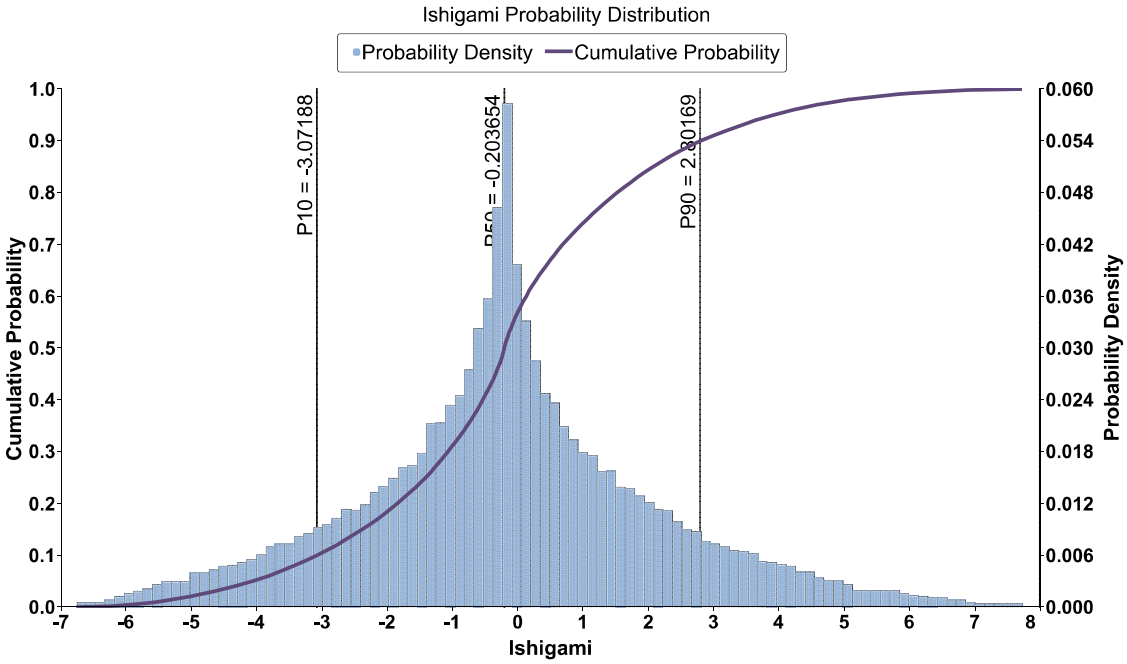


Figure 2.9: Ishigami function probability distribution and cumulative probability

Figure 2.9 shows a histogram of Ishigami function values, to illustrate the shape of the probability density function, as well as the cumulative probability. P10, P50, and P90 values are also highlighted. The result of Morris analysis is shown in the following Figure 2.10 accompanied by Table 2.2:

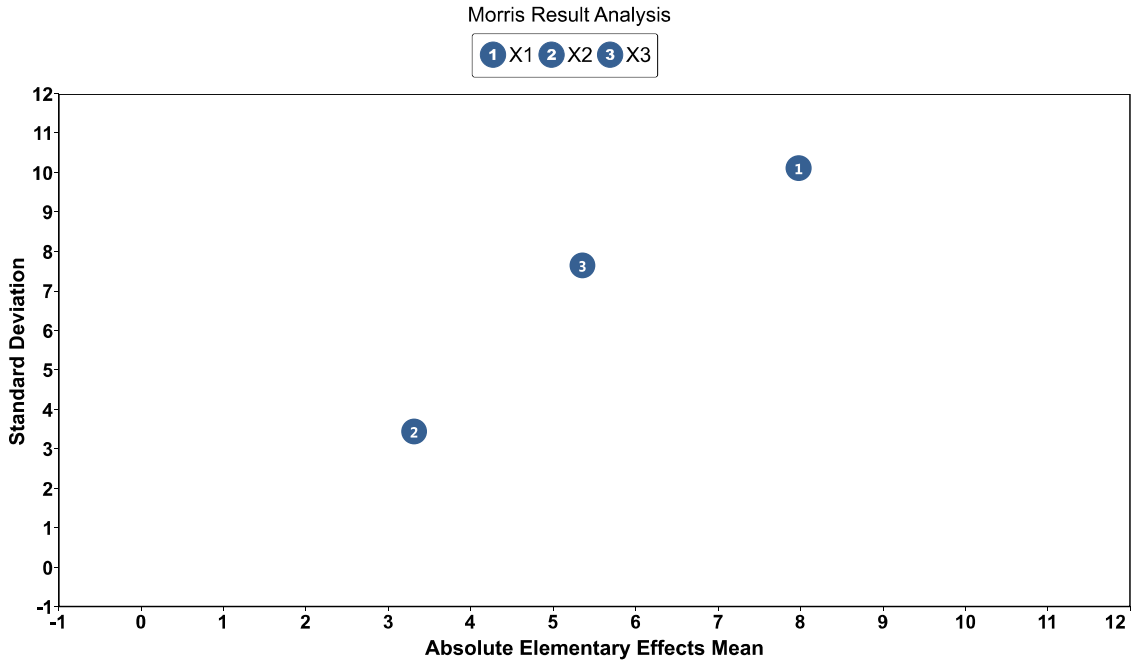


Figure 2.10: Morris analysis results for Ishigami function

Table 2.2: Absolute elementary effects mean and standard deviation values for Ishigami function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
X_1	7.9781829	10.12071
X_3	5.36013	7.666749
X_2	3.3134273	3.442772

The results demonstrate that the Morris method is successful in identifying parameter X_1 as the most influential input while capturing its interaction. It is also able to capture the fact that parameter X_3 is involved in interaction in the model but it overestimates the effects

of X_3 and completely fails to show the effect of parameter X_2. This proves that while Morris is a reliable method to do a sensitivity analysis screening of parameters involved in a model, it is best suited for models with large number of parameters where the cost of computation might be to prohibitive to run other more sophisticated sensitivity analysis methods. In cases of highly non-linear models such as Ishigami function it may fail to correctly display the sensitivity analysis results. We would discuss this more in the next chapter where we will introduce another method that does not have this deficiency. Also. case studies with the application of Morris method in real reservoir modeling scenarios are presented at the later sections.

2.7.3 Sobol G function

Sobol G function is used as an integrand for various numerical estimation methods, including sensitivity analysis methods, because it is fairly complex, and its sensitivity indexes can be expressed analytically. The exact value of the integral with this function as an integrand is 1.

$$f(X) = \prod_{i=1}^{d} \frac{|4X_i - 2| + a_i}{1 + a_i} \qquad (2.22)$$

For each index i, a lower value of a_i indicates a higher importance of the input variable X_i. Below are the values of a_i recommended by Crestaux et al. (2007). The same a_i coefficients displayed in the Table 2.3 are used in this case study as well.

Table 2.3: a_i values for Sobol G function

a_i	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}
Value	0.001	89.9	5.54	42.10	0.78	1.26	0.04	0.79	74.51	4.32	82.51	41.62

In many cases, for instance in Marrel et al. (2008), there is the constraint that $a_i \geq 0$. They conclude that:

- for a_i close to 0, X_i is very important.
- for a_i close to 1, X_i is relatively important.
- for a_i close to 9, X_i is non-important.
- for a_i close to 99, X_i is non-significant.

Based on the values of a_i used in this case study, we expect a ranking of $X_1 > X_7 > X_5 > X_8 > \ldots$ and so on.

First, we perform the Morris sampling in the 12-dimension parameter space hypercube. Based on the created sampling sequence for function parameters, the outputs of Sobol G function with respect to each input parameter can be observed in cross plots as shown in Figures 2.11 to 2.13. Through these cross plots, we can identify trends and relationships between our parameters, and objective function. As an example, in Figure 2.11, right away we can observe that compared to the other parameters, there is stronger correlation between parameter X_1 and the output. Same observation can be made for parameters X_7 and X_5 in Figure 2.12.

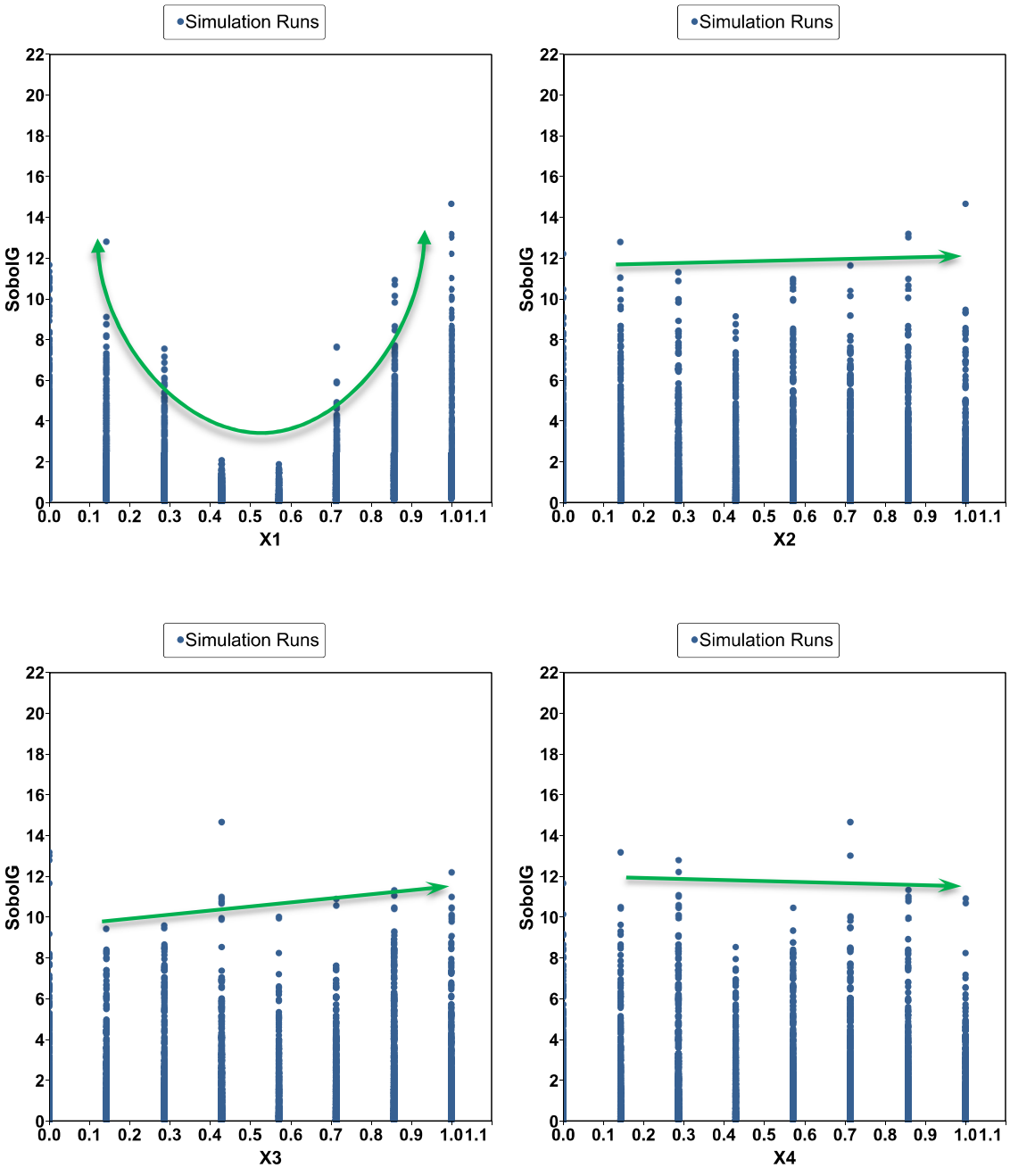


Figure 2.11: cross plots of Sobol G function vs. input parameters X_1 to X_4

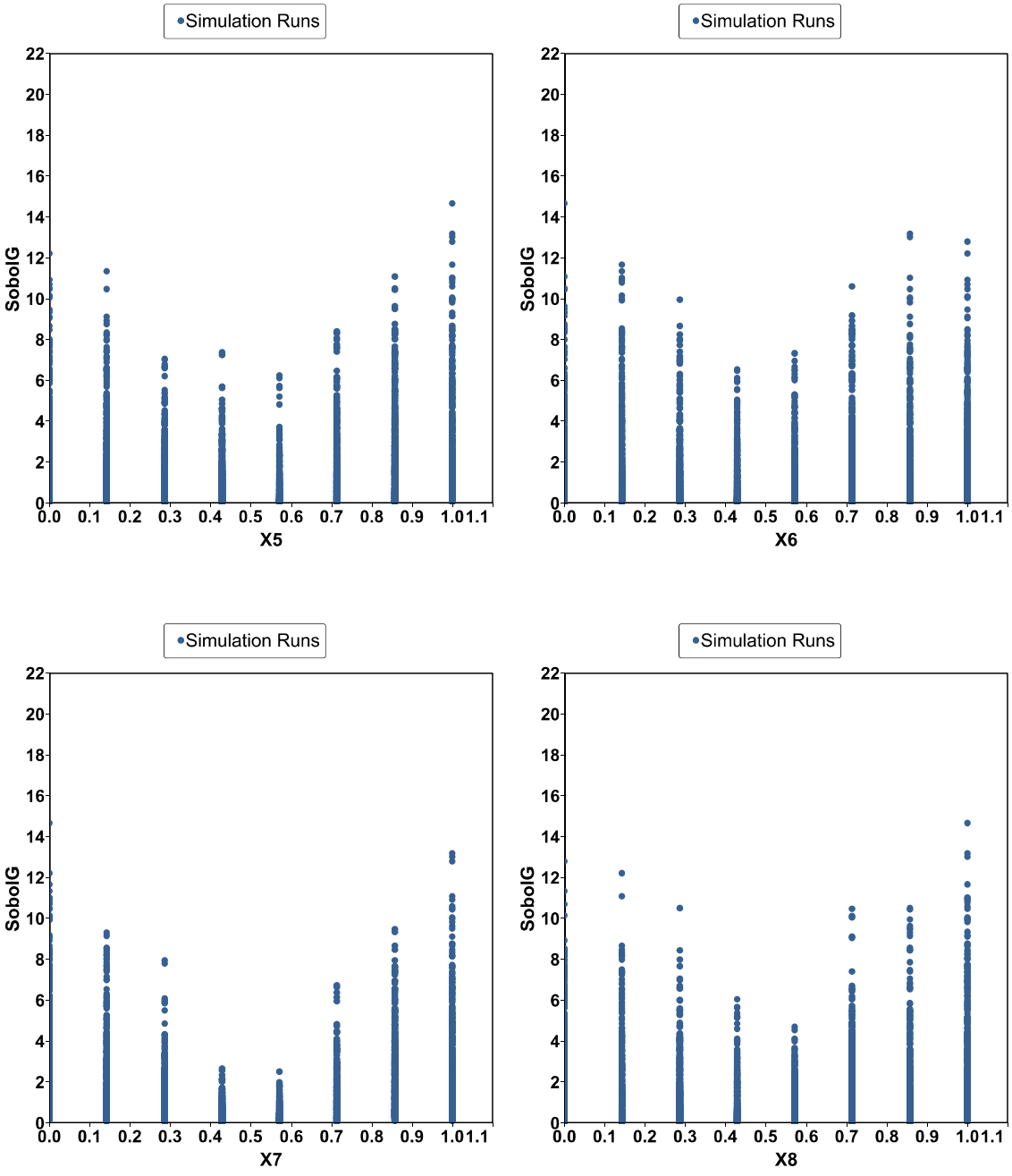


Figure 2.12: cross plots of Sobol G function vs. input parameters X_5 to X_8

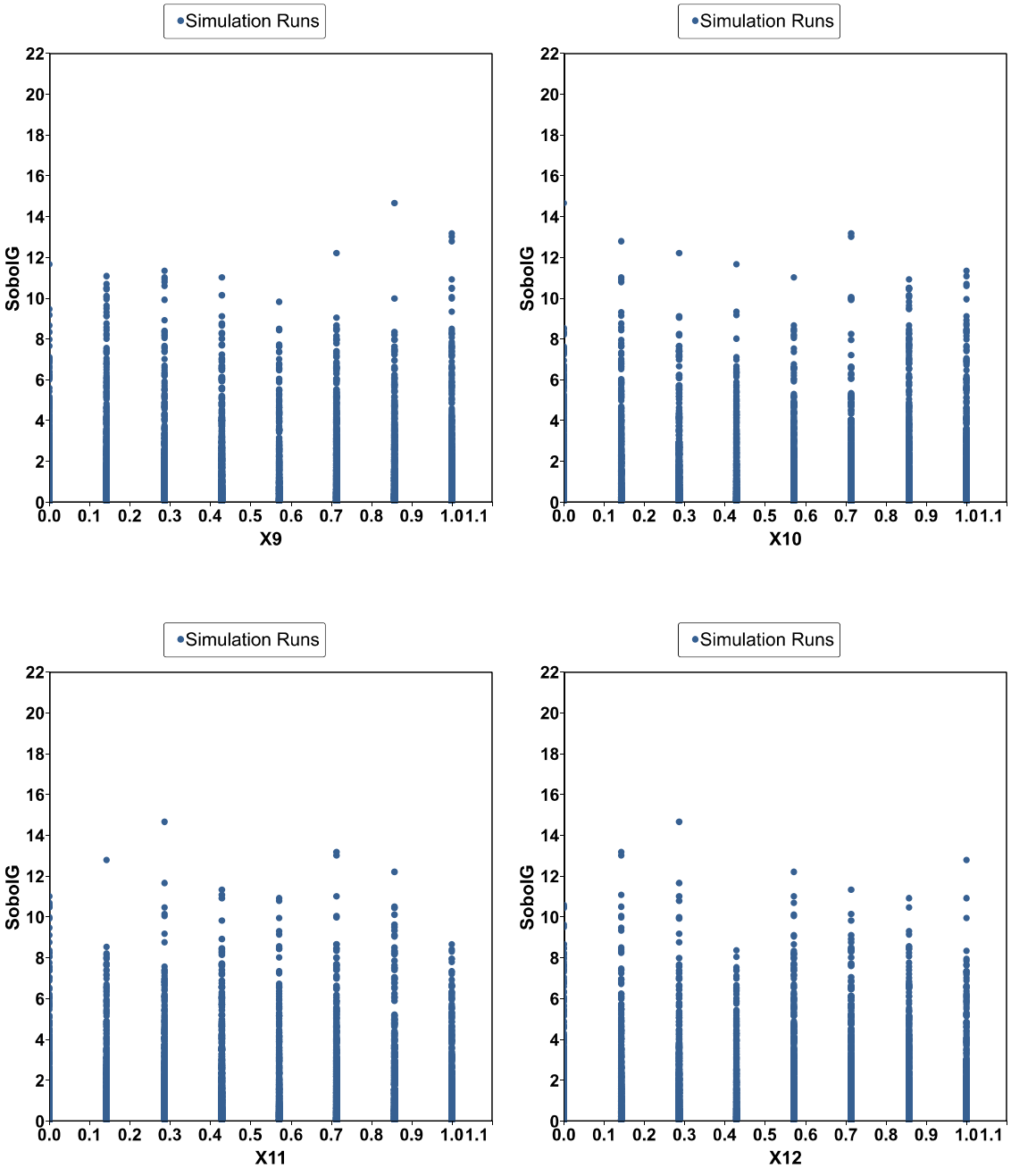


Figure 2.13: cross plots of Sobol G function vs. input parameters X_9 to X_{12}

Using Monte Carlo simulation, Figure 2.14 shows a histogram of our output function, to illustrate the shape of the probability density function, as well as the cumulative probability. P10, P50, and P90 values are also highlighted again.

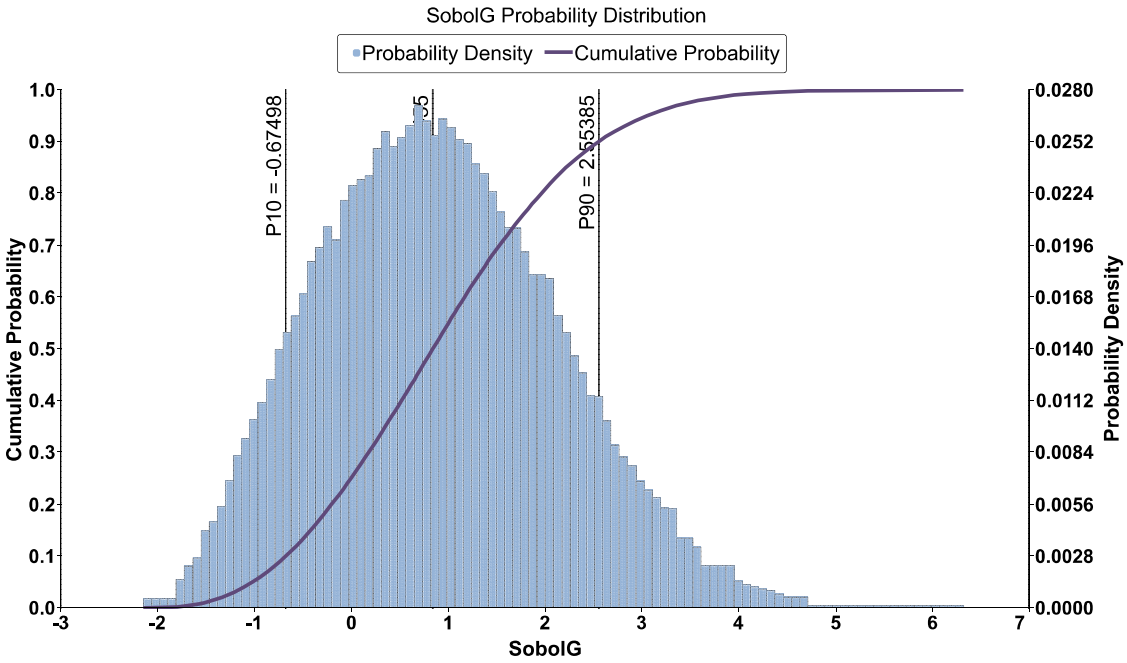


Figure 2.14: Ishigami function probability distribution and cumulative probability

Now calculating the absolute mean Elementary Effects and the standard deviations, we can plot the Morris chart in Figure 2.15. The values are tabulated in Table 2.4 as well.

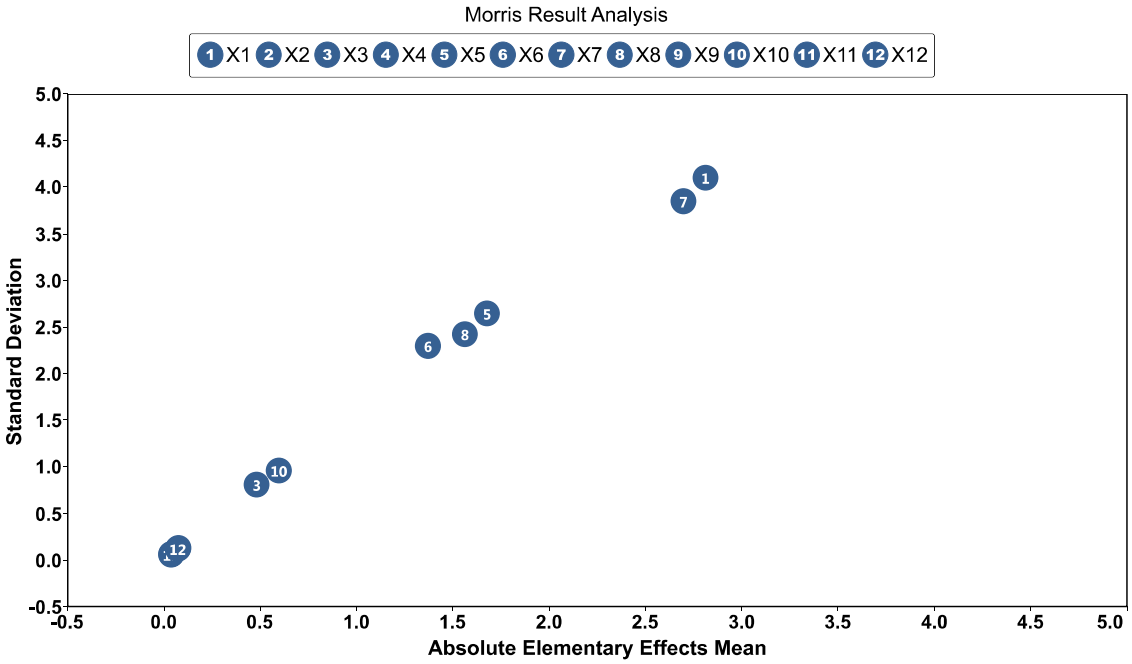


Figure 2.15: Morris analysis results for Sobol G function

Table 2.4: Absolute elementary effects mean and standard deviation values for Sobol G function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
X_1	2.8126255	4.1092466
X_7	2.6972195	3.8499568
X_5	1.6782282	2.6501719
X_8	1.5634205	2.4261587
X_6	1.3710143	2.3032483
X_{10}	0.59816215	0.96099294
X_3	0.48612539	0.81179284

X_4	0.077745173	0.13353746
X_{12}	0.074137749	0.12386474
X_9	0.042370137	0.071360284
X_2	0.038492282	0.061032131
X_{11}	0.038447828	0.068326209

Morris method can clearly categorize the parameters of Sobol G function relative to their importance. Parameters X_1 and X_7 are the most influential parameters in the model, followed by X_5, X_8 and X_6. X_{10} and X_3 are somewhat important and the rest of the parameters are not affecting the variability of the output and could be fixed to an average value within their distribution. The results directly correlate to the coefficient values used in this model which proves the accuracy of the method.

2.7.4 CSS pilot

Cyclic Steam Stimulation (CSS) is a petroleum extraction technique that uses hot steam injection into heavy-oil reservoirs or oil-sands deposits in order for extraction of oil through a common wellbore in cycles.

The production cycle of a CSS technique consists of the following stages:

1. Injection stage for heating and mobilizing the oil in the deposit for several days or weeks.
2. Soaking stage, allowing the separation of oil from other ingredients, and letting reservoir pressure to build.
3. Extraction by natural flow and also by forced pumping.

The three stages together comprise one cycle. Once the last stage is completed and oil production rates fall below a critical threshold due to the cooling of the reservoir, the production cycle is repeated a number of times until production becomes uneconomical. The CSS is also sometimes referred to as huff and puff technique.

The study of the CSS technique is important mainly because:

- It is widely used for petroleum extraction from oil sands and reservoirs with very heavy crudes.
- There is flexibility of using the same wells for steam flooding technique, at a later stage.

In this case study a single well cyclic steam pilot is studied which has been completed for three cycles. We determine the importance of reservoir and operating variables, for later history matching the model so that it can be used to develop an optimum ongoing strategy for future wells. The model has four geological layers, the properties of which are not well known. There is uncertainty about relative permeability, especially as it may affect injectivity vs. productivity of the well. Also, there is uncertainty about formation dilation and its possible effect on permeability. We anticipate the model could be sensitive to the following parameters;

- Porosity (single value)
- Permeability of each layer
- Kv/Kh ratio
- The Carmen-Kozeny exponential factor for permeability as function of porosity.
- Dilation onset pressure
- Dilation compressibility

Morris Method is used to determine how sensitive an objective function is to the model's existing parameters and their ranges in values. In particular, we want to see how much each of our history matching parameters impacts the production and injection of the well, as well as the linearity and non-linearity of their effect on the objective functions. This will be useful for determining which parameters we should modify when we start history matching. The most sensitive parameters will have a large impact on the results so these are the parameters that we should modify during the history matching process. Less

sensitive parameters can be held at their original value then modified at the end of the matching process to fine tune the results. Table 2.5 below lists the uncertain parameters of our case study.

Table 2.5: List of input parameters for the CSS pilot study

Input Parameters	Unit	Lower Limit	Upper Limit	Type
Dilation Compressibility	1/psi	0.002	0.007	Continuous
Dilation Onset Pressure	psi	200	500	Continuous
KvKhRatio	-	0.15	0.5	Continuous
Permeability Exponent	-	0	4	Continuous
Permeability Layer 1	mD	1000	4000	Continuous
Permeability Layer 2	mD	600	1000	Continuous
Permeability Layer 3	mD	250	450	Continuous
Permeability Layer 4	mD	1500	3000	Continuous
Porosity	-	0.25	0.35	Continuous

In addition, Table 2.6 includes the objective functions we would be studying in this example.

Table 2.6: List of objective functions for the CSS pilot study

Objective Functions	Unit
Cumulative Oil Production	bbl
Cumulative Water Production	bbl
Cumulative Steam Injection	bbl

We perform the sampling strategy and continue to calculate the Morris sensitivity indexes for our objective functions. The next objective function histogram plots, Figures 2.16 to 2.18, show the distribution of objective function values calculated from the sampling experiments, as well as the cumulative probability.

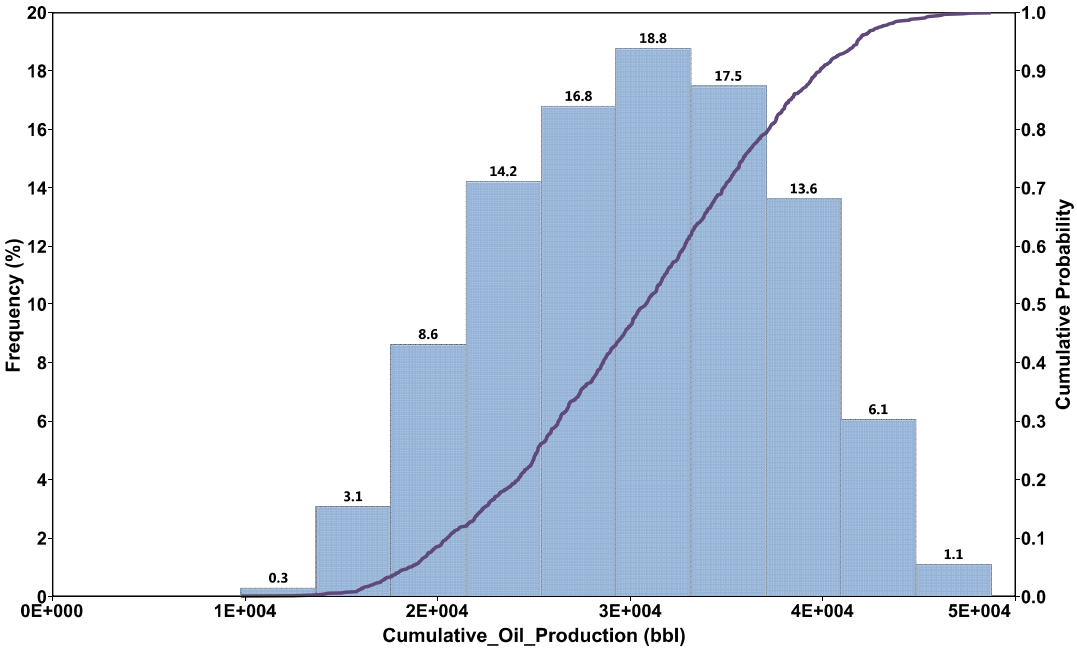


Figure 2.16: Distribution of Cumulative Oil Production objective function

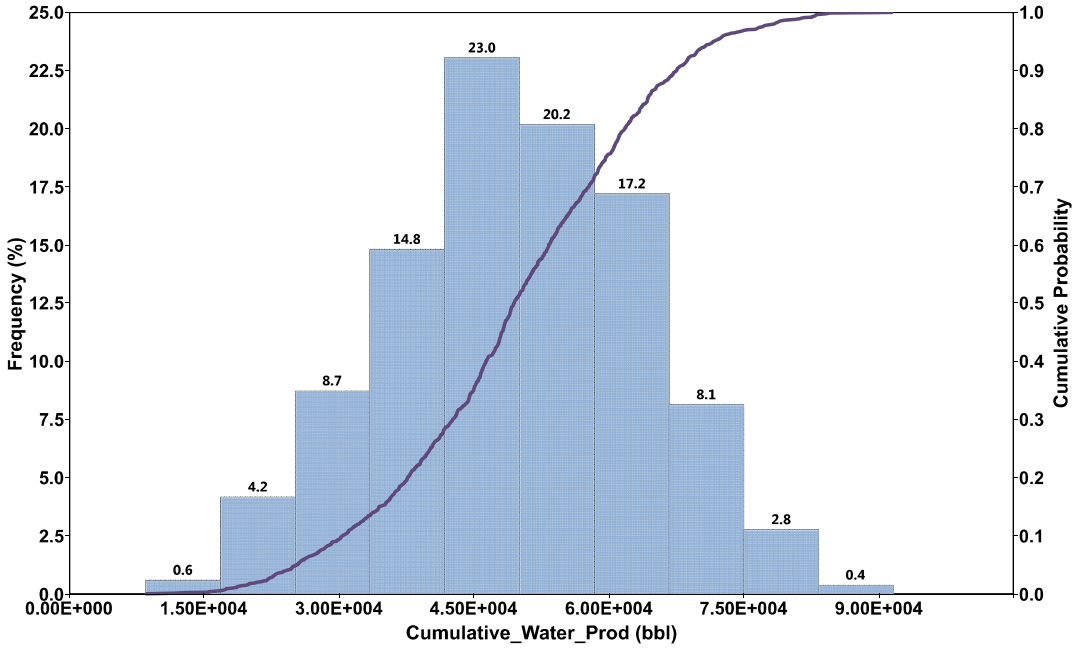


Figure 2.17: Distribution of Cumulative Water Production objective function

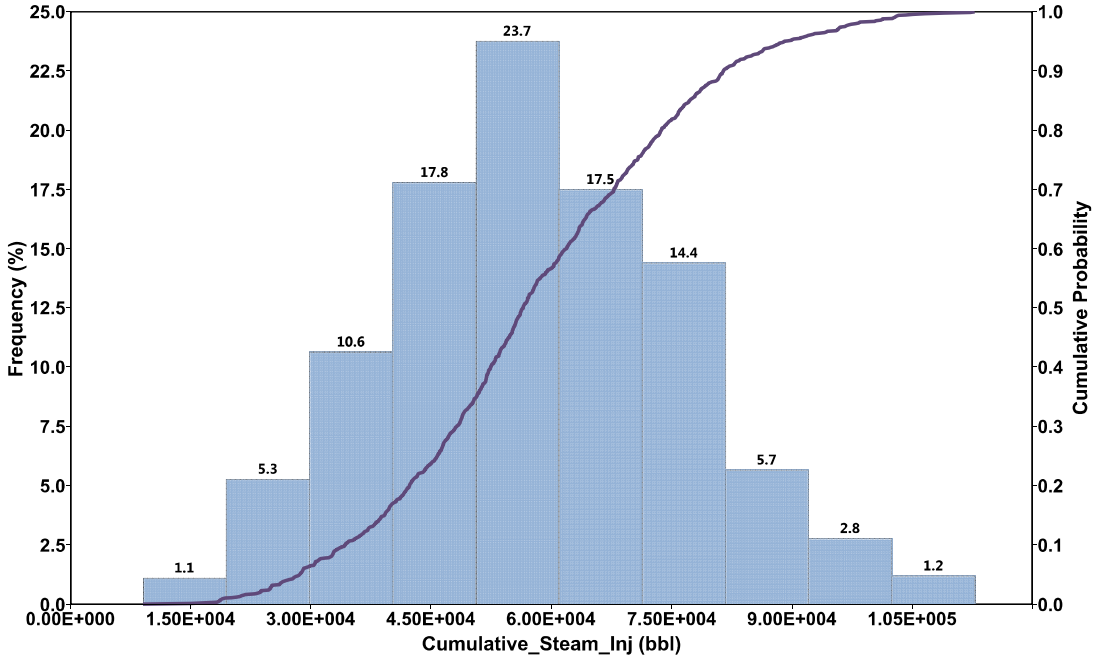


Figure 2.18: Distribution of Cumulative Steam Injection objective function

*2.7.4.1 Analyzing μ vs μ**

According to Morris sensitivity analysis procedure in this chapter, the calculation of the absolute mean elementary effects (μ^*) and mean elementary effects (μ) indicate that Permeability in Layer 1 is the most significant parameter contributing to the changes of the case study objective functions.

On top of that, the comparison between μ^* and μ provides valuable information on the signs of the effects that an input factor has on the output, too. If μ^* and μ are both high, it implies not only that the factor has a large effect on the output, but also that the sign of this effect is always the same. If, by contrast, μ is low while μ^* is high, it means that the factor examined has effects of different signs depending on the point of the space at which the effect is computed.

This can be observed about the sign of effect for Permeability in Layer 1 that it is the same, whereas the opposite is true for Permeability in Layer 4 and KvKhRatio.

The comparison charts for the 3 objective functions are included in Figures 2.19, 2.20 and 2.21 as follows.

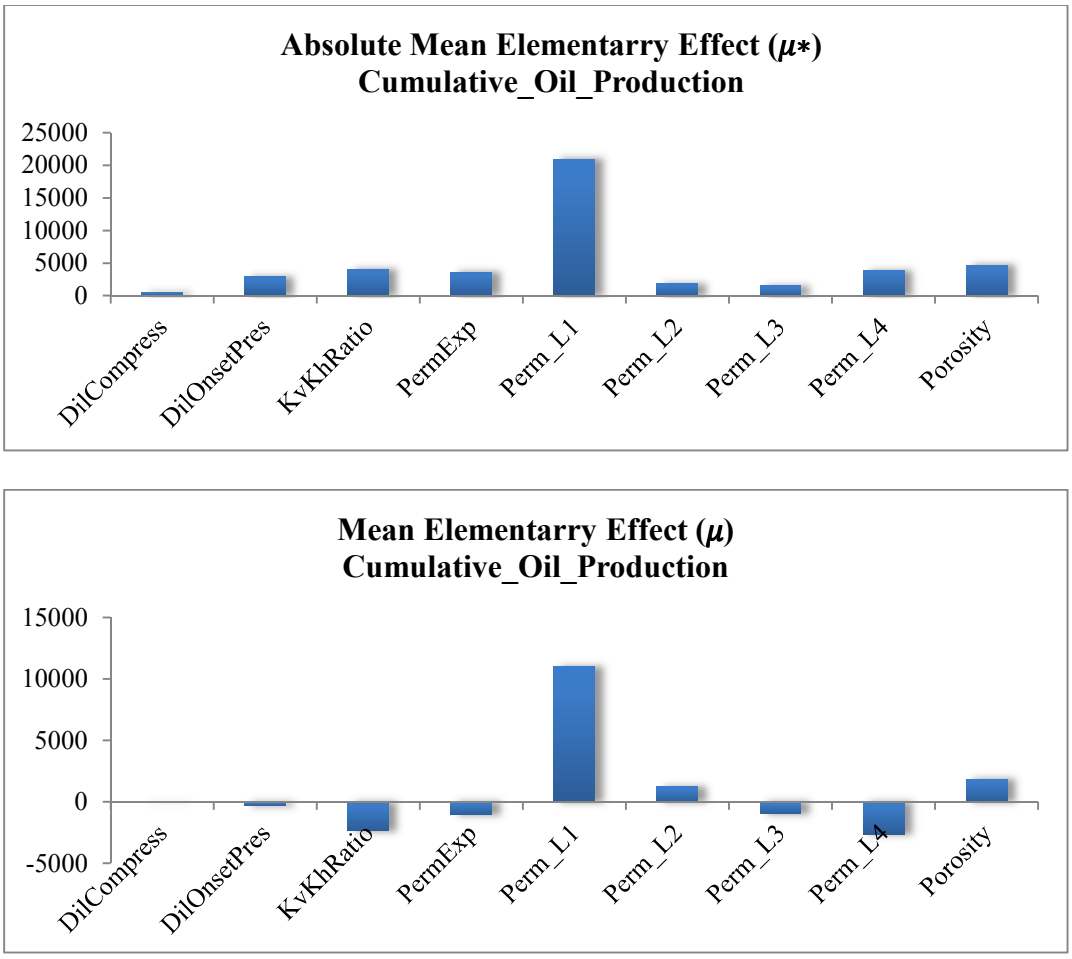


Figure 2.19: μ^* and μ charts for Cumulative Oil Production Objective Function

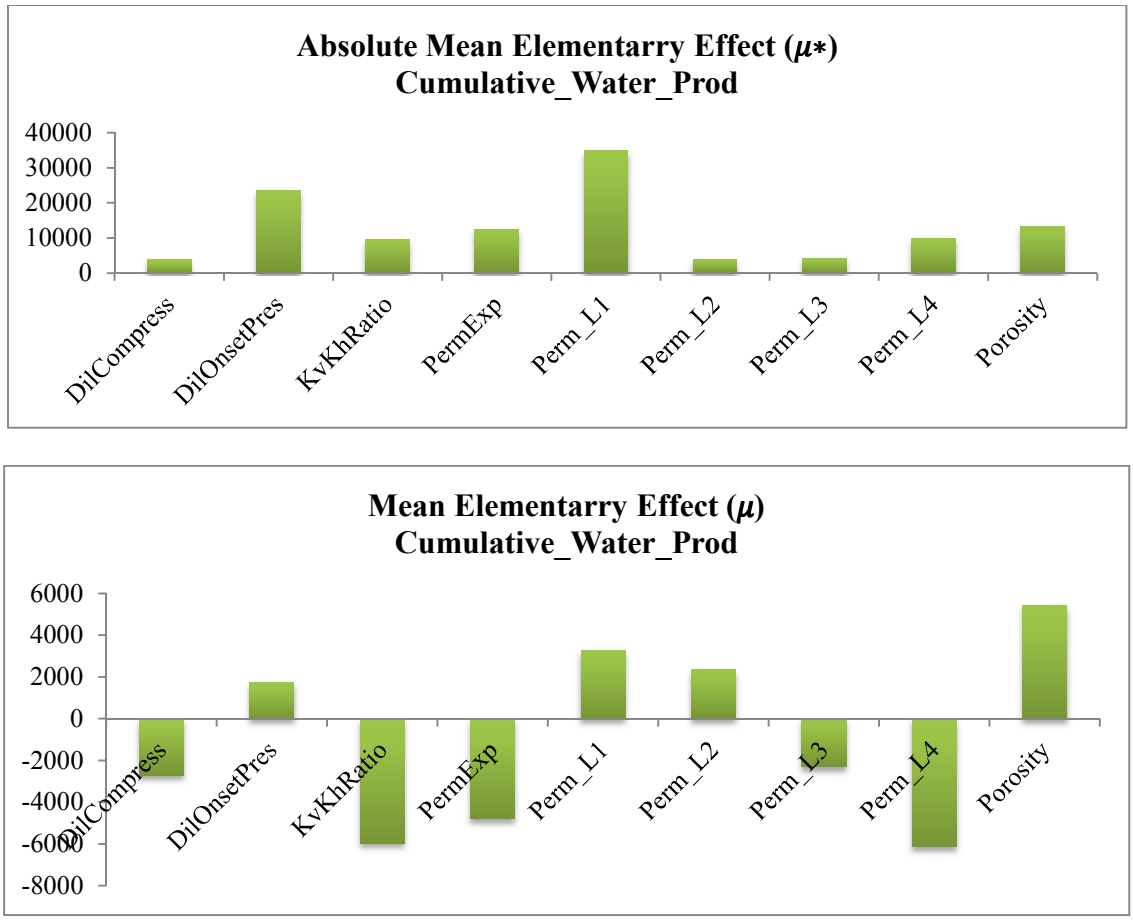


Figure 2.20: μ^* and μ charts for Cumulative Water Production Objective Function

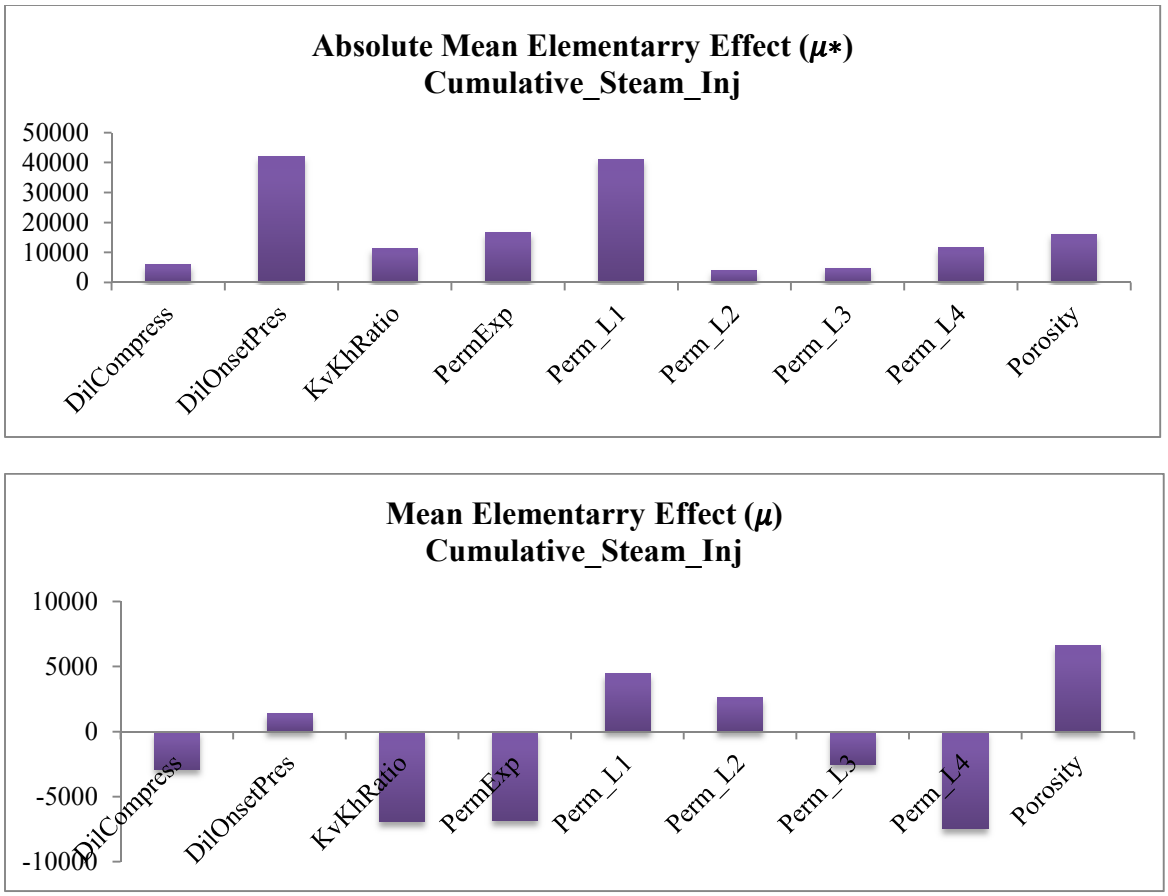


Figure 2.21: μ^* and μ charts for Cumulative Steam Injection Objective Function

To analyze the interaction existing in the model within the parameters for the objective function, we can plot μ^* vs σ charts and perform the analysis on the screening results in one simple and easy to understand graph.

The results are plotted in the respective Figures 2.22 to 2.24 as well as their values shown in Tables 2.7 to 2.9.

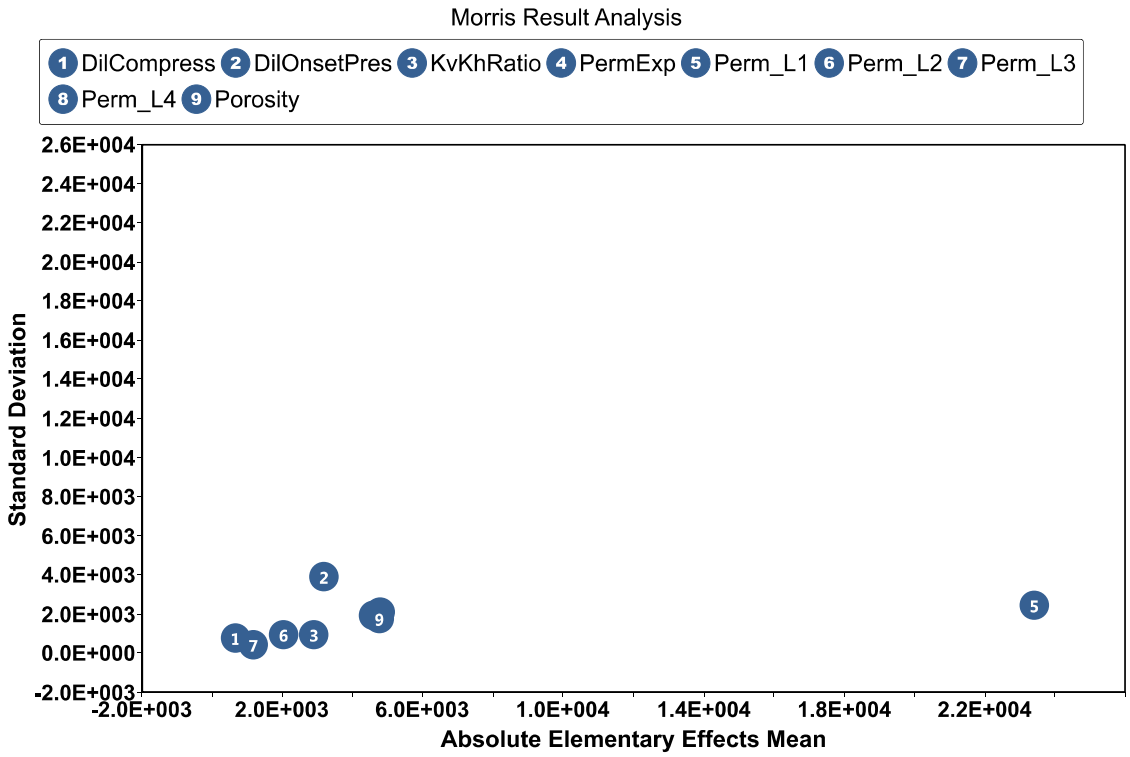


Figure 2.22: Morris analysis results for Cumulative Oil Production Objective Function

Table 2.7: Absolute elementary effects mean and standard deviation values for Cumulative Oil Production Objective Function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
Perm_L1	23426.691	2462.0648
Perm_L4	4804.1864	2101.3179
Porosity	4765.4586	1758.1085
PermExp	4608.3557	1937.4292
DilOnsetPres	3192.5541	3922.7191
KvKhRatio	2900.8129	958.47334

Perm_L2	2046.6206	982.73575
Perm_L3	1185.4842	451.83066
DilCompress	682.86063	796.02267

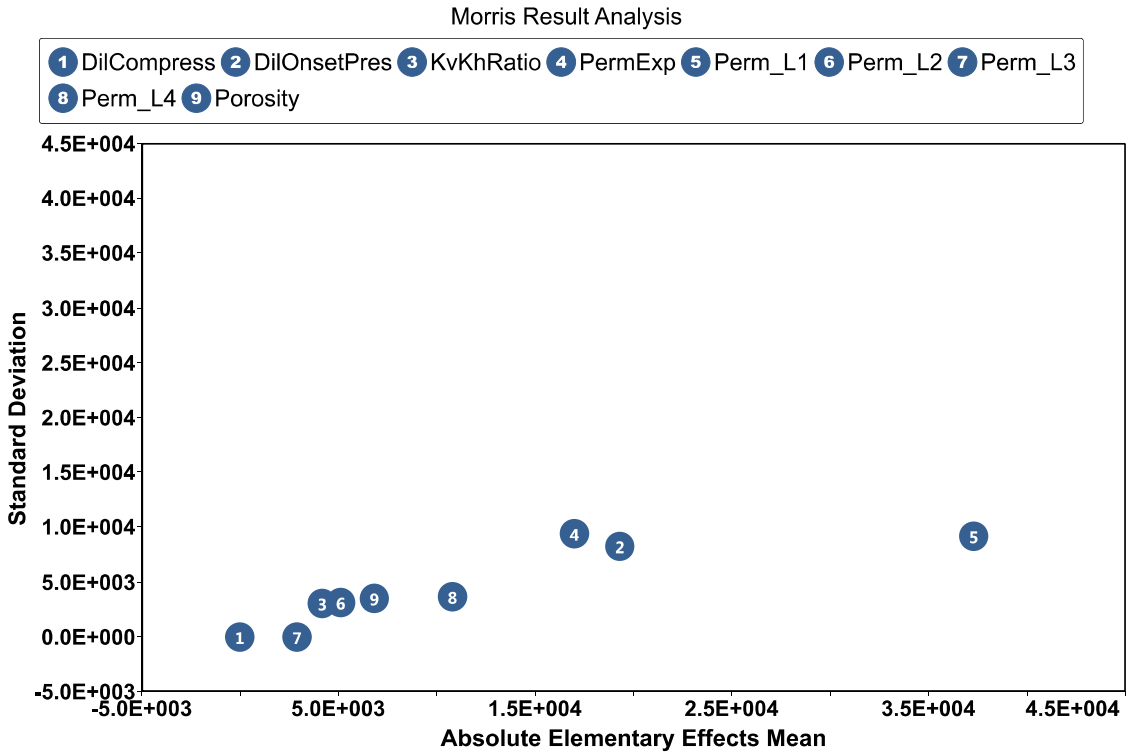


Figure 2.23: Morris analysis results for Cumulative Water Production Objective Function

Table 2.8: Absolute elementary effects mean and standard deviation values for Cumulative Water Production Objective Function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
Perm_L1	37298.357	9171.2405
DilOnsetPres	19313.925	8282.4511

PermExp	17009.35	9431.6227
Perm_L4	10827.851	3655.3468
Porosity	6849.7768	3477.1628
Perm_L2	5147.0792	3119.9474
KvKhRatio	4175.3693	3059.3926
Perm_L3	2902.8974	3.09E-05
DilCompress	0	0

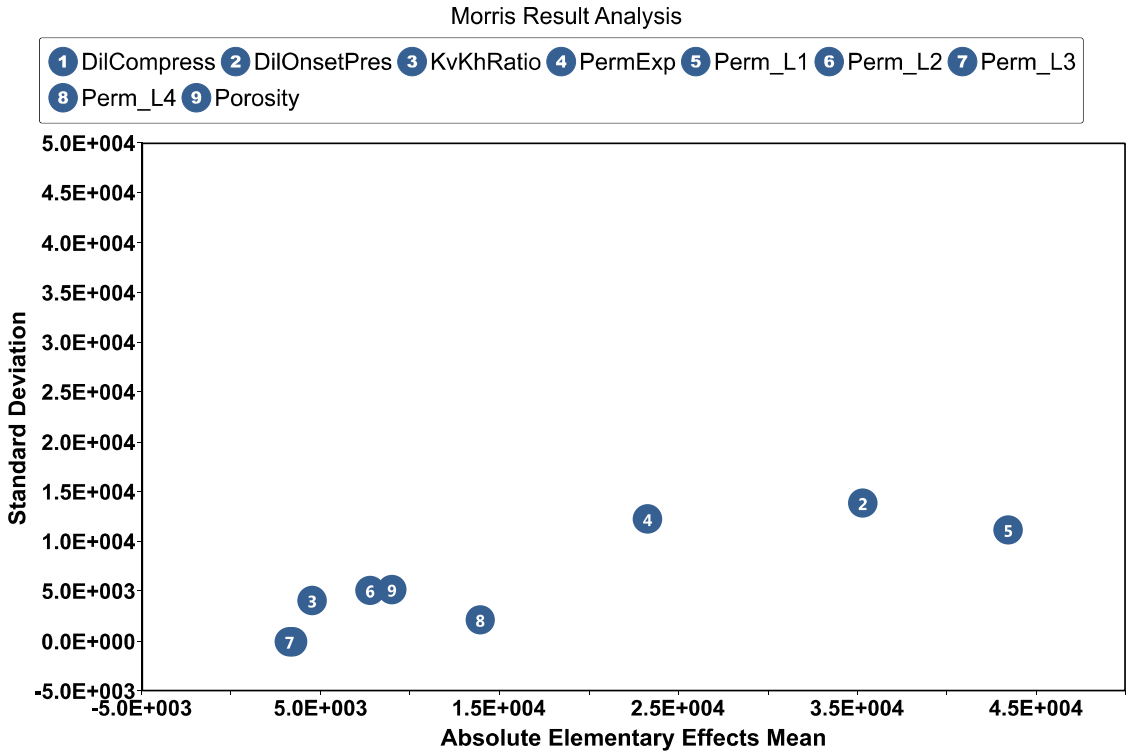


Figure 2.24: Morris analysis results for Cumulative Steam Injection Objective Function

Table 2.9: Absolute elementary effects mean and standard deviation values for Cumulative Steam Injection Objective Function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
Perm_L1	43446.994	11234.262
DilOnsetPres	35349.423	13886.031
PermExp	23308.314	12268.162
Perm_L4	13945.114	2189.1843
Porosity	9009.0722	5233.4737
Perm_L2	7765.565	5151.657
KvKhRatio	4525.4572	4107.658
DilCompress	3451.7614	2.12E-05
Perm_L3	3263.3999	2.87E-05

Based on the results, it is obvious that parameter 5 (Permeability in Layer 1) has the most impact on all three objective functions, and hence is the most influential factor in our model. While that is the case in general, parameter 2 (Dilation Onset Pressure) plays a more major role on Water Production and Injected Steam than Oil Production. At the same time, objective functions seem to be the least sensitive to parameters 7, 3 and 1 (Permeability in Layer 3, Kv to Kh ratio, and Dilation Compressibility).

Based on the overall relatively low values obtained for the Y axis of the Morris chart for all objective functions, we can conclude that strong interactions between parameters do not exist in the model and parameters have an almost linear effect on the outputs. This slightly changes for Cumulative Steam Injected, especially for parameter 2, followed by parameters

4 and 5 which indicates higher non-linear relation of these parameters to this objective function and\or stronger interaction effects between these and other parameters.

2.7.5 Tight oil reservoir

This case study will examine which factors affect well performance of hydraulically fractured wells in a tight oil reservoir.

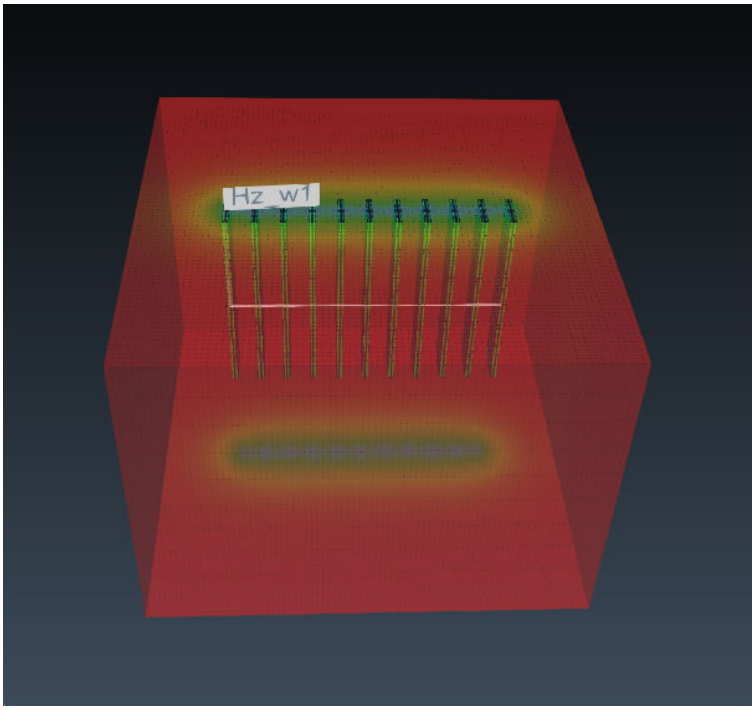


Figure 2.25: Hydraulically fractured well in a tight oil reservoir (Oil Potential)

For this purpose, a single well model has been created to represent a well in a tight oil reservoir (Figure 2.25), and we would analyze how the following parameters affect well

performance and use that information for a follow up history match using the same parameters:

- Matrix Permeability (Horizontal and Vertical)
- Matrix Porosity
- Fracture Half-length
- Fracture Permeability
- Fracture Height
- Relative Permeability Parameters:
    - Residual Oil to Gas (SORG, SOIRG)
    - Residual Gas Saturation (SGCRIT, SGCON)
    - Oil Relative Permeabilities at connate water (KROCW, KROGCG)
    - Gas Relative Permeability at connate liquid (KRGCL)
    - Corey Exponent for Oil (NOG)
    - Corey Exponent for Gas (NG)
- Formation Compressibility

The following parameters in Table 2.10 are therefore defined in the model:

Table 2.10: The parameters of the tight oil reservoir study

Parameter	Unit	Lower Limit	Upper Limit	Type
Compressibility	1/kPa	1E-6	1E-5	Continuous
FractureLayersdown	-	0	4	Discrete

FractureLayersup	-	0	3	Discrete
FracturePermeability	mD	2000	30000	Continuous
HalfLength	m	20	70	Continuous
KRGCL	-	0.4	1	Continuous
KROCW	-	0.4	1	Continuous
NG	-	2	3.5	Continuous
NOG	-	2	3.5	Continuous
PermH	mD	0.01	0.4	Continuous
KvKhRatio	-	0.05	0.5	Continuous
Porosity	-	0.075	0.125	Continuous
SGCRIT	-	0.01	0.06	Continuous
SORG	-	0.05	0.25	Continuous

We also define Cumulative Oil Production (m^3), Cumulative Gas Production (m^3), and GOR (Gas Oil Ratio) as three objective functions of this study. Using the same procedure as before, we create an experiment design using Morris sampling strategy, simulate each trajectory, and analyze the results by plotting the absolute mean elementary effects vs. the standard deviation for each objective function. Since we discussed these steps in detail in previous sections, we directly present the Morris results analysis in the following sections for each objective function.

2.7.5.1 Cumulative Oil Production

The Morris plot and the accompanying data table are presented below in Figure 2.26 and Table 2.11:

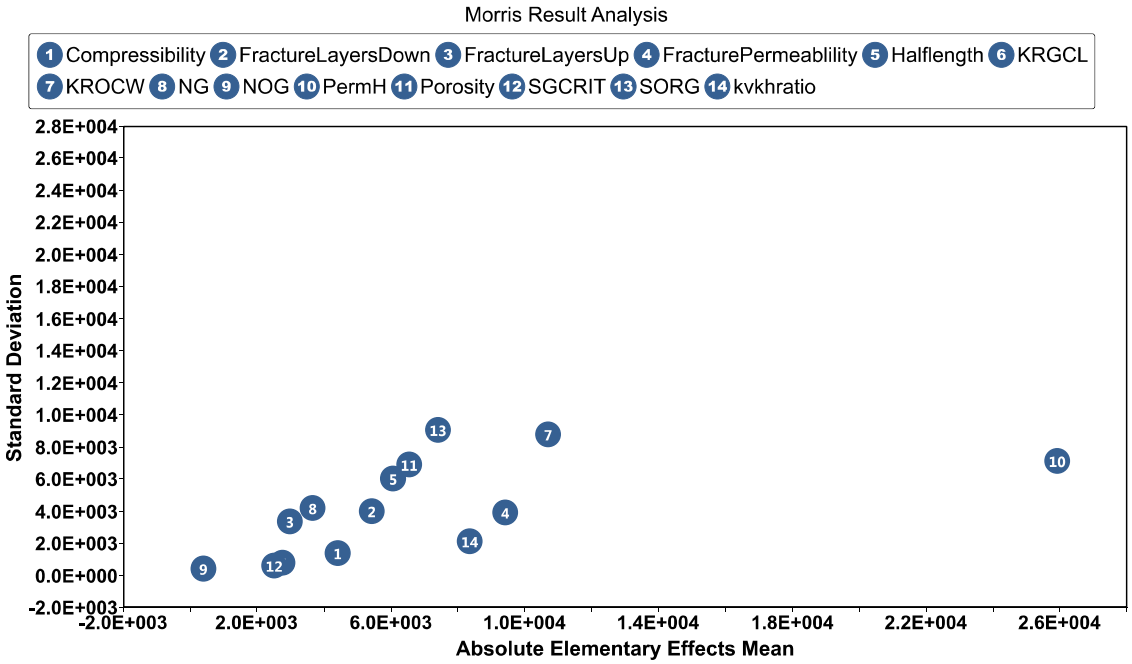


Figure 2.26: Morris analysis results for Cumulative Oil Production Objective Function

Table 2.11: Absolute elementary effects mean and standard deviation values for Cumulative Oil Production Objective Function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
PermH	25932.073	7147.2168
KROCW	10704.551	8811.3591
FracturePermeablility	9426.9771	3942.7798
kvkhratio	8371.7309	2124.4612
SORG	7419.1317	9070.6103
Porosity	6556.2425	6919.8161
Halflength	6071.8252	6058.5737
FractureLayersDown	5441.4044	4013.6412

Compressibility	4415.8816	1402.8684
NG	3668.5208	4209.757
FractureLayersUp	2991.341	3384.1171
KRGCL	2760.4194	825.03882
SGCRIT	2513.7298	620.41348
NOG	398.83123	442.48687

Based on the results above, Cumulative Oil Production objective function is mostly impacted by PermH parameter in this model. There are some interactions existing among the rest of the parameters for this objective function, but the amount seems not to be very significant.

2.7.5.2 Cumulative Gas Production

There is a linear relationship existing between the parameters and the objective function in this model for the Cumulative Gas Production as it can be observed in Figure 2.27 and Table 2.12 below:

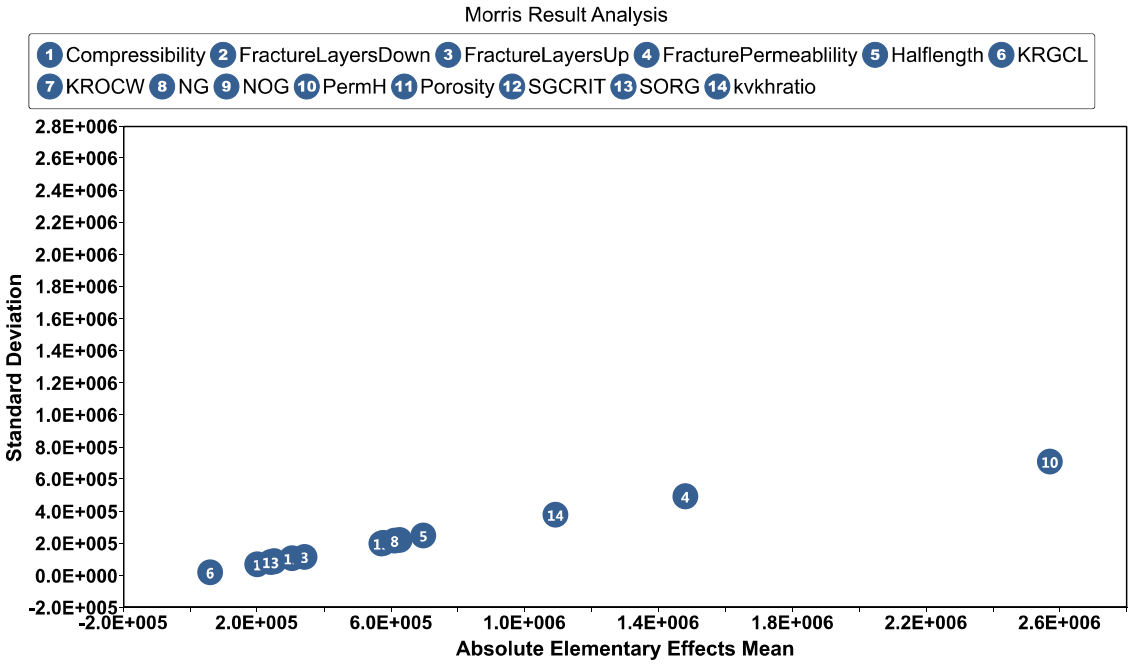


Figure 2.27: Morris analysis results for Cumulative Gas Production Objective Function

Table 2.12: Absolute elementary effects mean and standard deviation values for Cumulative Gas Objective Function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
PermH	2569882	711146.65
FracturePermeablility	1480557	494159.99
kvkhratio	1091790.7	380640.99
Halflength	697536.42	251342.01
KROCW	627004.69	224789.07
NG	610933.26	218252.19
FractureLayersDown	579738.49	203792.42
Porosity	571158.14	201008.32
FractureLayersUp	341704.35	119164.41

SGCRIT	304276.85	108877.02
NOG	251699.96	90612.735
SORG	241121.67	85755.378
Compressibility	200600.45	71594.823
KRGCL	59733.444	21658.658

The results show the high impact of Horizontal Permeability and Fracture Permeability on Cumulative Gas Production objective function.

2.7.5.3 GOR

The Morris results for GOR, demonstrate a high amount of interactions (non-linearity) for the parameters and the objective function as it can be observed in the Figure 2.28 and Table 2.13 below.

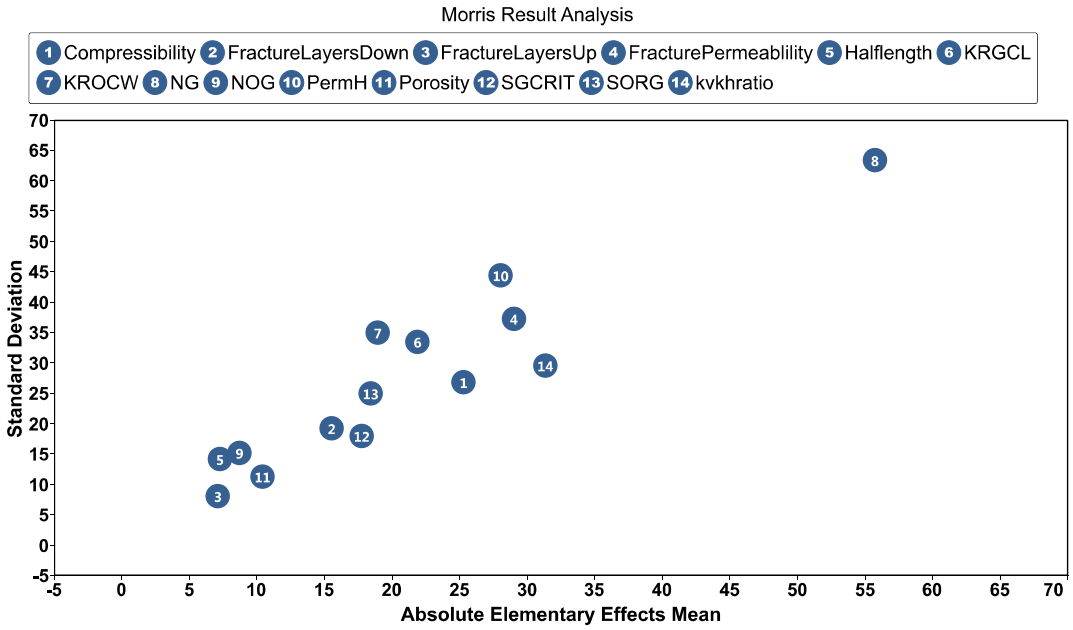


Figure 2.28: Morris analysis results for GOR Objective Function

Table 2.13: Absolute elementary effects mean and standard deviation values for GOR Objective Function

Parameter Name	Absolute Elementary Effects Mean	Standard Deviation
NG	55.746527	63.410448
kvkhratio	31.385794	29.613128
FracturePermeablility	29.0371	37.375662
PermH	28.062828	44.433401
Compressibility	25.313347	26.873887
KRGCL	21.919425	33.558472
KROCW	18.960486	35.03911
SORG	18.468441	25.013413
SGCRIT	17.801494	17.997552
FractureLayersDown	15.568036	19.270134
Porosity	10.439071	11.327892
NOG	8.7392161	15.231632
Halflength	7.3004165	14.155196
FractureLayersUp	7.1565734	8.1543781

For this objective function Corey Exponent for Gas is identified as the most significant factor.

This case study demonstrates the robustness of the Morris method in analyzing the sensitivities of the parameters of a model and categorizing them into different groups of importance while displaying indications of interactions or non-linear effect of the parameters on the objective function.

2.8 Summary and Conclusions

In this chapter, the Morris method of sensitivity analysis was presented. This screening allowed to classify the inputs in three groups: inputs having negligible effects, inputs having large linear effects without interactions and inputs having large non-linear and/or interaction effects. Morris method categorized the parameters by discretizing the input space for each variable, then performing a given number of OAT design. Such designs of experiments were randomly chosen in the input space, and the repetition of these steps allowed the estimation of elementary effects for each input. From these effects, sensitivity indices were derived. Overall, the Morris approach provided a good approximation of a global sensitivity measure, thereby overcoming the drawback OAT method whose results are only locally valid.

Chapter Three: A Robust Global Sensitivity Analysis Method for Complex Reservoir Systems

The focus of this chapter is on a global sensitivity analysis framework within which analysis takes place where we quantify uncertainty using variance of the response as the main metric. This method of sensitivity analysis allows us to better understand the effects of uncertainty in reservoir modeling in order to make well-informed decisions aimed at uncertainty reduction. It can be used to study and understand how variation in the output of a model can be allocated to different sources of input variation, either qualitatively or quantitatively, and therefore show how the given model or function is dependent upon the input information (Saltelli et al., 2008).

The variance-based SA methods are very useful when the model output uncertainty results from "epistemic uncertainty" in the model inputs. In such cases, knowing which input parameters are the main contributors to the uncertainty in the output is very useful. Variance-based sensitivity measures provide a ranking\sorting of parameters based upon their contribution to the output variance; hence, these measures can be useful by providing information on where study efforts should be focused, and what parameters and parameter combinations need to be better understood. This information then could be used to reduce the model output variance most effectively. Furthermore, variance-based SA methods have the important advantage of being model independent; which means these methods make no assumptions about the form or the structure of the model. This is in contrast to regression

methods for example. Consequently, they are not limited by model complexity and can be used for any type of problem in general. However, as we would see later in this chapter, these methods are highly demanding computationally and require many evaluations of the model. We will discuss and develop strategies how to address this problem later in this chapter as well. Although a variety of variance-based importance measures are available (Saltelli et al., 2008), the Sobol method of sensitivity analysis is one of the most robust and well suited methods for reservoir modeling applications and will be discussed in the following section.

3.1 Sobol Method of Sensitivity Analysis

Sobol method is a form of variance based sensitivity analysis method. The main idea similar to other variance based methods is to quantify the amount of variance that each input factor X_i contributes to the unconditional variance of the output $V(Y)$. Sobol method is defined within a probabilistic framework, and works by decomposing the variance of the model's output into fractions. These fractions could be attributed to the model inputs and interpreted as sensitivity measures. For example, given a model with two inputs and one output, we may find that 50% of the output variance is caused by the variance in the first input, 35% by the variance in the second, and 15% due to Interactions between the two inputs.

The data analysis in Sobol method based on variance-based measures of sensitivity is defined as:

- First-order sensitivity index, or main effect: this is the contribution to the output variance of the model which is due to the variation of X_i on its own. It is normalized by the total variance to provide a fractional contribution. (S_i) The first-order index represents the main effect contribution of each input factor to the variance of the output, and is used as an indicator of the importance of X_i on the Y, i.e. the sensitivity of Y to X_i. Various other names for this ratio can be found in the literature, including importance measure, and correlation ratio.

- Higher order interaction effects: input factors are said to interact when their effect on Y cannot be only expressed as a sum of their single effects. Interactions may imply, for instance, that values of the output Y are uniquely associated with particular combinations of model inputs, in a way that is not described by the first-order effects S_i just mentioned. Interactions represent important features of models, and are more difficult to detect than first-order effects.

- Total order effect: the sum of all the first and higher order effects that an input factor accounts for is called the total effect. So for an input X_i, the total sensitivity index S_{Ti} is defined as the sum of all indices relating to X_i. For example, having a model with three input factors (k = 3), the total sensitivity index for input factor S_1 would be: $S_{T1} = S_1 + S_{12} + S_{13} + S_{123}$.

Computing all order effects to obtain the total effect by brute force is not advisable when the number of input factors k increases, since the number of terms needed to be evaluated are as many as $2^k - 1$. Instead, there are techniques that enable us to estimate total indices at the same cost of first-order indices thus circumventing the so called curse of dimensionality. We compute the set of all S_i plus the set of all S_{Ti} to obtain a fairly good description of the model sensitivities at a reasonable cost. The methods on how to compute these indices are discussed later in this chapter.

Sobol variance-based measure of sensitivity is attractive because it handles nonlinear responses very well, it measures sensitivity across the whole input space, hence it is reliably a global sensitivity analysis approach, and it also works well in non-additive systems by measuring the effect of interactions effectively.

Sobol introduced the first order sensitivity index by decomposing the model function into summands of increasing dimensionality:

$$f(X) = f_0 + \sum_{i=1}^{d} f_i(X_i) + \sum_{i<j}^{d} f_{ij}(X_i X_j) + \cdots + f_{12\ldots d} \qquad (3.1)$$

This representation of the model function $f(X)$, holds if f_0 is a constant, and the integrals of every summand over any of its own variables are zero, i.e.:

$$\int_0^1 f_{i_1 i_2 \ldots i_s}(X_{i_1}, X_{i_2}, \ldots, X_{i_s}) dX_{i_k} = 0, for\ k = i_1, \ldots, i_s \qquad (3.2)$$

As a consequence of this, all the summands are mutually orthogonal. The total variance $V(Y)$ is defined as:

$$V(Y) = \int_{\Omega^k} f^2\,(X)dX - \ f_0^2 \qquad (3.3)$$

and the partial variances are computed from each of the terms in the decomposed model function:

$$V_{i_1 i_2 \dots i_s} = \int_0^1 \dots \int_0^1 f_{i_1 i_2 \dots i_s}^2 \left(X_{i_1}, X_{i_2}, \dots, X_{i_s}\right) dX_{i_i} \dots dX_{i_s} \qquad (3.4)$$

where

$1 \le i_1 < \cdots < i_s \le k \text{ and } s = 1, \dots, k$

If the function is analytically tractable, it is possible to calculate the indices above analytically through evaluating the integrals in the decomposition. However, in the vast majority of cases, Monte Carlo method is used for their estimation.

The Monte Carlo approach involves generating a sequence of randomly distributed points inside the unit hypercube. In practice for real applications, it is best to substitute random sequences with other sampling sequences to improve the efficiency of the estimators. This is a necessary step for reservoir modelling as Monte Carlo methods may sometimes lead to misleading and unreliable results, specifically in cases where a huge number of Monte Carlo simulations is not computationally feasible to achieve the desired results accuracy. We will examine this in a subsequent section, where the estimation accuracy of the Monte Carlo sampling is compared side by side with another strategy.

3.2 Latin Hypercube Design

Latin hypercube design (LHD) was developed by Mckay et al. (1979) to address the drawbacks of random sampling design. It is an effective approach to efficiently sample the search space with minimum number of experiments (Mckay et al., 1979).

In LHD, a probability distribution such as Gaussian or uniform is assigned for each input parameter to generate a set of samples. Suppose for example that our goal is to run n experiments, where each experiment is a different combination of the parameters' values of the model. Here, each parameter's uncertainty space is first divided into n-sections with the same probabilities. Then the n-generated samples of the first parameter are combined randomly with those obtained for the next parameter. Again, these n-paired samples are mixed with samples corresponding to next input parameter, and so on. This is done repeatedly in order to generate a final matrix design (Wyss and Jorgensen, 1998).

To further obtain optimal space-filling property for LHD, many studies have been done. Koehler and Owen (1996) demonstrated how projection of the optimal LHD onto a subset of variables retains good spatial properties. Jin et al. (2005) introduced the enhanced stochastic evolutionary algorithm for finding various space-filling designs. Ye et al. (1998) studied constructing of optimal symmetrical designs and developed a modified LHD called orthogonal Latin hypercube design (OLHD) to generate uncorrelated experiments. More details can also be found by referring to Helton and Davis (2001).

3.3 Quasi-Monte Carlo Experiment Design Approach

In addition to Latin Hypercube design, there are other experiment design approaches for better distributing points inside the unit hypercube called Low-discrepancy sequences. These sequences, also known as Quasi-Monte Carlo sequences, are specifically designed to generate samples of points as uniformly as possible over the unit hypercube. A low-discrepancy sequence is a sequence with the property that for all values of N, its sub-sequence has a low discrepancy. The reason Low-discrepancy sequences are also called quasi-random or sub-random sequences, is due to their common use as a replacement of uniformly distributed random numbers. The "quasi" modifier indicates that the values of a low-discrepancy sequence are not random or pseudorandom, but such sequences share some properties of random variables. In certain applications this feature of having lower discrepancy is a major contributing advantage. Opposite to the behavior of random numbers, successive quasi-random points know about the position of previously sampled points and tend to fill the space between them.

Sobol sequence is one example of quasi-random low-discrepancy sequences. It outperforms crude Monte Carlo sampling, Latin hypercube design and or orthogonal Latin hypercube design in the estimation of multi-dimensional integrals for our sensitivity analysis purposes. Therefore, it is discussed in this section and implemented efficiently to be used for Sobol SA method. Figure 3.1 demonstrates the advantage of Sobol low-discrepancy sampling vs. pseudo-random sampling.

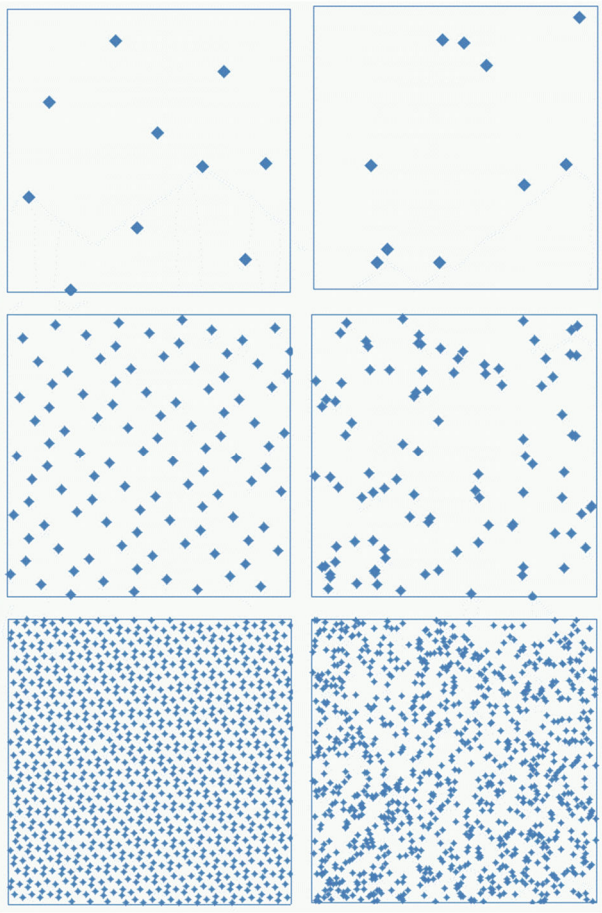


Figure 3.1: Sampling points from Sobol low-discrepancy sequence (left); compared with the points from the pseudo-random sampling (right). The Sobol sequence covers the space more evenly.

3.3.1 Low-discrepancy sequence sampling

Evaluating the integral of a function, $f(x)$, using Monte-Carlo method in the s-dimensional unit cube means calculating the average of the function at a set of points that are randomly sampled. When there are N sample points the integral is:

$$v = (\frac{1}{\mathrm{N}}\sum_{\mathrm{i=1}}^{\mathrm{N}} f(\mathrm{x}^{(\mathrm{i})}) \qquad (3.5)$$

Here v shows the approximation to the integral and $x^{(1)}, \ldots., x^{(N)}$ are N, s-dimensional, sample points. If a pseudo-random number generator is used the points $x^{(i)}$ will be independently and identically distributed.

If we set $X^{(\mathrm{i})} = f(\mathrm{x}^{(\mathrm{i})})$ then since $x^{(i)}$ is independently and identically distributed $X^{(\mathrm{i})}$ is also independently and identical distributed. The mean of $X^{(\mathrm{i})}$ is v and the variance is Δ^2 .

It is a well-known statistical property that the variance of n is given by $\frac{\Delta^2}{N}$. Therefore the estimated integral v has a standard error of $\Delta N^{-1/2}$, and the estimated error of the integral will decrease at the rate of $N^{-1/2}$. If we choose a Cartesian grid, sample each grid point exactly once, and selection points are chosen such that they lie on this grid, the Monte-Carlo method effectively becomes a deterministic quadrature scheme, whose fractional error decreases at the rate of N^{-1} or faster. There is a problem with this approach though, because it would be necessary to decide in advance how fine the grid needs to be, as well as the fact that all the grid points should be used. It therefore becomes not possible to sample until some convergence criterion has been met.

In this case, Low-discrepancy number sequences provide a medium between the flexibility of pseudorandom number generators and the advantages of a regular grid. They are

designed to have a high level of uniformity in multidimensional space, but unlike pseudo-random numbers they are not completely statistically independent. (Levy, 2011)

Table 3.1 below shows a sample for rates of convergence of Monte Carlo simulations versus the low discrepancy quasi-random sequence implemented.

Table 3.1: Comparison of integral estimate error for Monte Carlo vs low low-discrepancy sequence samplings

		Monte Carlo sampling	**low-discrepancy sequence**
Dimension	**N**	**Estimate Error**	
1	1,000	0.03162	0.001
1	100,000	0.00316	0.00001
2	10,000	0.01	0.0001
5	10,000	0.01	0.0001
10	10,000	0.01	0.0001
50	100,000	0.00316	0.00001

Since N equates to number of simulations and very large number of runs are not physically possible, the use of a low-discrepancy sequence is quite clear for our global sensitivity analysis purposes.

3.3.2 Orthogonality of the design

Orthogonality refers to the property of the experiment design which makes sure all specified parameters could be estimated independently of one another. The degree of orthogonality is measured by the normalized value of the determinant of the information matrix. An orthogonal design matrix having one row to estimate each parameter (mean, factors, and interactions) has a measure of 1. In order to check for the orthogonality, the design is orthogonal if the sum of the factors columns in standard format equals 0. The following categories in Table 3.2 are used to evaluate the experiment design quality:

Table 3.2: Experiment design quality categories

Perfectly Orthogonal	(0, 1e-8)
Nearly Orthogonal	(1e-8, 0.03)
Almost Orthogonal	(0.03, 0.2)
Not Orthogonal	(0.2, 1)

The definitions and categories above are used according to the classification in CMG CMOST^{TM}, a well-known commercial application that works in conjunction with reservoir simulators to perform sensitivity analysis, history matching, optimization, and uncertainty assessment based on design of experiments. (Computer Modelling Group, 2015). We test the quality of the implemented low-discrepancy sequence as we increase the number of experiments. The results are displayed in the Table 3.3 below:

Table 3.3: Low-discrepancy sequence orthogonality quality for different number of experiments

Number of experiments in the design	Minimum sample distance	Orthogonality value
16	0.88445254	0.70612284
32	0.7378403	0.18047223
64	0.69236311	0.15913404
128	0.56784193	0.17341673
256	0.41943877	0.021162374
512	0.33531418	0.01081669
1024	0.30882249	0.0045188653
2048	0.23216381	0.0028774907

An efficient implementation of Sobol sequence is implemented and used in this work to enhance the accuracy and performance of the employed sensitivity analysis methods.

3.4 Calculating Sobol Indices

To calculate the indices by the Quasi Monte Carlo method, we:

- Generate a $(N, 2k)$ matrix of random numbers (k is the number of inputs) using the Sobol low-discrepancy sequence described previously and define two matrices of

data (A and B), each containing half of the sample. N is called a base sample; to give an order of magnitude, N can vary from a few hundreds to a few thousands.

$$A = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_i^{(1)} & \cdots & x_k^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \cdots & x_i^{(2)} & \cdots & x_k^{(2)} \\ \vdots & & \ddots & & \vdots & \\ x_1^{(N-1)} & x_2^{(N-1)} & \cdots & x_i^{(N-1)} & \cdots & x_k^{(N-1)} \\ x_1^{(N)} & x_2^{(N)} & \cdots & x_i^{(N)} & \cdots & x_k^{(N)} \end{bmatrix} \quad (3.6)$$

$$B = \begin{bmatrix} x_{k+1}^{(1)} & x_{k+2}^{(1)} & \cdots & x_{k+i}^{(1)} & \cdots & x_{2k}^{(1)} \\ x_{k+1}^{(2)} & x_{k+2}^{(2)} & \cdots & x_{k+i}^{(2)} & \cdots & x_{2k}^{(2)} \\ \vdots & & \ddots & & \vdots & \\ x_{k+1}^{(N-1)} & x_{k+2}^{(N-1)} & \cdots & x_{k+i}^{(N-1)} & \cdots & x_{2k}^{(N-1)} \\ x_{k+1}^{(N)} & x_{k+2}^{(N)} & \cdots & x_{k+i}^{(N)} & \cdots & x_{2k}^{(N)} \end{bmatrix} \quad (3.7)$$

- Define a matrix C_i formed by all columns of B except the i^{th} column, which is taken from A:

$$C_i = \begin{bmatrix} x_{k+1}^{(1)} & x_{k+2}^{(1)} & \cdots & x_i^{(1)} & \cdots & x_{2k}^{(1)} \\ x_{k+1}^{(2)} & x_{k+2}^{(2)} & \cdots & x_i^{(2)} & \cdots & x_{2k}^{(2)} \\ \vdots & & \ddots & & \vdots & \\ x_{k+1}^{(N-1)} & x_{k+2}^{(N-1)} & \cdots & x_i^{(N-1)} & \cdots & x_{2k}^{(N-1)} \\ x_{k+1}^{(N)} & x_{k+2}^{(N)} & \cdots & x_i^{(N)} & \cdots & x_{2k}^{(N)} \end{bmatrix} \quad (3.8)$$

- Compute the model output for all the input values in the sample matrices A, B, and all C_i, obtaining vectors of model outputs of dimension $N \times 1$:

$$y_A = f(A), y_B = f(B), y_{C_i} = f(C_i) \quad (3.9)$$

These vectors are all we need to compute the Monte Carlo estimates of total and partial variances and hence being able to calculate first and total effect indices:

$$S_i = \frac{V\left[E\left(Y \mid X_i\right)\right]}{V\left(Y\right)} = \frac{\frac{1}{N}\sum y_A^{(j)} y_{C_i}^{(j)} - \frac{1}{N^2}\sum y_A^{(j)} \sum y_B^{(j)}}{\frac{1}{N}\sum \left(y_A^{(j)}\right)^2 - f_0^2} \quad (3.10)$$

Where

$$f_0^2 = \left(\frac{1}{N}\sum_{j=1}^{N} y_A^{(j)}\right)^2 \quad (3.11)$$

is the mean, and the symbol (·) denotes the scalar product of two vectors. Similarly, the method estimates total effect indices as follows:

$$S_{Ti} = 1 - \frac{V\left[E\left(Y \mid X_{\sim i}\right)\right]}{V\left(Y\right)} = 1 - \frac{\frac{1}{N}\sum y_B^{(j)} y_{C_i}^{(j)} - f_0^2}{\frac{1}{N}\sum \left(y_A^{(j)}\right)^2 - f_0^2} \quad (3.12)$$

The difference $S_{Ti} - S_i$ is a measure of how much X_i is involved in interactions with any other input factor.

In order to interpret Sobol results:

- Whatever the strength of the interactions in the model, S_i indicates by how much one could on average reduce the output variance if X_i could be fixed; hence it is a measure of main effect.

- By definition, S_{Ti} is greater than S_i, or equal to S_i in the case that X_i is not involved in any interaction with other input factors. The difference $S_{Ti} - S_i$ is a measure of how much X_i is involved in interactions with any other input factor.

- $S_{Ti} = 0$ implies that X_i is non-influential and can be fixed anywhere in its distribution without affecting the variance of the output.

- The sum of all S_i is equal to 1 for additive models and less than 1 for non-additive models. The difference $1 - \sum_i S_i$ is an indicator of the presence of interactions in the model.

- The sum of all S_{Ti} is always greater than 1. It is equal to 1 if the model is perfectly additive.

Sobol method is powerful in quantifying the relative importance of input factors as well as the interactions. The main drawback, similar to other variance-based methods, is the cost of the analysis, which in the case of computationally intensive models can become prohibitive even when using the approach described above that reduces the number of model executions. In terms of computational time, a large number of runs can sometimes be either trivial or unfeasible, depending on the model at hand. This problem is addressed by the application of advanced surrogate models discussed in the next section.

Another viable alternative for extremely (computationally) expensive models is the Morris method introduced previously. The elementary effect method is a good proxy for the total

sensitivity indices. If the model is both expensive to run and rich in factors it is recommended to use the elementary effect method to reduce the number of factors and then run a variance-based analysis on a reduced set of factors.

3.4.1 The role of Surrogate models

The estimation of sensitivity indices for Sobol SA method requires a high number of model evaluations coupled with the often times excessive run-time of reservoir simulation models, as crude discrete integration methods are applied. In this case, it is clearly impractical to perform a standard Monte Carlo approach that requires many thousand model evaluations. Therefore, time-consuming simulations are replaced by a faster-to-evaluate second model, which is called a surrogate model, proxy model, metamodel, emulator or response surface.

Surrogate models are tools for predicting the output from a complex simulation model based on an often limited set of observations from this model in the form of input/output mappings. The term surrogate model is used because this model, once created, serves as a quick-running surrogate to the actual model being analyzed in a way that multiple approximate simulations can be obtained with negligible computational demand from this surrogate. The use of surrogate models for performing sensitivity analysis of complex simulation models is advantageous because with a relatively limited set of observations from the simulation model we would be able to build our surrogate model and utilize it to make numerous model evaluations. Of course, as the surrogate model is only an approximation of the real model, a metamodel error has to be taken into account and the

accuracy of the prediction has to be assessed carefully before continuing with further evaluations.

In the next section, we discuss two different types of surrogate models that are found to perform consistently well in the sensitivity analysis cases considered.

3.5 Neural Network Based Surrogate Modelling for Sensitivity Analysis

The present part of the research utilizes a novel single-layer Neural Network surrogate model (proxy) to efficiently predict the production performance of hydrocarbon reservoirs from a limited number of reservoir simulations. The model is shown to provide powerful means for learning reservoir's dynamics from input-output relationships that is defined by multiple combinations of inputs and controls. Based on this, the model can be used as a surrogate for the Global Sensitivity Analysis application to overcome the computational costs associated with it. The workflow is organized as follows:

1. Different numbers of numerical simulations of the same reservoir are conducted for different combinations of the operational parameters.

2. The time series of the simulated performance are used as a training data to build Radial Basis Function (RBF) Neural Network which represents production performance as objective functions of the operational parameters and time. The total number of network neurons can vary from 1 to the total size of the training data set.

3. The proxy model is then used to predict the production data for the given reservoir for any time and any combination of the operational parameters.

4. The predicted data are compared with the actual simulation results for the same time period and the same combinations of the operational parameters to evaluate the prediction quality.

5. The verified model is used to perform analysis by the SA algorithm, replacing the need to run a very large number of simulations.

The proposed approach is shown to provide a very efficient forecasting mechanism for the models considered. The difference between the predicted and the actual data is at the order of few percent for the majority of the operational parameters.

3.5.1 Single-layer Neural Network surrogate model

First we consider the static interpolation in *D*-dimensional parameter space when time is fixed. In our context such interpolation is performed with the aid of single layer RBF Neural Networks. This method was proposed by Broomhead et al. (1988) which has capabilities similar to those of the multilayer Neural Network, but with a much faster training speed. The RBF network has its origin in performing the exact interpolation of a set of data points in a multi-dimensional space and, in that respect, is similar to various Kriging and Kernel models.

A Radial Basis Function (RBF) network consists of an input layer of source nodes, a single hidden layer of nonlinear processing units, and an output layer of linear weights. The input-output mapping performed by the RBF network can be described as:

$$\tilde{y}(x_p) = w_{M+1} + \sum_{i=1}^{M} w_i \varphi(x_i, x_p) \quad (3.13)$$

where $\varphi(x_i, x_p)$ is the radial basis function (or neuron function) that depends on the distance between the input parameter vector x_p and the center x_i and, in general case, on mutual orientation of those vectors in *D*-dimensional parameter space. The distance and the orientation scales, as well as precise shape of the radial function are fixed parameters of the model. Here *M* (the total number of neurons) can vary in the range $1 \leq M \leq N$, where *N* stands for the size of the training data set (the total number of training simulations in our case). However, the network can provide more or less accurate prediction only with substantial number of neurons, starting from $M \approx N/2$.

RBFs are simply a class of neuron functions that could be employed to approximate any function and as a building block in a network. In this work, the RBF is best approximated by the following power function:

$$\varphi(x_i, x_p) = 0.01 \left\| x_i - x_p \right\|^{1.5} \quad (3.14)$$

where $\left\| x_i - x_p \right\|$ stands for the Euclidian distance between points x_i and x_p.

The parameters w_i are estimated by the minimization of the error between the forecasted value (3.13) and the exact simulation output as

$$E(\vec{w}) = \sum_{i=1}^{N} \{y_i - \tilde{y}(x_i, \vec{w})\}^2 \quad (3.15)$$

where values y_i stand for an exact output of the objective function Y for i^{th} simulation in the training set, and the value $\tilde{y}$ is defined by the Eq. (3.13). In our case the minimization of the error (3.15) is complemented by the additional constraint of total sum of neuron weights to be equal to 1 as

$$\sum_{i=1}^{M} w_i = 1 \quad (3.16)$$

In the case $M < N$ (the number of neurons is less than the size of the training data set) there is a problem of selection of M "center" points in Eq. (3.13). In this study we select those M points randomly from the original Latin Hypercube data set.

Minimization of the error (3.15) along with the constraint (3.16) results in the following linear system of equations for coefficients w_i:

$$\sum_{\beta=1}^{M+1} \Delta_{\alpha\beta} w_\beta = R_\alpha, \quad (3.17)$$

$$R_\alpha = \sum_{i=1}^{N} y_i \varphi(\overrightarrow{x_i}, \overrightarrow{x_\alpha}) \quad (3.18)$$

where Δ is a constraint version of the Moore-Penrose matrix (Bishop, 2009):

$$\Delta_{\alpha\beta} = \theta(M - \beta)\left[\sum_{i=1}^{N} \varphi(\overrightarrow{x_i}, \overrightarrow{x_\alpha})\varphi(\overrightarrow{x_i}, \overrightarrow{x_\beta})\theta(M - \alpha) + \delta_{\alpha,M+1}\right] + \theta(M - \alpha)\delta_{\beta,M+1} \quad (3.19)$$

Here $\delta_{\alpha,\beta}$ and $\theta(x)$ stand for Kronecker symbol and Heaviside step-function, respectively. Even in the case $M < N$ all training data contributes to the definition of the Moore-Penrose matrix and the right-hand side vector, as it can be seen from Eqs. (3.18), (3.19). In the case of $M = N$ (the number of neurons is equal to the size of a training data set), Eq. (3.17) leads to the exact match of the forecasted outcome for the training parameters and the actual simulation output. Smaller numbers of neurons introduce some error for the training objective function. As it will be shown below, this can result in somewhat better accuracy for the verification data in the case of very large sizes of training data set. We will call the RBF Neural Network with $M < N$ a truncated RBF with $m = M / N$.

Parameter values are provided to the neuron functions in Eq. (3.13) in dimensionless and normalized form, according to the following expression (Myres & Montgomery, 2002):

$$x_i = \frac{z_i - [max(z_i) + min(z_i)]/2}{[\max(z_i) - min(z_i)]/2} \quad (3.20)$$

$\forall i \in (1, \dots, M)$, where $\max(z_i)$ and $\min(z_i)$ are the maximum and minimum values of i^{th} parameter defined from the training data set.

While RBF method as it is presented by Eq. (3.13) is not new in reservoir modeling, its current version defined by Fedutenko et al. (2013) based on the combination of the neuron function (3.14) and the weight constraint (3.16) has shown to be particularly accurate for

our purposes. Note that the fractional power distribution of the Radial Basis Function in Eq. (3.14) may be a result of fractal dimensionality of variograms of irregularly spaced data (Kentwell et al., 1999).

The shortcoming of the method presented above is the poor extrapolation capability, i.e. forecasting quality for parameters outside of the maximum and minimum values of training data set is significantly lower than the one for parameters within them. This is why the proposed proxy has to be used with a proper experimental design which includes both minimum and maximum parameter values of the current study.

The time axis cannot be considered simply as an additional "operational dimension" to those of the space domain as it can result in invalid prediction for some parameter combination (Myers et al., 1990, Rouhani et al., 1990). Therefore, in their previous work (Fedutenko et al., 2013) the space-time separability hypothesis based on 1D cubic spline interpolation is introduced in the time domain. This research employs the same approach.

The following algorithm is used to perform the time interpolation for any given combination of parameters:

1. Define the time series of particular output (either any field cumulative production value, production or injection rate, or ratios like SOR cumulative) of i^{th} training simulation as $Y_i(p_1^{(i)}, \dots, p_M^{(i)}; t)$.

2. Divide the total simulation time horizon T into $K=60$ time steps with $\Delta t = T/K$.

3. $\forall t_\alpha = \alpha\, \Delta t$, $0 \leq \alpha \leq K-1$, obtain the values $K_{i\alpha} = Y_i(p_1^{(i)}, \dots, p_M^{(i)}; t_\alpha)$.

4. $\forall\alpha, 0 \leq \alpha \leq K-1$ build K D-dimensional space interpolators for the time $t_\alpha = \alpha\, \Delta t$.

5. For the new set of parameters perform K interpolations $\forall t_\alpha = \alpha\, \Delta t$ by using the correspondent interpolator from the Step 4.

6. For any arbitrary intermediate time $t_\alpha \leq \tau \leq t_{\alpha+1}$ perform 1D cubic spline interpolation based on the interpolated values of objective functions at time steps $t_\beta, \forall\alpha, 0 \leq \beta \leq K-1$ (Press et al., 2007).

The discussed RBF Neural Network proxy model is used successfully as a necessary component of the sensitivity analysis framework to address the computational cost. Examples are provided in the case studies section.

3.5.2 Multi-Layer Neural Network surrogate model (Levenberg-Marquardt Optimization)

Although the RBF Neural Network proxy model has very good prediction capabilities in many cases, its configuration is fixed, and in situations where the desired accuracy is not achieved, the proxy model is simply not reliable anymore for further use and there is no flexibility to address it. In order to overcome this shortcoming in such cases, a multi-layer neural network is introduced. It is advantageous over the RBF neural network proxy because it is adjustable in its configuration, and it provides better accuracy than RBF when right network configuration is used. It is also faster to predict than RBF. At the same time,

due to its flexibility, its accuracy is sensitive to the network configuration and if the right network configuration is not used it would produce less accurate results than RBF.

Multi-layered Neural Network (ML NN) consists of

1. An Input Layer (*D*-dimensional input parameter)
2. Any number of Hidden Layers (HL) with any number of nodes in each hidden layers
3. An Output Layer (OL) with the same number of nodes as in the last HL

Basic architecture of multi-layer neural network is presented in the Figure 3.2 below:

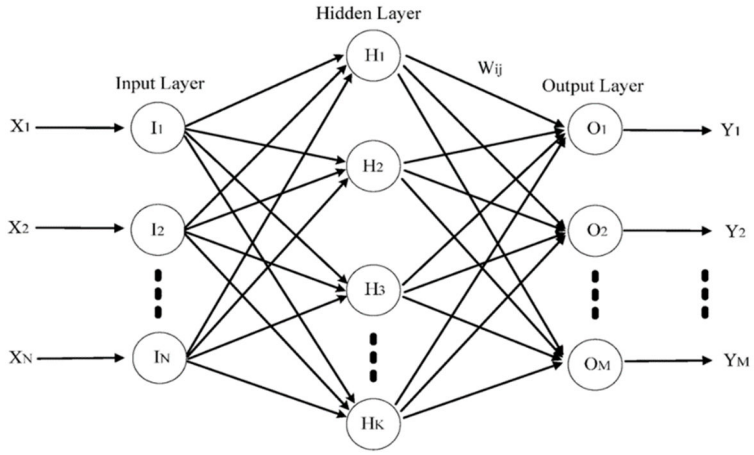


Figure 3.2: Basic multi-layer neural network architecture

Assuming that multi-layer neural network contains *M* hidden layers, any Objective Function (OF) is represented by the output layer as:

$$\tilde{Y} = T\left\{\sum_{j=1}^{L_M+1} H_j{}^{(M)} w_j\right\} \quad (3.21)$$

where Transformation Function $T(z) = \frac{1}{1+e^{-z}}$ for $z \geq 0.0$ or $T(z) = TanH(z)$ for $-\infty \leq z \leq \infty$, L_M stands for the number of neurons in the M^{th} hidden layer, $H_j{}^{(M)}$ stands for the output of the M^{th} hidden layer (its dimensionality is $L_M + 1$, and $H_{L_M+1}{}^{(M)} \equiv -1.0$), and w_j stands for the weight vector of the output layer.

For any K^{th} hidden layer ($1 \leq K \leq M$), an output is represented as

$$H_j{}^{(L)} = T\left\{\sum_{l=1}^{L_{K-1}+1} H_l{}^{(K-1)} w_{lj}{}^{(K)}\right\} \quad (3.22)$$

where $H_l{}^{(K-1)}$ is an output of the previous, $(K-1)^{\text{th}}$ HL (assuming that $H_l{}^{(1)} = P_l$, where P_l is an input parameter, $1 \leq l \leq D, H_{D+1}{}^{(1)} = -1.0$), and $w_{lj}{}^{(K)}$stands for the weight matrix of the K^{th} hidden layer.

The goal of the neural network training is to optimize all the weight matrices of hidden layers and the weight vector of the output layer according to the provided training data set. The optimization target is the minimization of the mean-square error for the given weight distribution:

$$E^2(\vec{w}) = \frac{1}{N-1}\sum_{i=1}^{N}\left[Y\left(\vec{x}^{(i)}\right) - \tilde{Y}(\vec{x}^{(i)}, \vec{w})\right]^2 \quad (3.23)$$

where Y and $\tilde{Y}$ are the simulator's and proxy's outputs for Objective Functions, respectively. Every time for the set of prediction, the gradient error information is propagated back, as presented in the Figure 3.3:

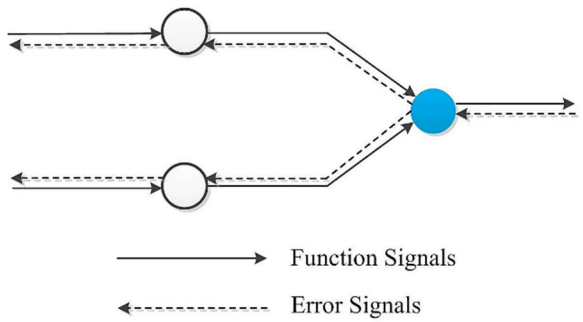


Figure 3.3: Multi-Layer Training by Error Analysis

The implementation of multi-layer neural network used in this research utilizes the gradient-based Levenberg-Marquardt Algorithm of the NN optimization:

1. Randomly populate initial weights of the Network
2. For N training parameters, compute nonlinear output (OF) according to a set of nonlinear rules
3. Compute an error by comparing with the actual OF for that parameter
4. Update the model's weights and coefficients to find such weight correction $\vec{h}$ that minimizes $\chi^2(\vec{w}+\vec{h})$ where $\vec{w}$ is the current weight

5. Each layer weights update at i+1[th] iteration step according to the minimization equation $\vec{w}_{(i+1)} = \vec{w}_{(i)} + \vec{h}$,

$$\left[\hat{J}^T\hat{J} + \lambda\hat{I}\right]\vec{h} = \hat{J}^T\left[\hat{\vec{y}} - \vec{y}\right] \quad (3.24)$$

6. $\hat{J}$ is a Jacobian (second derivatives contribution is usually neglected), if calculated properly for each layer – makes it possible to be used for any number of layers

7. λ is a control parameter balancing between Gradient ($\lambda \to \infty$) and Newtonian ($\lambda \to 0$) BEP

The crucial part of any gradient-based optimization algorithm is the proper evaluation of derivatives of an objective function with respect to the neural network weights. In our case it is done analytically, under an assumption that Transformation Function is differentiable as $T'(z) = F(T(z))$. Assuming that we have *M* hidden layers, it can be shown that for the output layer

$$\frac{\partial\tilde{Y}}{\partial w_\alpha} = H_\alpha^{(M)} F\left(\tilde{Y}\right) \quad (3.25)$$

where $H_\alpha^{(M)}$ is an output of M^{th} (the last) hidden layer. Then it can be proven (by Mathematical Induction, see Appendix A) that for each Hidden Layer

$$\frac{\partial\tilde{Y}}{\partial w_{\alpha\beta}^{(M-K)}} = H_\alpha^{(M-K-1)} G_\beta^{(M-K)} \quad (3.26)$$

where $0 \le K \le M - 1$, $H_\alpha^{(0)}$ is an input parameter, and the function *G* is defined as

$$G_{\beta}{}^{(M-K)} = F\left(H_{\alpha}{}^{(M-K)}\right) \sum_{\mu=1}^{N_{M-K+1}} w_{\beta\mu}{}^{(M-K+1)}\, G_{\beta}{}^{(M-K+1)} \qquad (3.27)$$

where for $K = 0$ (the last, M^{th} Hidden Layer) the value of $G_{\beta}{}^{(M)}$ can be easily calculated as

$$G_{\beta}{}^{(M)} = F\left(H_{\beta}{}^{(M)}\right)F\left(\tilde{Y}\right)w_{\beta} \qquad (3.28)$$

w_{β} is output layer weights, N_i is a number of nodes in i^{th} hidden layer.

Equations (3.25) – (3.28) provide simple and reliable recurrent algorithm of back-propagation of derivative information from Output Layer through all Hidden Layers back to the Input Layer in accordance with the general approach of Figure 3.3.

3.6 Case Studies

3.6.1 Polynomial function

To demonstrate the proposed method in action, here the same polynomial function used as an example in previous chapter is demonstrated here. We know from before that it is a simple polynomial function with 3 parameters, with X_2 being the most important input parameter, and interactions existing between parameters X_1 & X_2:

$$Y = X_1 + 10X_2 + X_3 + 50X_1X_2 \quad (3.29)$$

All parameters are again uniformly distributed between [0,1]. The Results from Sobol analysis is displayed in Figure 3.4:

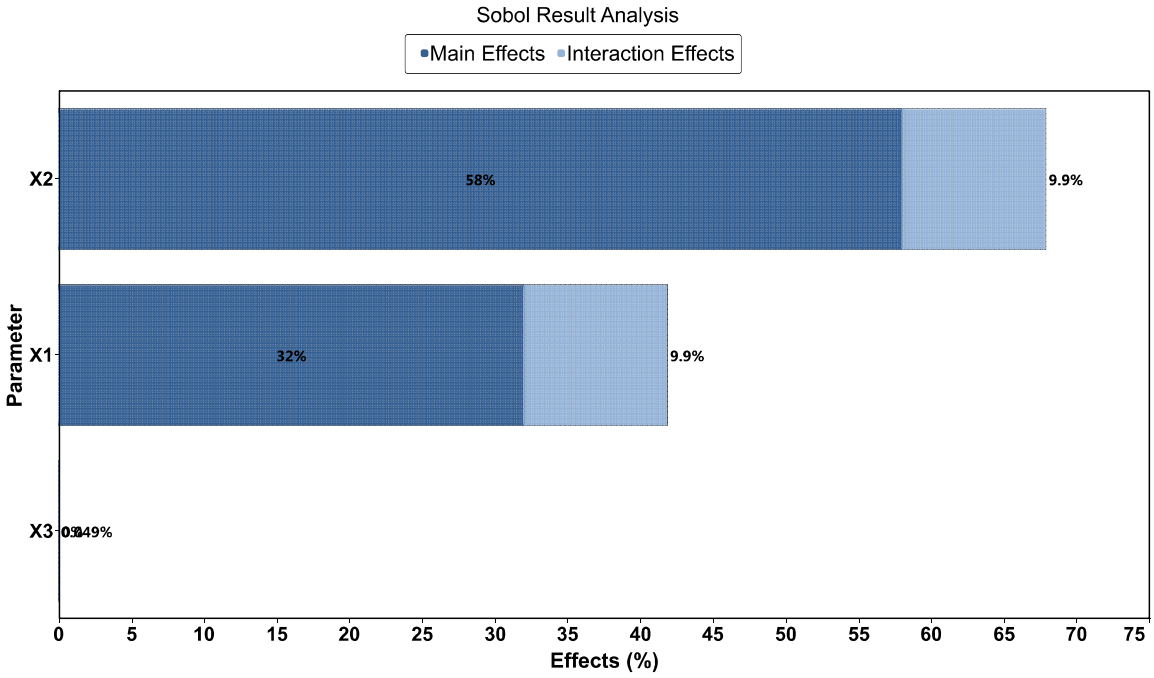


Figure 3.4: Sobol chart for the polynomial Objective Function

In this example, it is clear that parameter X_2 is the biggest contributor to the variability of the objective function. Based on the results from Figure 3.4 and Table 3.4, 58% of the output variance, on average, could be reduced if X_2 could be fixed. This is followed by parameter X_1 that contributes about 32% to the variability of the objective function.

Table 3.4: Sobol values for the polynomial Objective Function

Parameter Name	Main Effects	Interaction Effects	Total Effects
X2	58%	~ 10%	68%
X1	32%	~ 10%	42%
X3	~ 0%	0%	~ 0%

The difference $1 - \sum_i S_i = 1 - (0.58 + 0.32)$ is an indicator of the presence of about 10% interactions in the model. Since the interaction effect for parameter X_3 is zero, we can conclude that the interactions are happening between parameters X_1 and X_2 only. This could be verified by checking the polynomial example function as well.

Figure 3.5 displays side by side comparison of the obtained results versus the Morris analysis from the previous chapter.

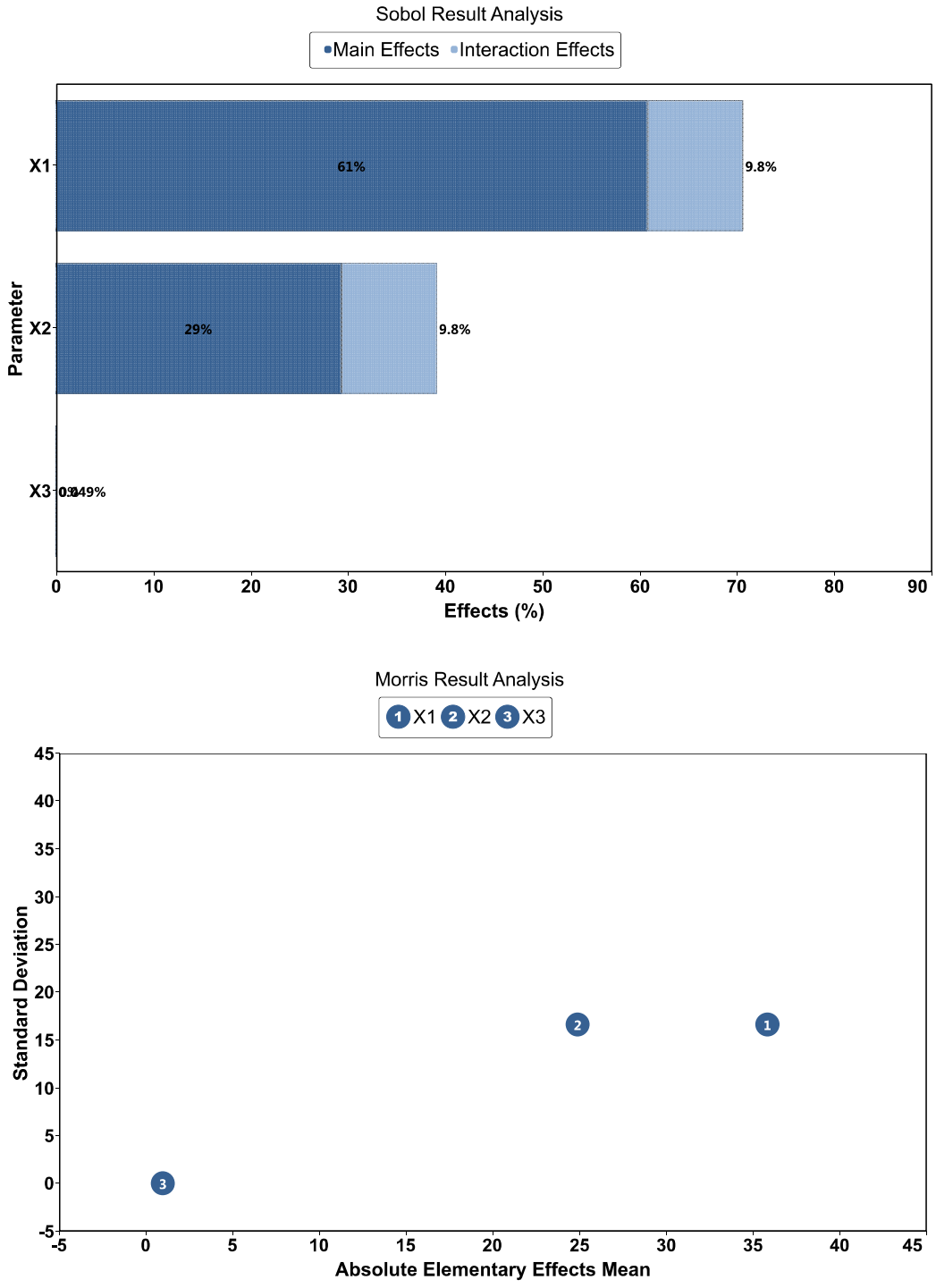


Figure 3.5: Side by side comparison of Morris and Sobol results

3.6.2 Ishigami function

In the previous chapter, Ishigami function was introduced as a commonly used test case for sensitivity analysis exhibiting strong non-linear and non-monotone behavior.

$$f(x) = Sin(X_1) + a\,Sin^2(X_2) + bX_3^4\,Sin(X_1) \qquad (3.30)$$

where $a = 7$ and $b = 0.1$.

The analytical indexes of Ishigami function rank the overall influence of three input parameters as $X_1 > X_2 > X_3$. The expected results in terms of main effects, interaction effects and total effects are summarized in the Table 3.5:

Table 3.5: Analytical sensitivity analysis indexes for Ishigami function

First order effects	S_1	0.31
	S_2	0.44
	S_3	0
Interaction effects	S_{12}	0
	S_{23}	0
	S_{13}	0.24
	S_{123}	0
Total effects	S_{T1}	0.56
	S_{T2}	0.44
	S_{T3}	0.24

We calculate the Sobol indices and compare the results with the above table.

3.6.2.1 Sampling algorithm of parameters

8192 (2^{13}) experiments are created using the low-discrepancy sequence introduced previously. We can view the progress of the simulation runs in terms of study parameters through the parameter run progress plots in Figures 3.6, 3.7, and 3.8. In run progress plots, each blue data point represents one simulation experiment. As the run progresses, more and more experiments (data points) are added. By examining run progress plots, we can observe how our sampling algorithm is picking the data points and covering the parameter space. This will help us for example to determine how even and space filling our sampling strategy is and whether any parameter ranges are being over\under sampled.

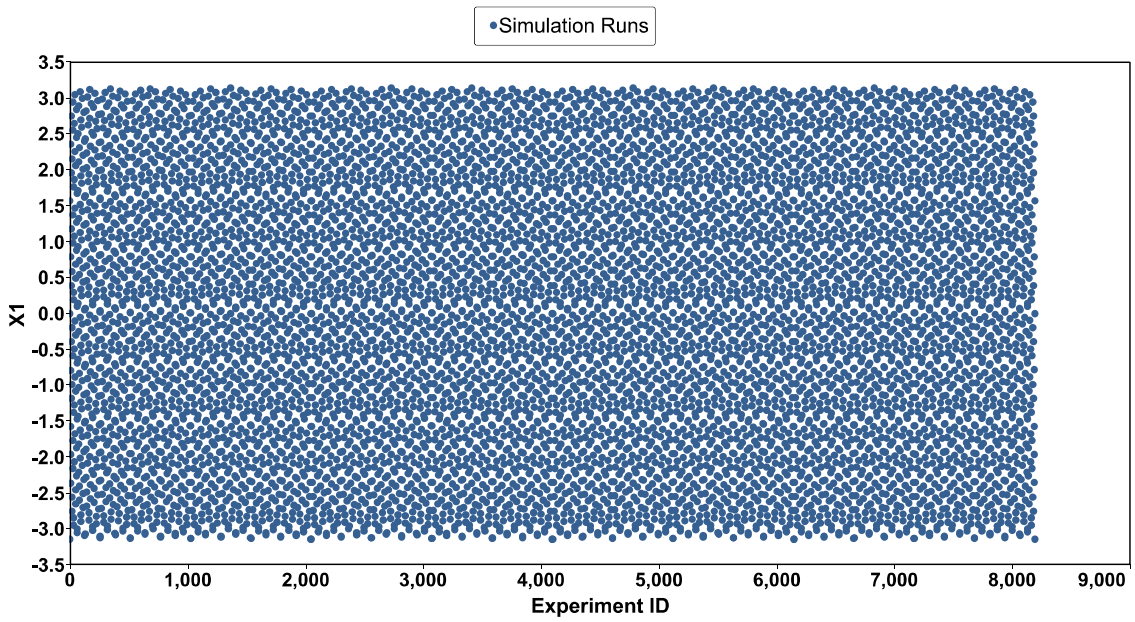


Figure 3.6: Run progress for parameter X_1

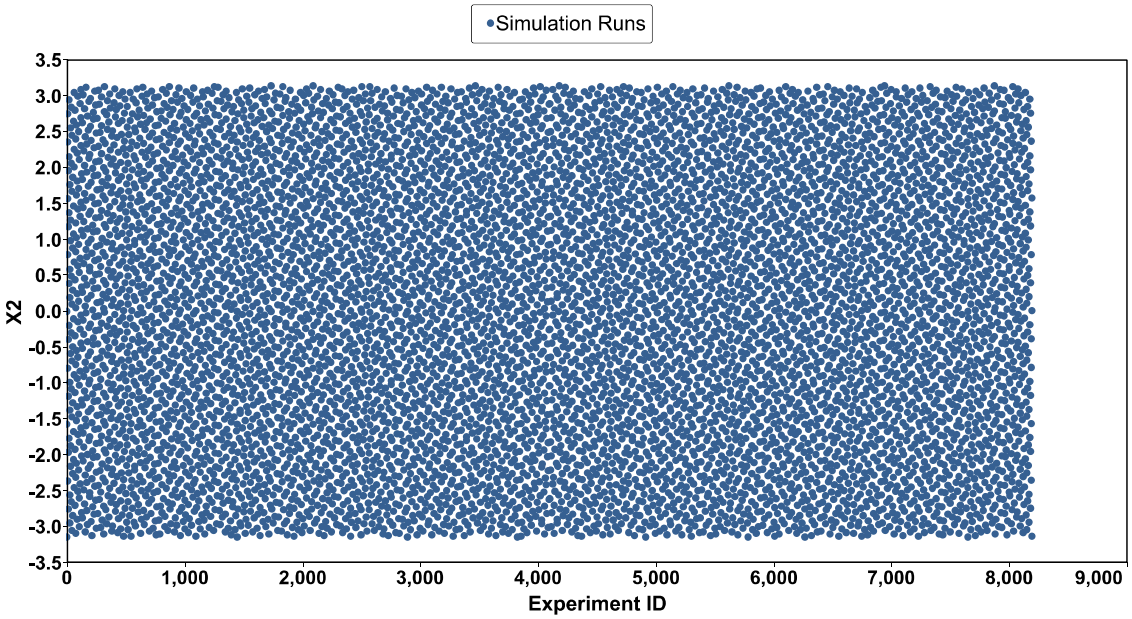


Figure 3.7: Run progress for parameter X_2

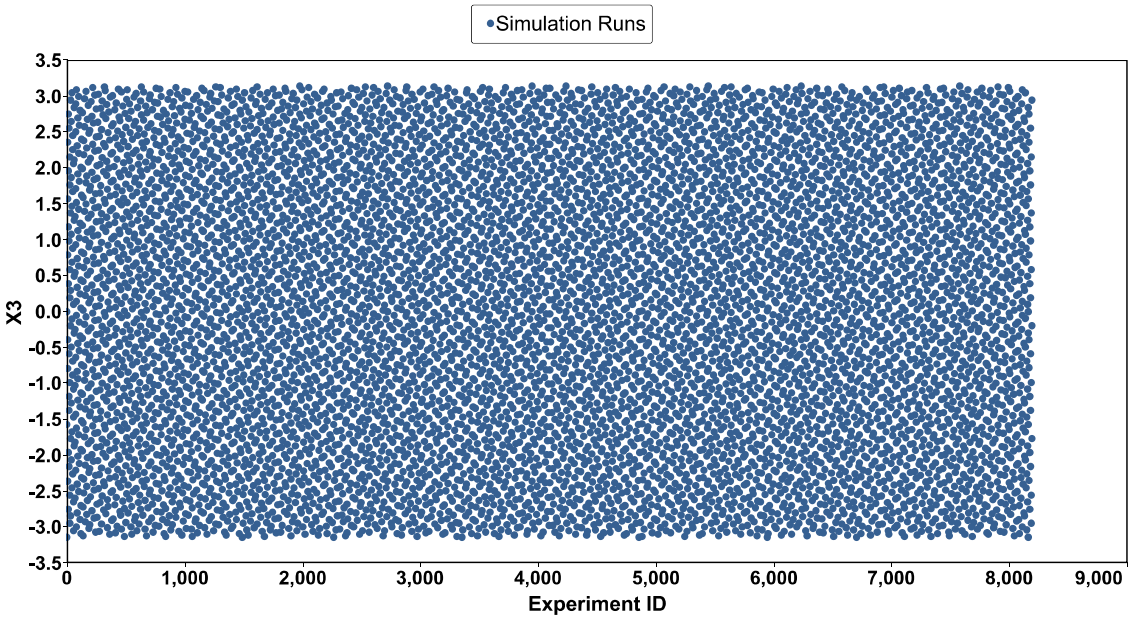


Figure 3.8: Run progress for parameter X_3

3.6.2.2 Comparison with LHD

For the sake of comparison, we generate the same number of experiments this time using the Latin Hypercube sampling and plot the run progress in Figures 3.9 to 3.11. Note that in order to find the optimum results, we iteratively generate Latin hypercube samples to find the best one for 500 iterations.

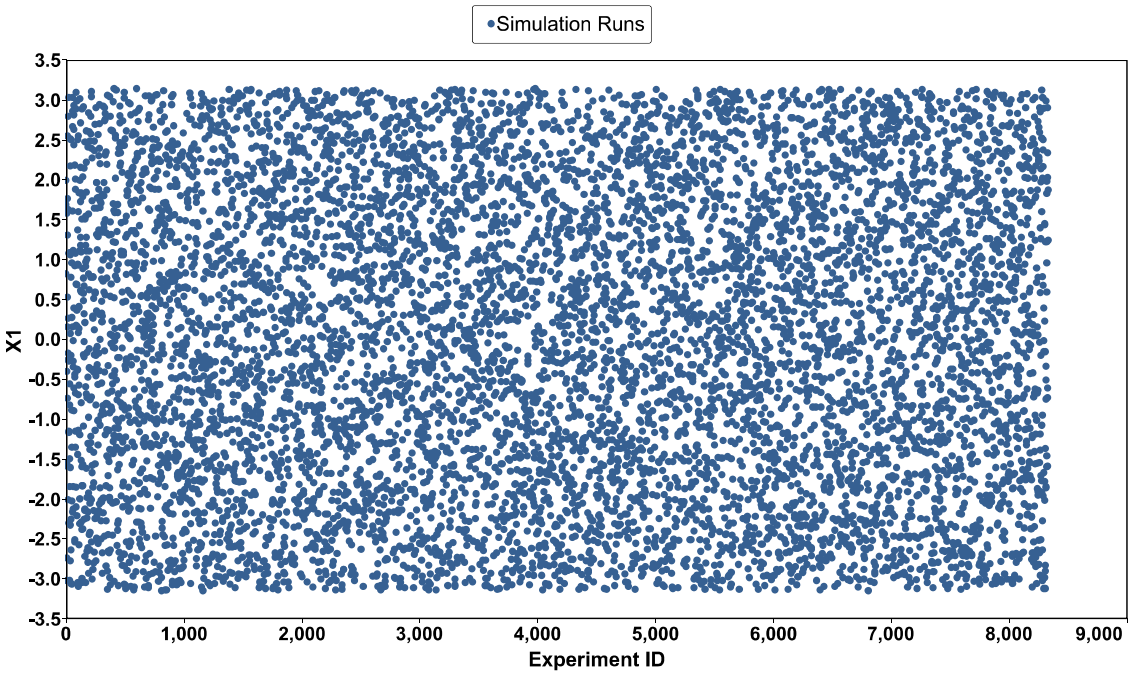


Figure 3.9: Run progress for parameter X_1 using Latin Hypercube Design

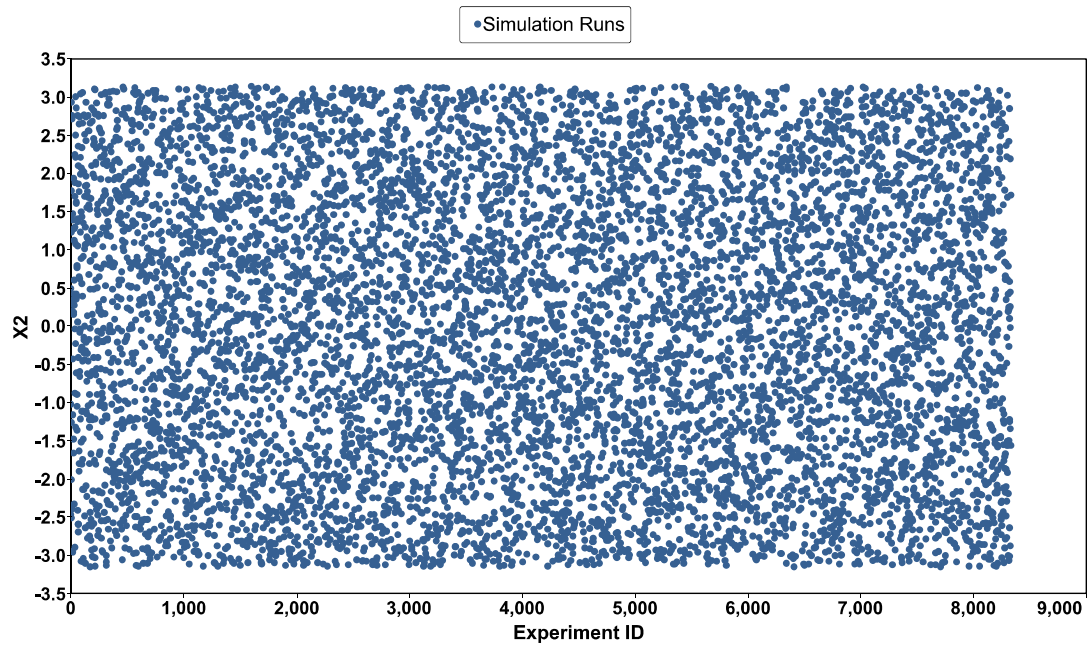


Figure 3.10: Run progress for parameter X_2 using Latin Hypercube Design

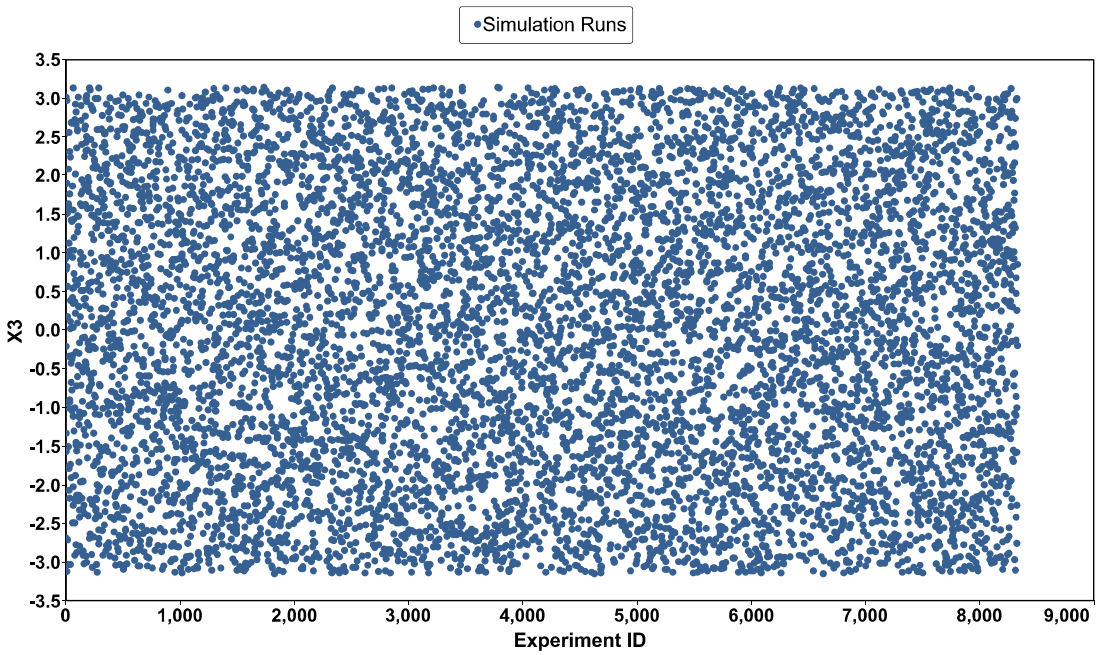


Figure 3.11: Run progress for parameter X_3 using Latin Hypercube Design

The comparison between low-discrepancy sequence and Latin Hypercube is shown in Table 3.6.

Table 3.6: Comparison of low-discrepancy sequence and Latin Hypercube design quality

	Orthogonality value	Minimum sample distance
LHD	0.00859	0.00078
Low-discrepancy sequence	9.388E-7	0.01341

The comparison shows we can expect more accurate results based on the low-discrepancy sequence. Therefore, we continue to evaluate the Ishigami function using the low-discrepancy sequence sampling that we generated. Using this sampling strategy, output of the Ishigami function with respect to each input parameter is calculated. Once the model evaluations are complete, we calculate the main effect, total effect and interaction sensitivity indexes for Ishigami function. The summery of the results are shown in Table 3.7 and Figure 3.12.

Table 3.7: Main, Interaction, and Total effects for Ishigami function

Parameter Name	Main Effects	Interaction Effects	Total Effects
X_1	0.314974	0.242516	0.557489
X_2	0.442467	0	0.442467
X_3	0.00160296	0.242095	0.243698

The results match the expected values for the analytical values presented in Table 3.5.

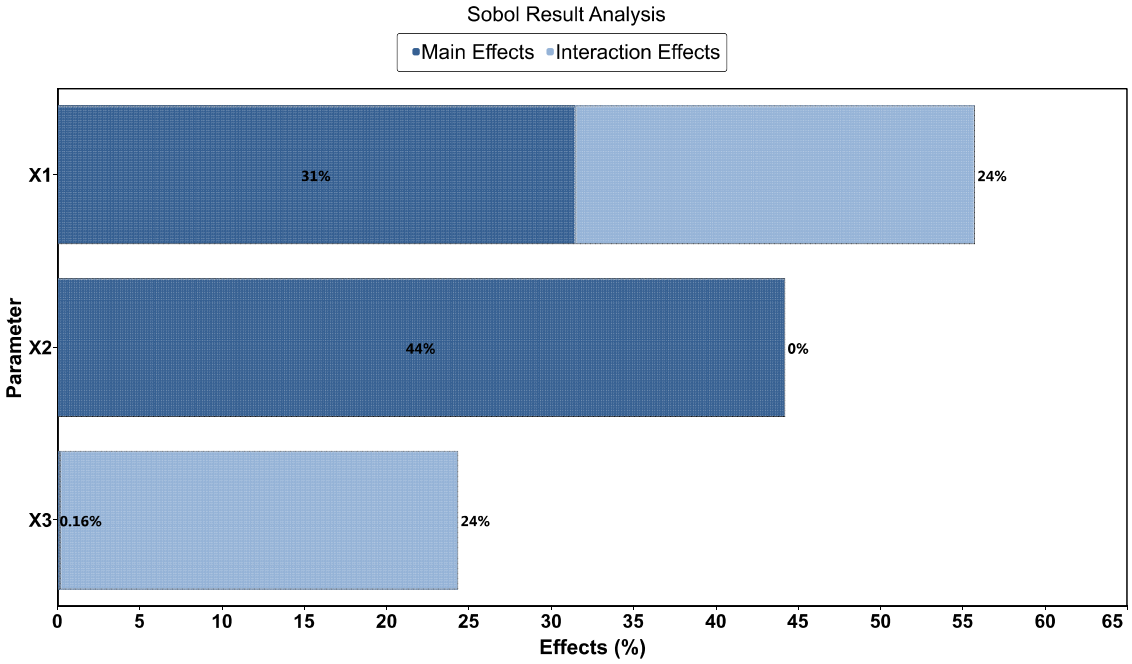


Figure 3.12: Sobol analysis results for Ishigami function

While using the Morris elementary effects method in the previous chapter we were not able to fully capture the complex sensitivities of the Ishigami function, we can exactly match the true values with the more sophisticated variance based Sobol sensitivity analysis method. Since we are dealing with an analytical function in this case, we can afford the high cost of the computation for Sobol method easily here. That is not obviously the case for real reservoir simulation models, and we will use Sobol method in real reservoir modeling scenarios in the latter case studies of this chapter.

3.6.3 Sobol G function

Another test function that we tested with Morris method was the Sobol G function. Using the same values for the a_i coefficients of this function (Table 3.8), we perform the Sobol GSA analysis to evaluate the results.

$$f(X) = \prod_{i=1}^{d} \frac{|4X_i - 2| + a_i}{1 + a_i} \quad (3.31)$$

Table 3.8: a_i values for Sobol G function

a_i	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}
Value	0.001	89.9	5.54	42.10	0.78	1.26	0.04	0.79	74.51	4.32	82.51	41.62

Once again, since the computational cost of evaluating this analytical function is minimal we generate the low-discrepancy sequence for 8192 sample points in the parameter space. Figures 3.13 and 3.14 that follow display the sampling sequence for Sobol G function parameters. It can be observed how the low-discrepancy sequence provides a space filling design which is necessary for the accuracy of the estimations.

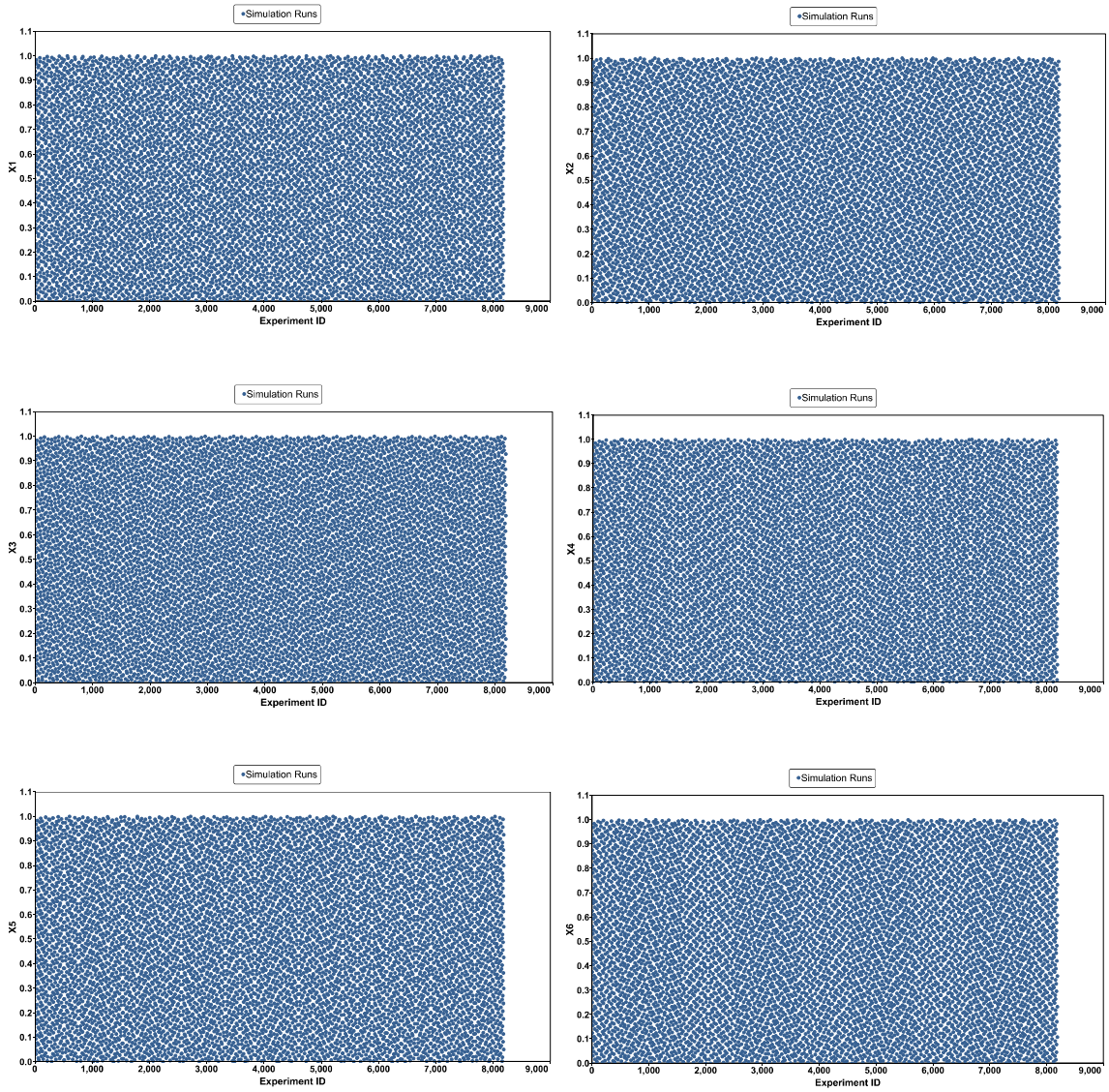


Figure 3.13: Run progress for input parameters X_1 to X_6

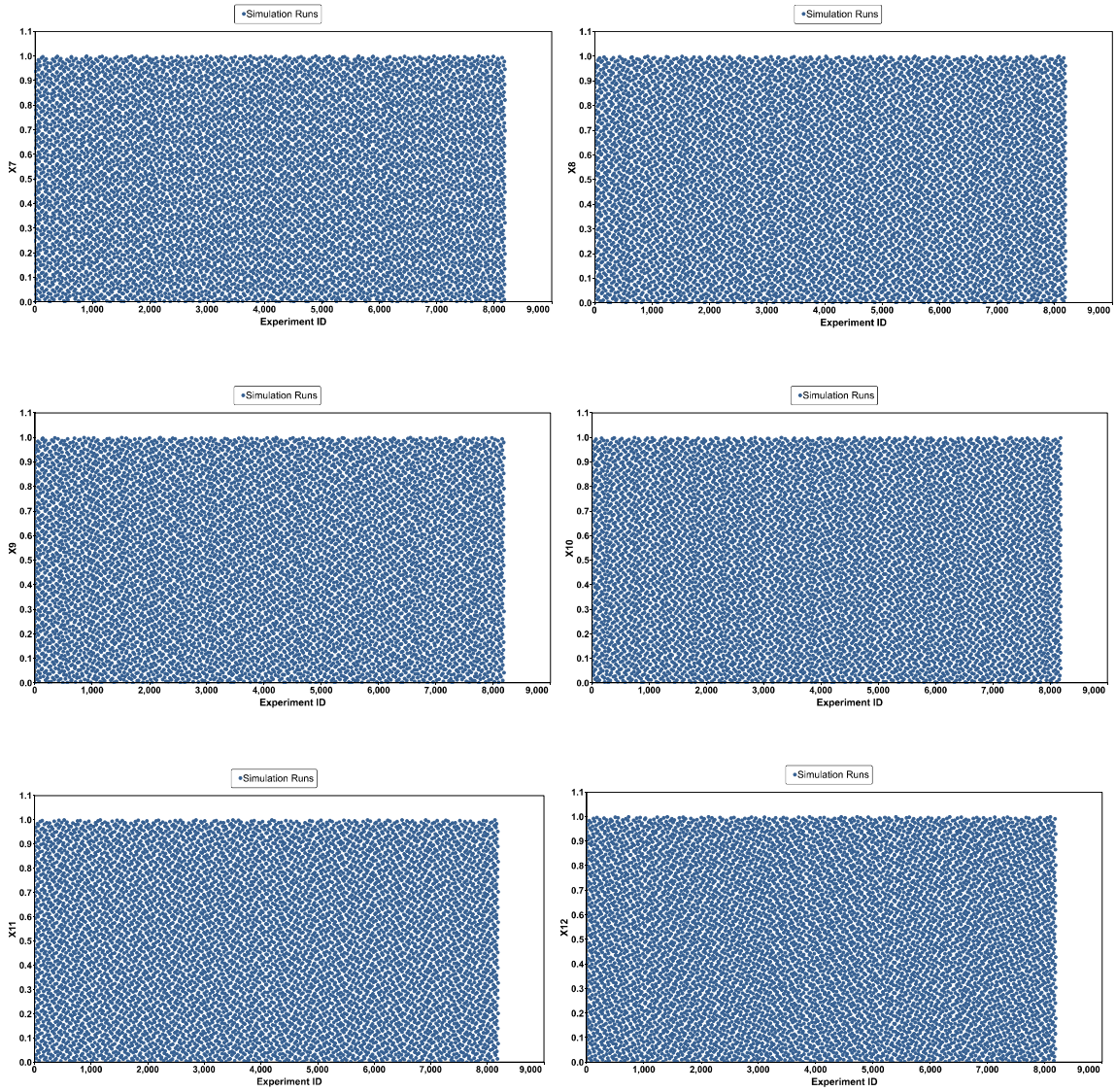


Figure 3.14: Run progress for input parameters X_7 to X_{12}

The comparison of orthogonality and minimum sample distance values between the used low-discrepancy sequence and LHD are summarized in the Table 3.9 below.

Table 3.9: Comparison of low-discrepancy sequence and Latin Hypercube design quality

	Orthogonality value	**Minimum sample distance**
LHD	0.029169	0.23512
Low-discrepancy sequence	5.81E-5	0.25249

Again, we can observe the low-discrepancy sequence outperforming LHD in terms of orthogonality. Next step, the outputs of the Sobol G function with respect to each input parameter is observed in Figures 3.15 to 3.18. In Figure 3.15, comparing parameters X_1 and X_2 for example show how the output is influenced by the choice of parameter value.

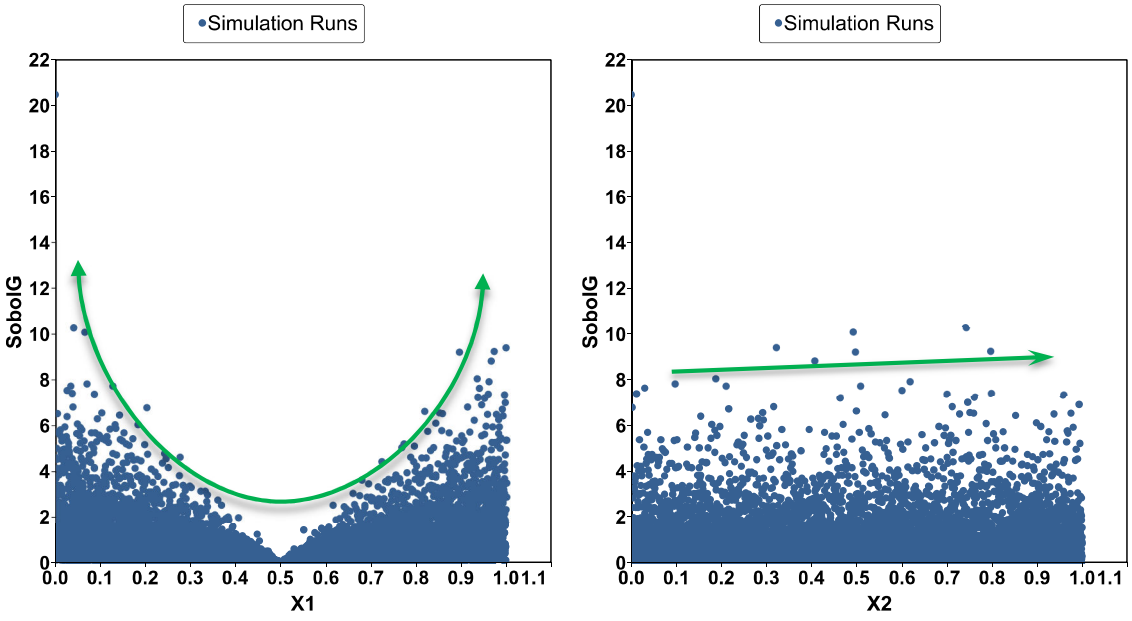


Figure 3.15: cross plot of Sobol G function vs. input parameters X_1 and X_2

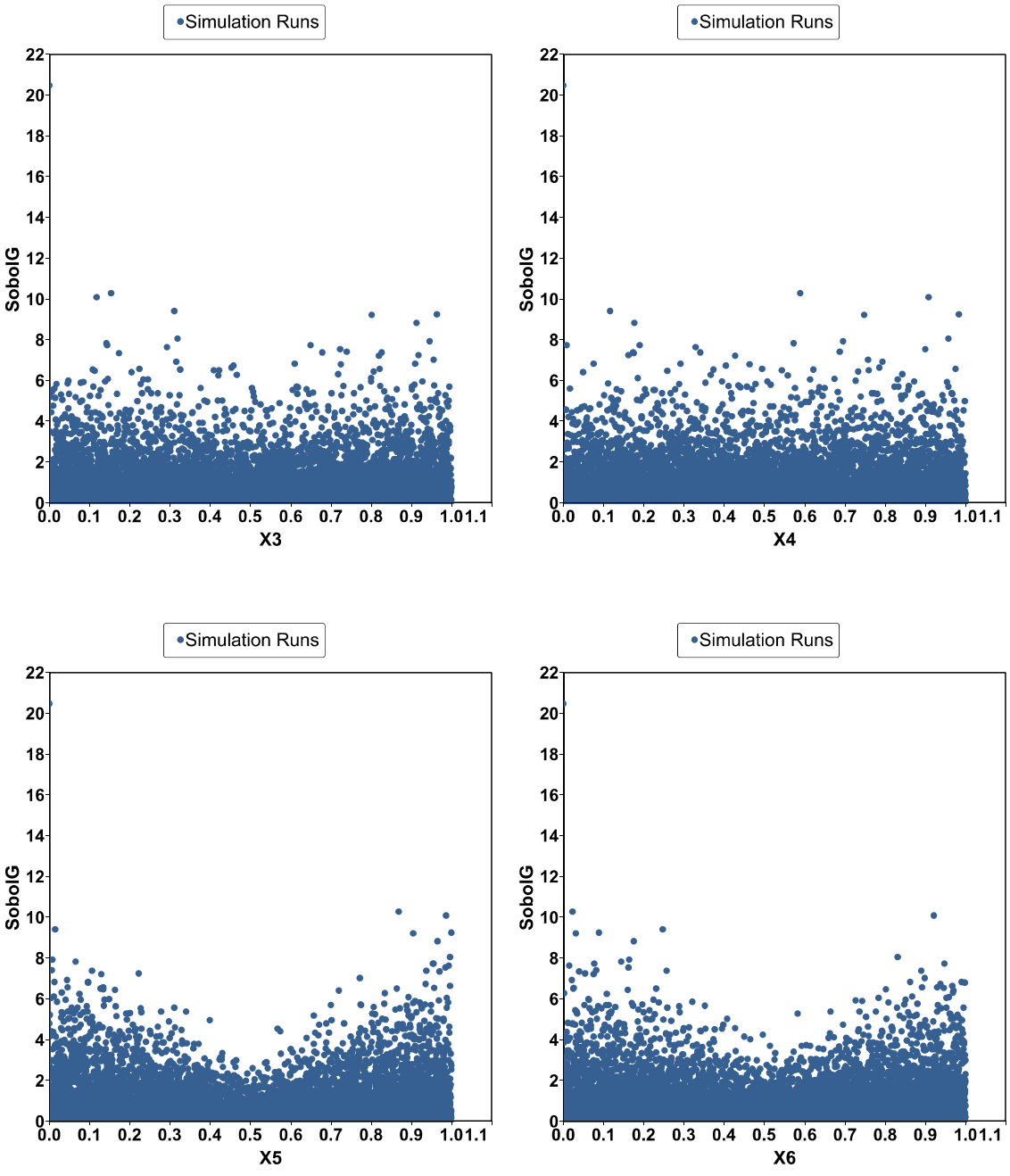


Figure 3.16: cross plot of Sobol G function vs. input parameters X_3 to X_6

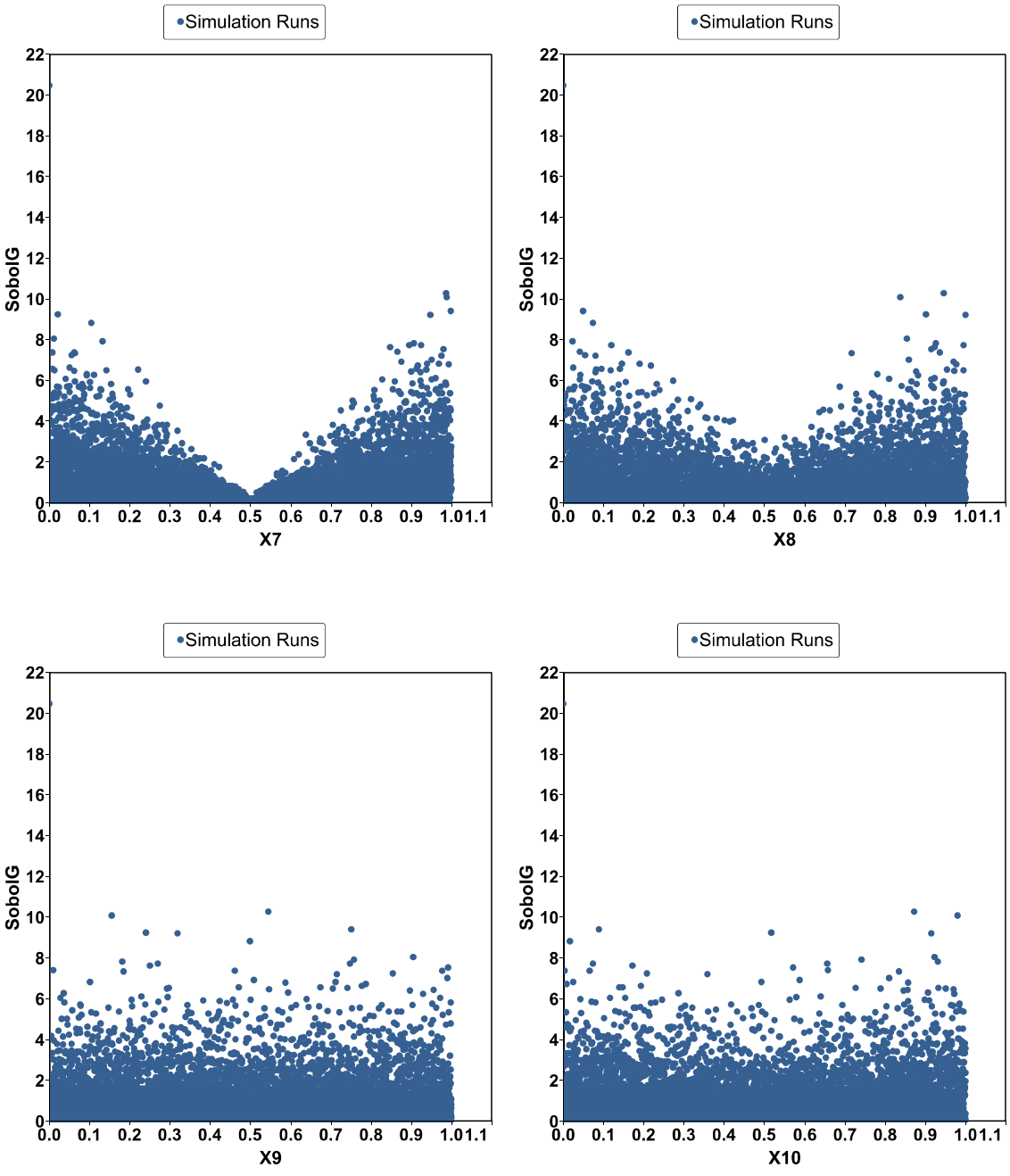


Figure 3.17: cross plot of Sobol G function vs. input parameters X_7 to X_{10}

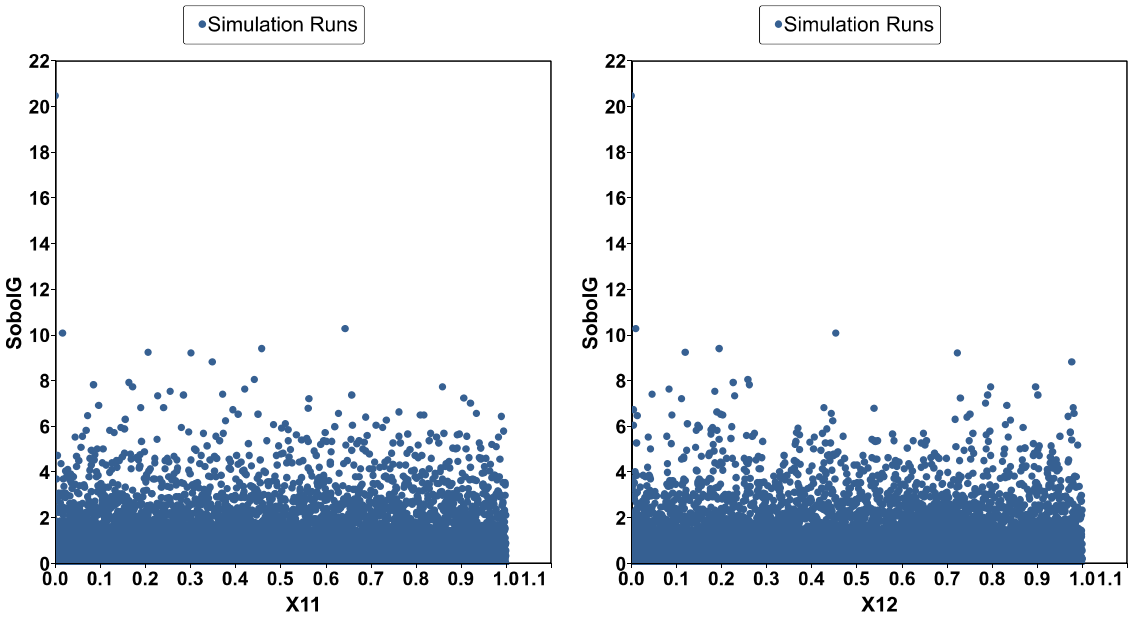


Figure 3.18: cross plot of Sobol G function vs. input parameters X_{11} and X_{12}

Overall, by analyzing the plots above, a strong correlation between the output of the function and parameters X_1 and X_7 is observed. This is aligned with what we expect. Calculating the main effect, total effect and interaction sensitivity indexes gives us the full picture in Table 3.10.

Table 3.10: Main, Interaction, and Total effects for Sobol G function

Parameter Name	Main Effects	Interaction Effects	Total Effects
X1	0.272057	0.162756	0.434812
X7	0.255369	0.133925	0.389294
X8	0.102942	0.0592103	0.162153
X5	0.102906	0.036165	0.139071

X6	0.0731247	0.031461	0.104586
X10	0.0347487	0	0.0347487
X3	0.032368	0	0.032368
X12	0.026683	0	0.026683
X2	0.0264884	0	0.0264884
X4	0.0264334	0	0.0264334
X11	0.0264271	0	0.0264271
X9	0.0264234	0	0.0264234

Plotting the result using stacked bars in Figure 3.19 gives us a quick and an easy to interpret visualization of parameter sensitivities.

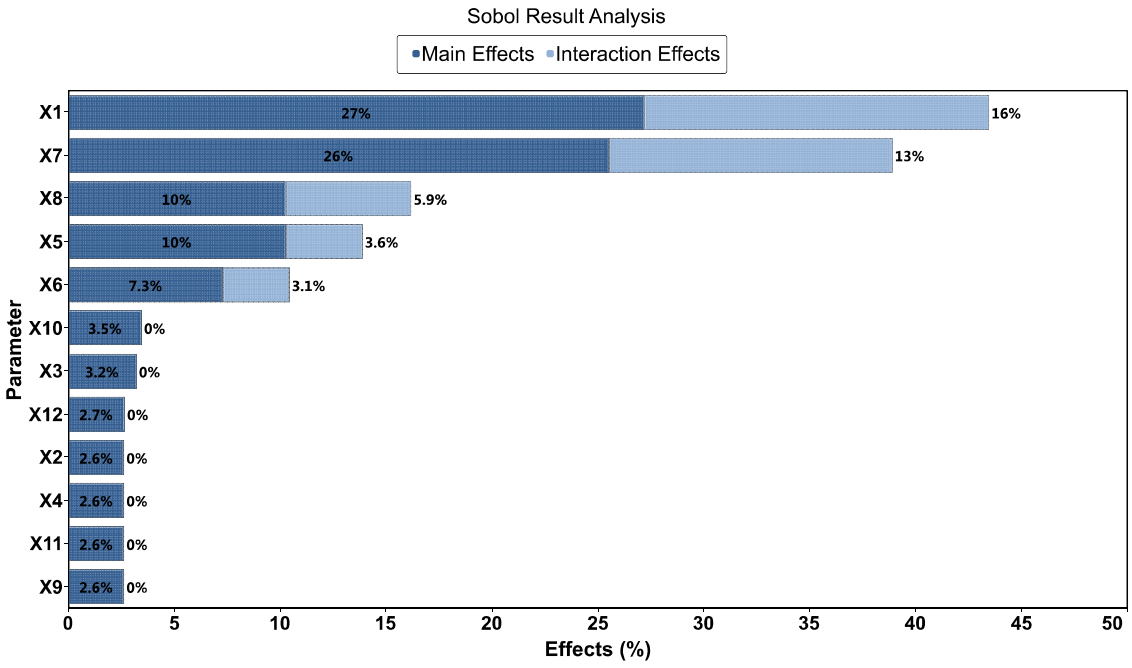


Figure 3.19: Sobol analysis results for Sobol G function

The results are as expected and validate the methodology used.

3.6.4 CSS pilot

The same reservoir from chapter 2 is analyzed using the developed Sobol method and results are validated against the Morris results. Here, unlike the previous test functions where we could directly evaluate the response from the function, we cannot afford the computational cost of running as many simulations. Therefore, first we construct the previously introduced surrogate model from the numerical simulation data and subsequently use that model as our proxy for further analysis.

Here we do a preliminary step before starting to create the low-discrepancy sequence for sensitivity analysis. Instead of trying to simulate the 8192 experiments which would be very time consuming and CPU exhausting, we create a Latin Hypercube design of 642 experiments and run it with the simulator. These 642 simulations would be used to train our proxy models.

We need to be careful about the fact our proxy model acts as an approximation of the real reservoir model, and its prediction error has to be taken into account. Hence, the accuracy of the prediction has to be assessed very carefully before continuing with further SA evaluations. In order to do so, we further create another LHD sampling of 64 simulations, but we will not use these simulations as part of the training data experiments for the surrogate model. Instead we use these 64 experiments as verification points to test the prediction capability of the trained surrogate models and make sure the proxy is an accurate representation of our real model.

3.6.4.1 RBF Proxy

For each of three objective functions in this case study, we first create the RBF Neural Network proxy and assess its prediction accuracy using the following plots. In these plots, the blue points are the data points used for training the proxy model, and the green points are the verification points. We are plotting the proxy predicted values of our objective functions on the X axis while the true (simulated) values from our finite-difference simulator are plotted on the Y axis. Therefore, the closer the points are to the 45-degree line, the more accurate our proxy model is. We also check the R^2 value of the training and verification points. Again, the closer the values are to 1, the better our proxy is deemed in terms of its prediction capability. Note that RBF neural network always goes through all the training points so the R^2 value for RBF training is always equal to 1. The proxy QC plots for our objective functions Cumulative Oil Production, Cumulative Water Production, and Cumulative Steam Injection are as follows in Figures 3.20, 3.21 and 3.22:

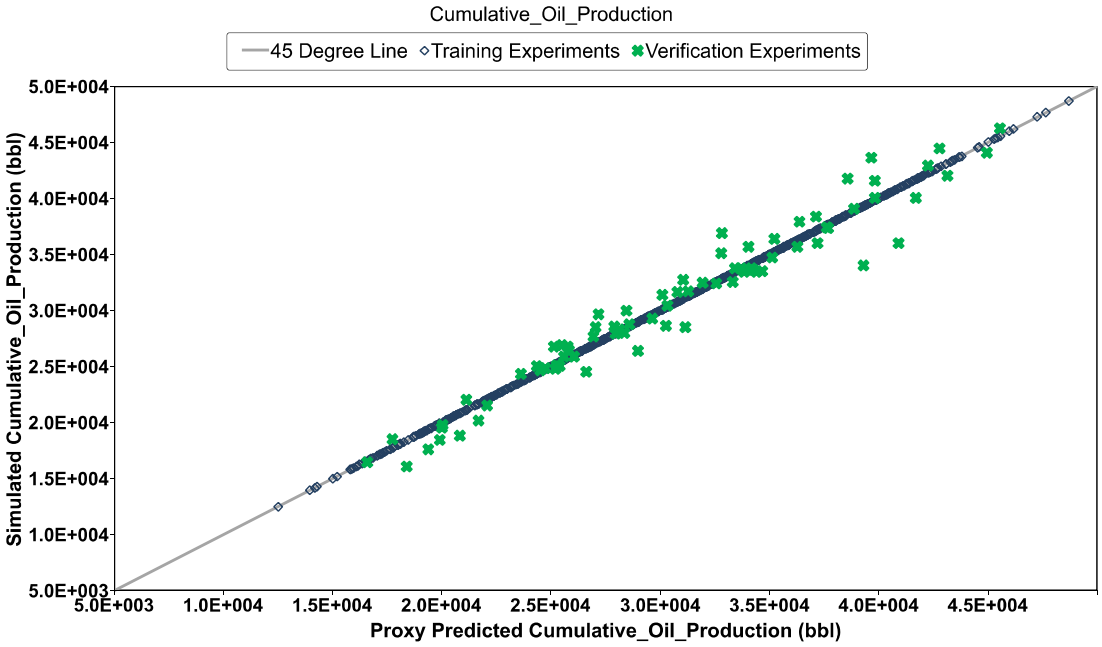


Figure 3.20: RBF proxy quality control plot for Cumulative Oil Production objective function (R^2-training=1.000, R^2-verification=0.953)

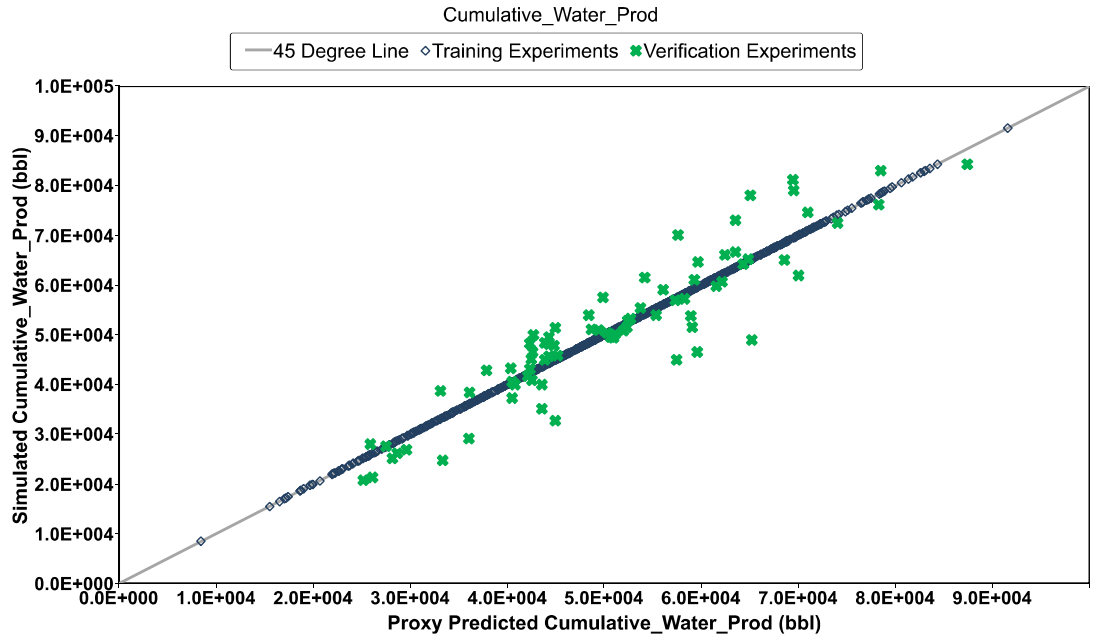


Figure 3.21: RBF proxy quality control plot for Cumulative Water Production objective function (R^2-training=1.000, R^2-verification=0.860)

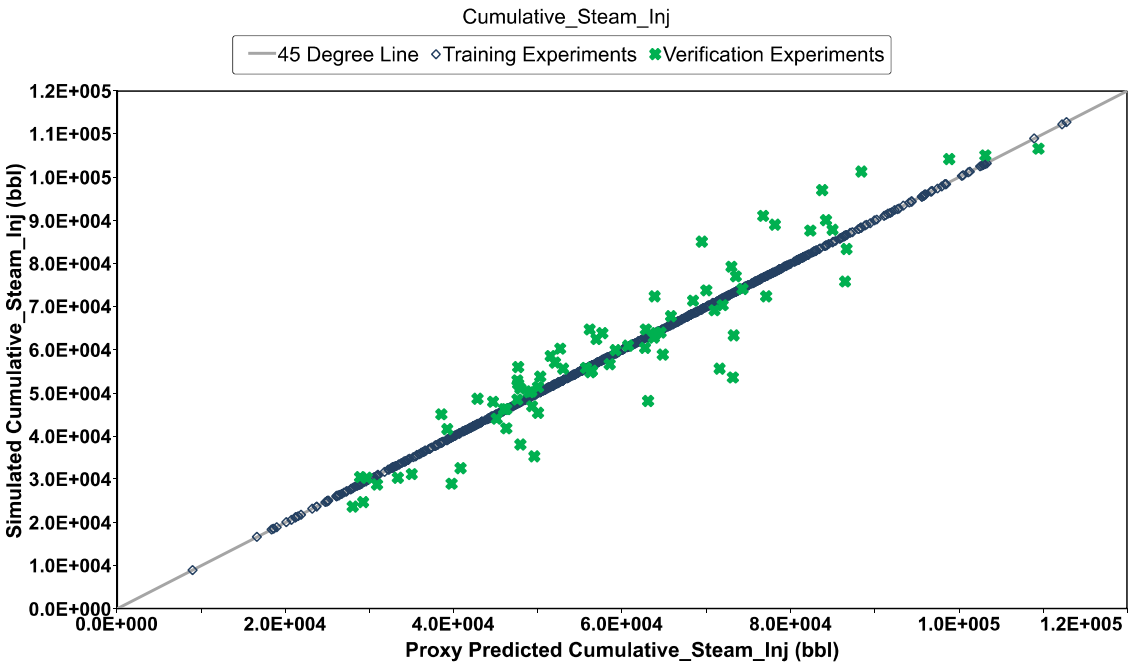


Figure 3.22: RBF proxy quality control plot for Cumulative Steam Injection objective function (R^2-training=1.000, R^2-verification=0.886)

The R^2 values for the training and verification points per objective function are calculated and shown in Table 3.11:

Table 3.11: R^2 values for training and verification experiments of RBF proxy per objective function

Objective Function	R^2-training	R^2-verification
Cumulative Oil Production	1	0.953
Cumulative Water Production	1	0.860
Cumulative Steam Injected	1	0.886

As we can see the RBF neural network surrogate model demonstrates reliable prediction capabilities for this model and is a low computational cost alternative to the simulation. Next using the already simulated 642 experiments from before we build the multilayer neural network model and compare the accuracy with the same 64 verification points. As mentioned, the main advantage of Multilayer NN over RBF in our case is configurability of the architecture. We tried several different configurations and used a network of 4 hidden layers, with 4 nodes in the 1st layer, 2 nodes in the 2nd and 3rd layers, and 4 nodes in the final 4th layer (4,2,2,4 architecture). The QC plots are presented in Figures 3.23 to 3.25.

3.6.4.2 Multi-Layer NN (with 4,2,2,4 architecture)

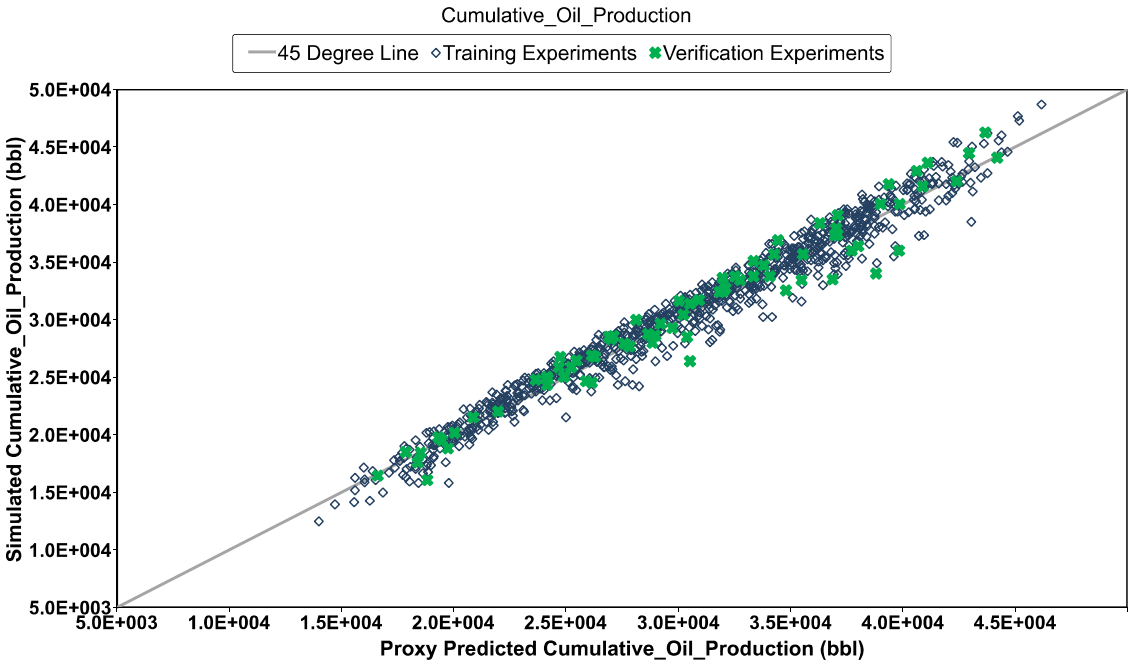


Figure 3.23: Multi-Layer NN proxy quality control plot for Cumulative Oil Production objective function (R^2-training=0.934, R^2-verification=0.957)

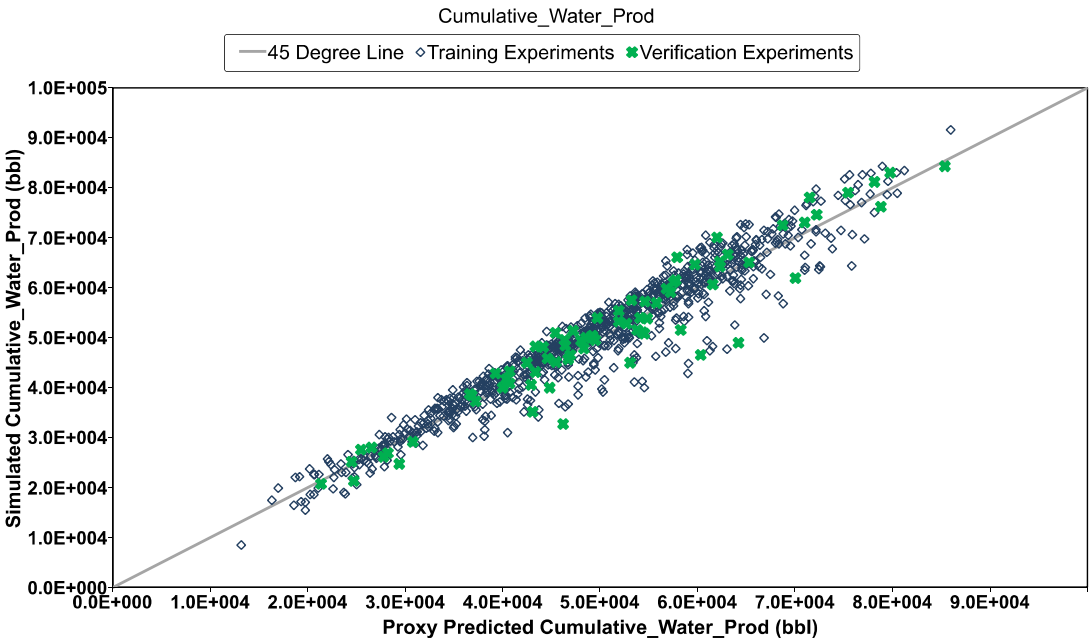


Figure 3.24: Multi-Layer NN proxy quality control plot for Cumulative Water Production objective function (R^2-training=0.848, R^2-verification=0.914)

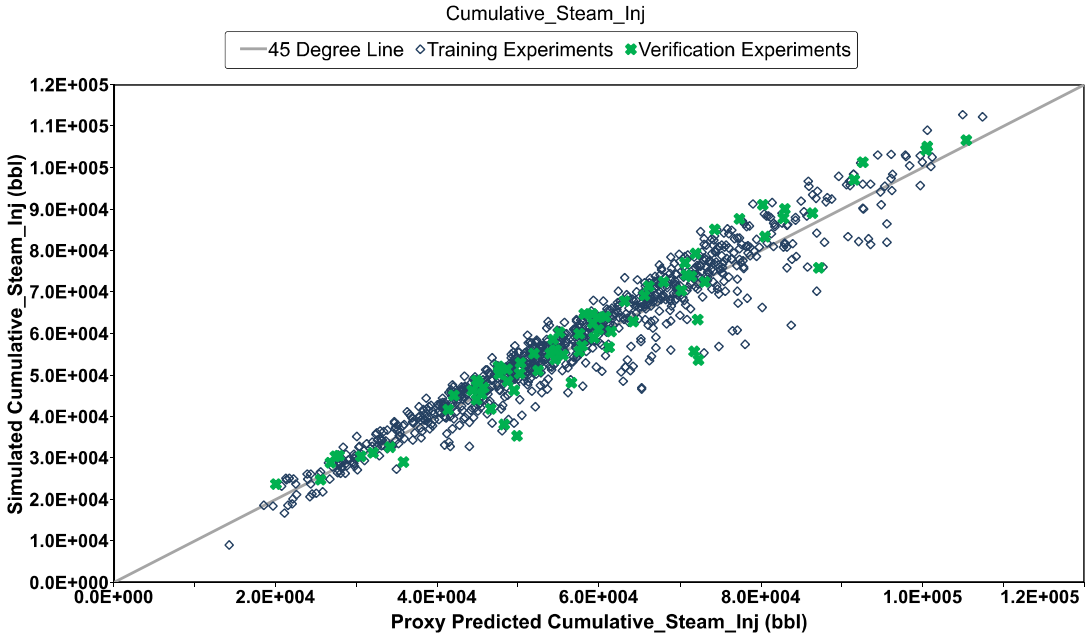


Figure 3.25: Multi-Layer NN proxy quality control plot for Cumulative Steam Injection objective function (R^2-training=0.872, R^2-verification=0.924)

We can observe that the verification points for all three objective functions are very close to the 45-degree line in our QC plots, therefore the multilayer NN proxy model exhibits good prediction capabilities. Comparing the R^2 values for the verification points in Table 3.12 with the RBF proxy in Table 3.11 shows an overall improvement in accuracy, but both surrogate models are reliable for the purposes of this particular case study. This might not be the case in other reservoir models and the flexibility of the configuration in Multilayer NN could potentially be a great advantage.

Table 3.12: R^2 values for training and verification experiments of Multi-Layer NN proxy per objective function

Objective Function	R^2-training	R^2-verification
Cumulative Oil Production	0.934	0.957
Cumulative Water Production	0.848	0.914
Cumulative Steam Injected	0.872	0.924

Before we continue to generate the low-discrepancy sequence from the input parameters and evaluate the objective functions using the proxy model (we use the Multi-Layer NN), we also take a look at the distribution of values for each objective function with all uncertain parameters sampled from the prior probability density functions. The plots show a histogram of objective function values, to illustrate the shape of the probability density function, as well as the cumulative probability. P10, P50, and P90 values are also highlighted. We use the Monte Carlo simulation size of 65000. This is again feasible only

using the proxy models, as this number of simulations for a reservoir model is practically impossible. The probability distributions and cumulative probabilities of the three objective functions are displayed in Figures 3.26, 3.27, and 3.28 below.

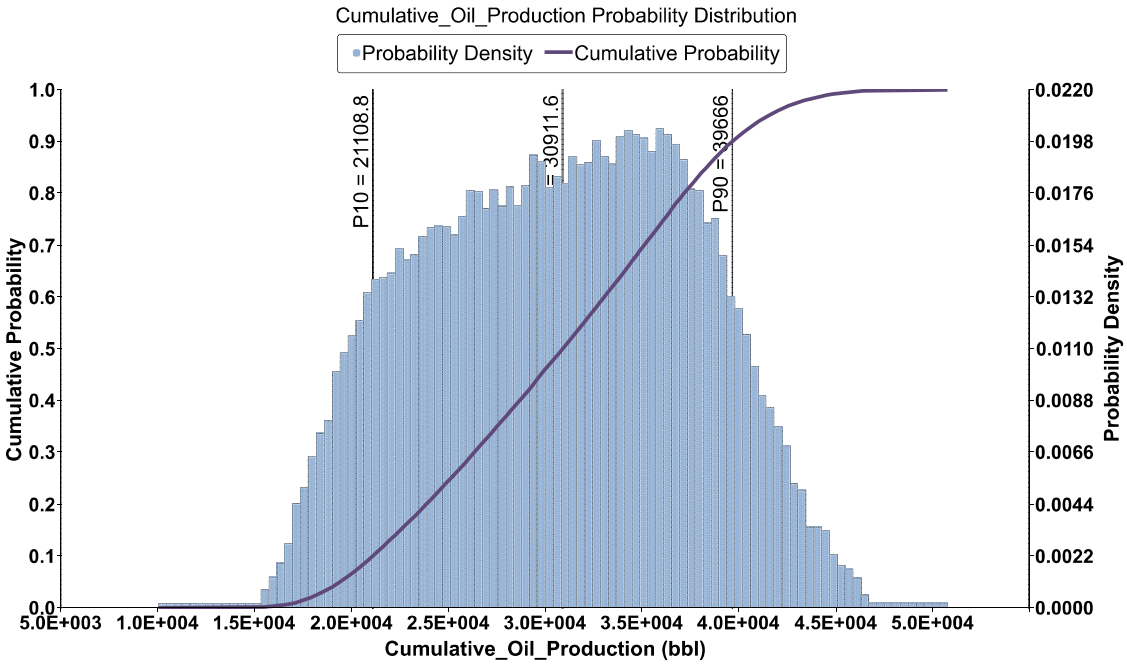


Figure 3.26: Probability distribution and cumulative probability of Cumulative Oil Production objective function

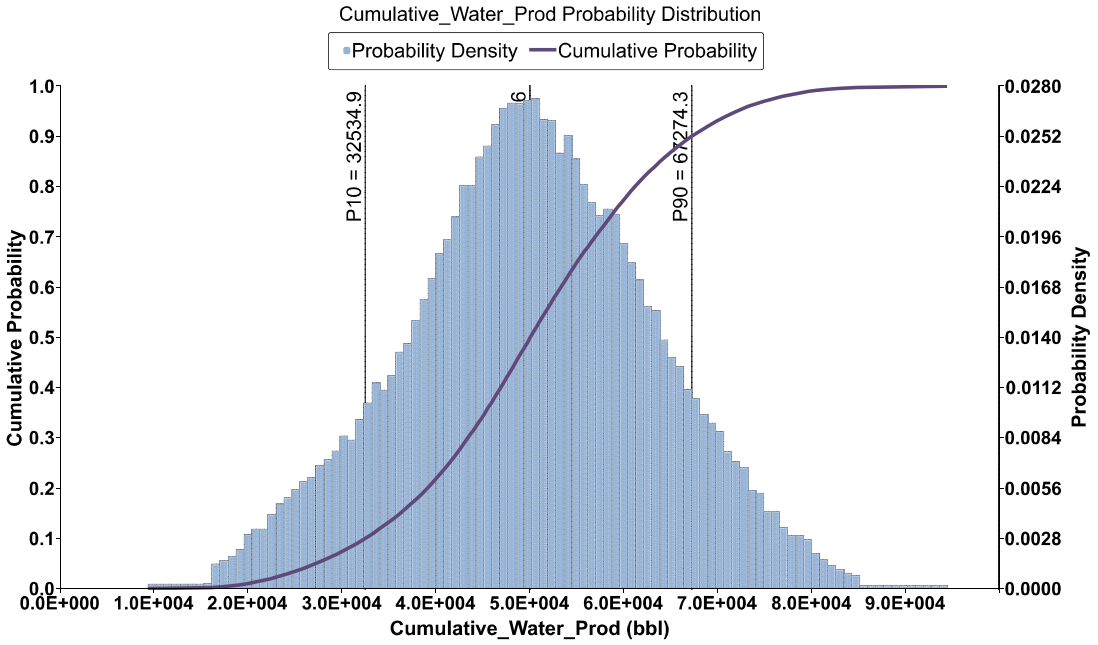


Figure 3.27: Probability distribution and cumulative probability of Cumulative Water Production objective function

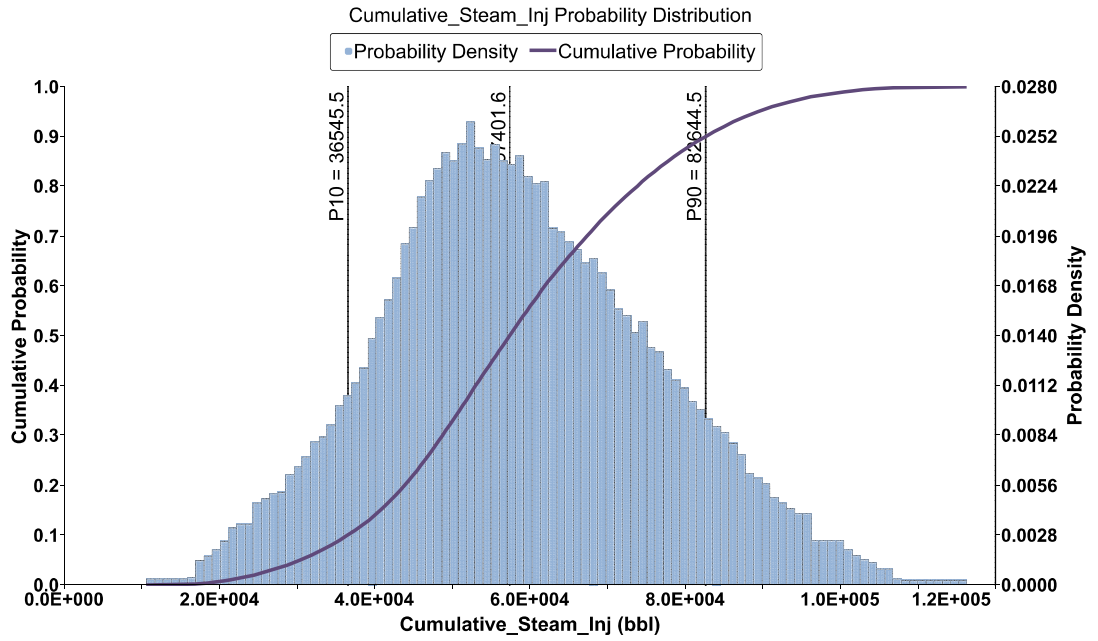


Figure 3.28: Probability distribution and cumulative probability of Cumulative Steam Injection objective function

Now that a reliable proxy model is available to use to approximate our reservoir model, we construct a low-discrepancy sequence of 8192 experiments and output the predictions of our surrogate model to calculate the sensitivity indexes for each objective function.

The scatter matrix plot of our experiment design and objective functions are displayed in Figure 3.29. Scatter matrix plots can be used to identify and investigate the nature of pairwise relationships between variables, and identify clustering of data into groups and data outliers

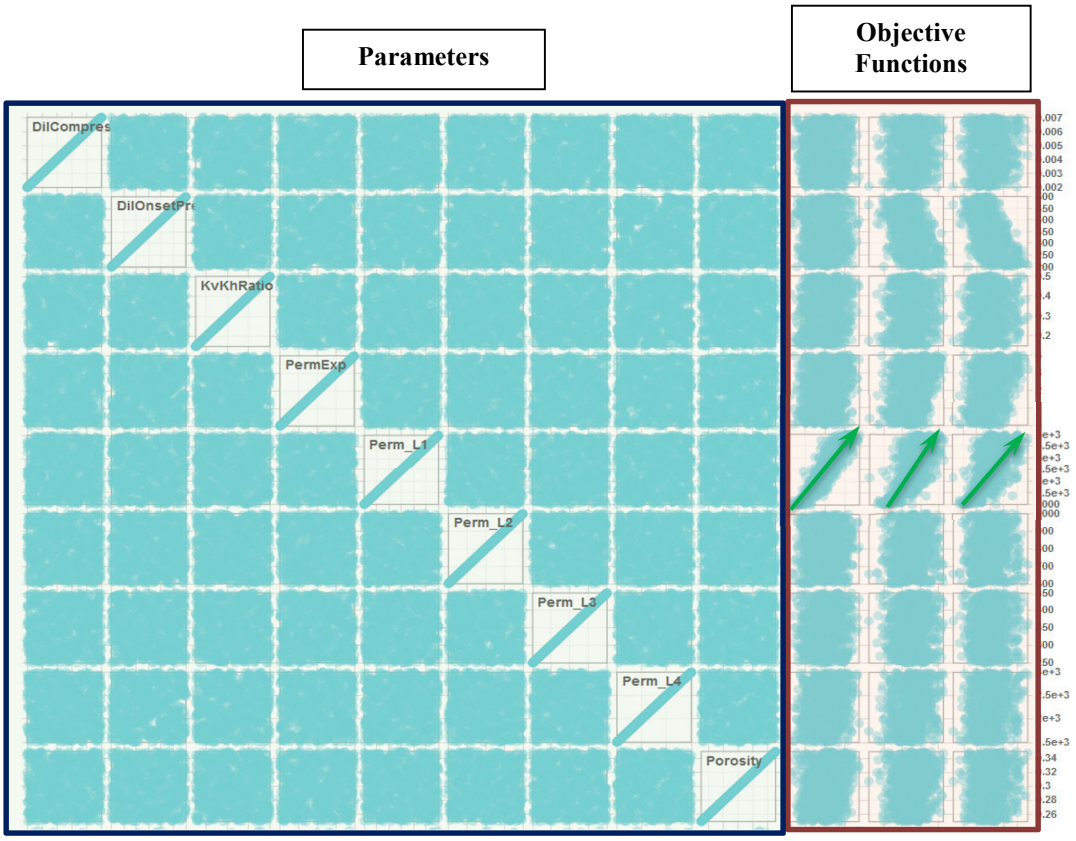


Figure 3.29: Scatter plot matrix for input parameters and objective functions of CSS pilot case study

Figure 3.29 is a great way to roughly determine if we have a linear correlation between parameters and objective functions. Here for example, we can see a linear correlation between parameter Perm_L1 (Permeability in Layer 1) with all 3 objective functions. We will confirm the influence of this parameter once the Sobol sensitivity indexes are calculated.

The global sensitivity approach based on Sobol's method using the Quasi Monte Carlo sequence and Multi-Layer NN surrogate is used to evaluate the uncertainty in the parameter space analysis. The results as shown in the following Figure 3.30 and Table 3.13 are in agreement with the previous conclusion drawn from Morris method.

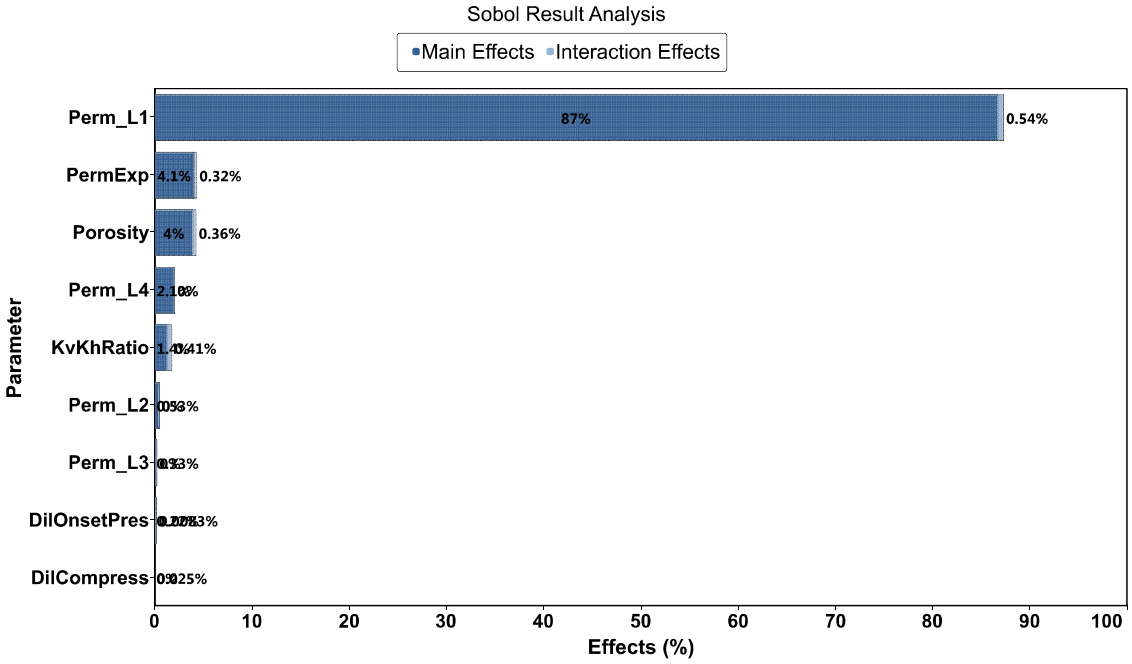


Figure 3.30: Sobol analysis results for Cumulative Oil Production objective function

Table 3.13: Main, Interaction, and Total effects for Cumulative Oil Production objective function

Parameter Name	Main Effects	Interaction Effects	Total Effects
Perm_L1	0.867881	0.00537187	0.873253
PermExp	0.0407045	0.00322674	0.0439313
Porosity	0.04001	0.00361514	0.0436252
Perm_L4	0.0209635	0	0.0209635
KvKhRatio	0.0138181	0.00406075	0.0178789
Perm_L2	0.00534442	0	0.00534442
Perm_L3	0.00328453	0	0.00328453
DilOnsetPres	0.00222106	8.29E-05	0.00230397
DilCompress	0.000246557	0	0.000246557

Permeability in Layer 1 is the most influential parameter for Cumulative Oil Production Objective Function in this case. The cross plot for Perm_L1 versus the objective function demonstrates a linear positive relationship between this parameter and the objective function.

An additional and valuable information that the Sobol approach provides is the quantifications that exactly associate parameters to the amount of variability in the objective function that we could reduce, if we could know the true value of the parameters. As an example, by examining Figure 3.30 and Table 3.13, we can understand that 85% of the variability of our objective function, Cumulative Oil Production, is coming from Permeability in Layer 1. We already could tell that this is our most important factor from

Morris analysis, but Morris lacks the ability to quantify this contribution. The same is true for the level of interaction present among parameters. Examining the cross plot of Permeability in Layer 1 vs. the Cumulative Oil Production objective function in Figure 3.31 shows the trend of the effect of this parameter on the output.

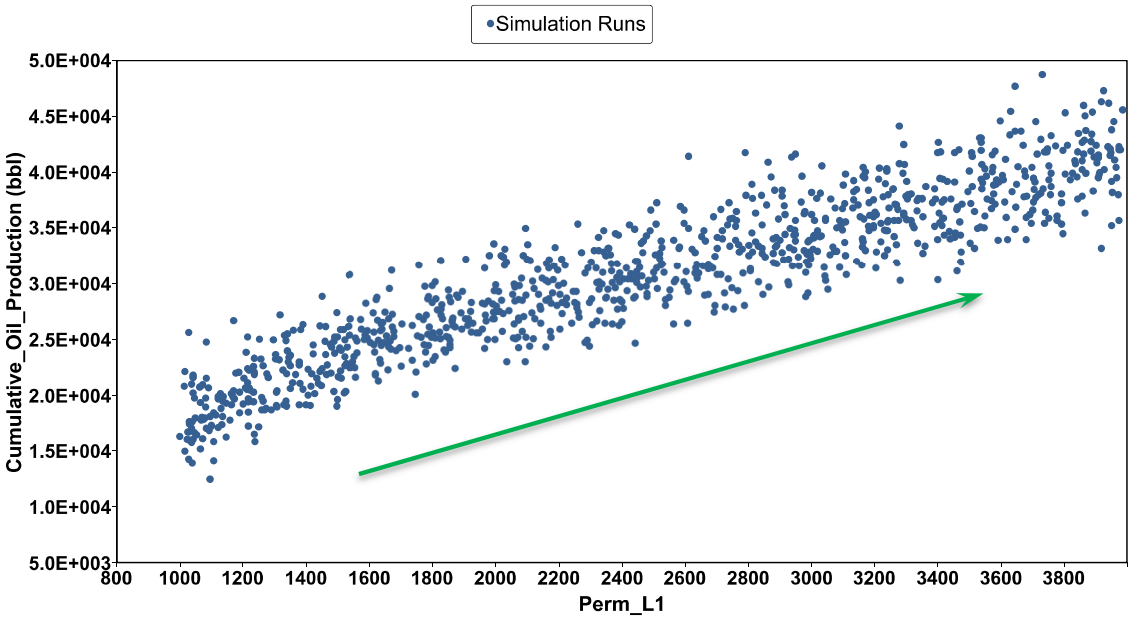


Figure 3.31: Cross plot of parameter 'Permeability in Layer 1' vs. Cumulative Oil Production objective function

The results for the next objective function are presented next in Figure 3.32 and Table 3.14:

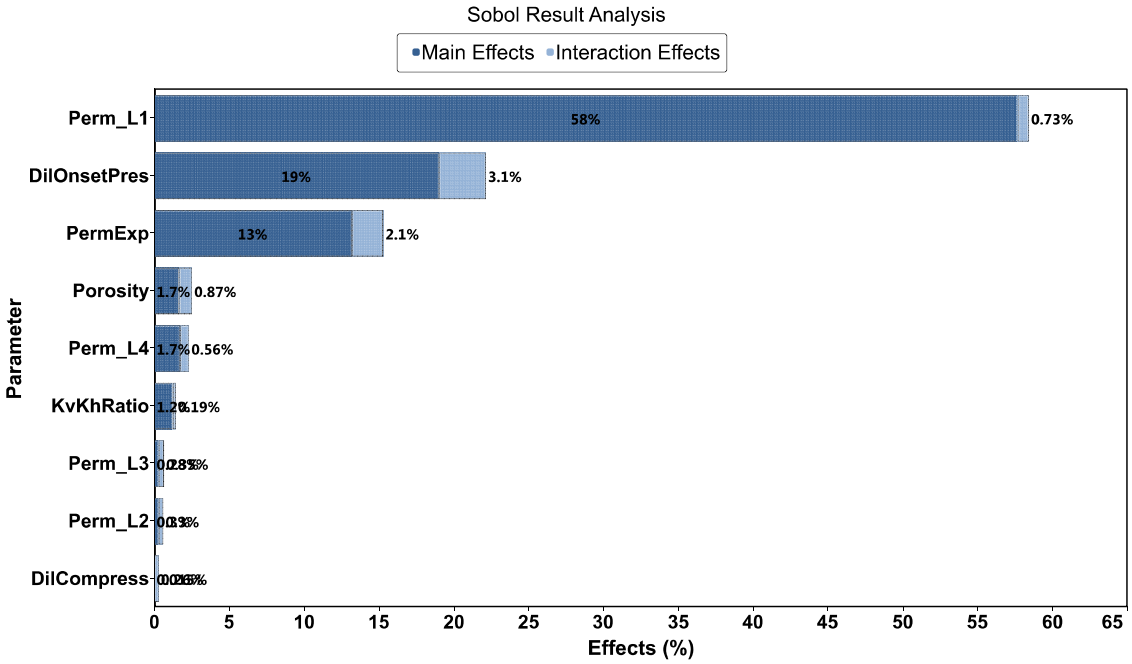


Figure 3.32: Sobol analysis results for Cumulative Water Production objective function

Table 3.14: Main, Interaction, and Total effects for Cumulative Water Production objective function

Parameter Name	Main Effects	Interaction Effects	Total Effects
Perm_L1	0.576922	0.00730719	0.58423
DilOnsetPres	0.190474	0.0307815	0.221255
PermExp	0.132176	0.0208567	0.153033
Porosity	0.0166608	0.00869757	0.0253583
Perm_L4	0.0174736	0.0055775	0.0230511
KvKhRatio	0.012497	0.00194677	0.0144438
Perm_L3	0.00281159	0.00351528	0.00632687
Perm_L2	0.00295319	0.00297259	0.00592578
DilCompress	0.000152179	0.00263893	0.00279111

Here the Permeability in Layer 1, Dilation Onset Pressure, and the Permeability exponential factor in the Carmen-Kozeny equation for permeability as function of porosity. Examining the cross plots (Figures 3.33 to 3.35) demonstrates the correlation between Cumulative Water Production and these three parameters.

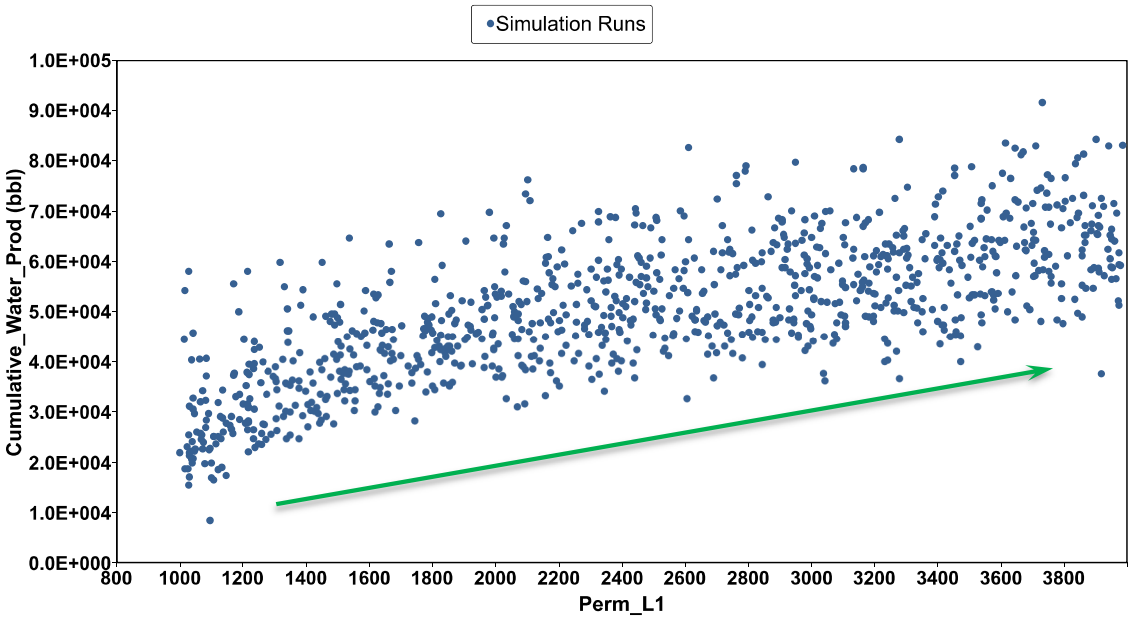


Figure 3.33: Cross plot of parameter 'Permeability in Layer 1' vs. Cumulative Water Production objective function

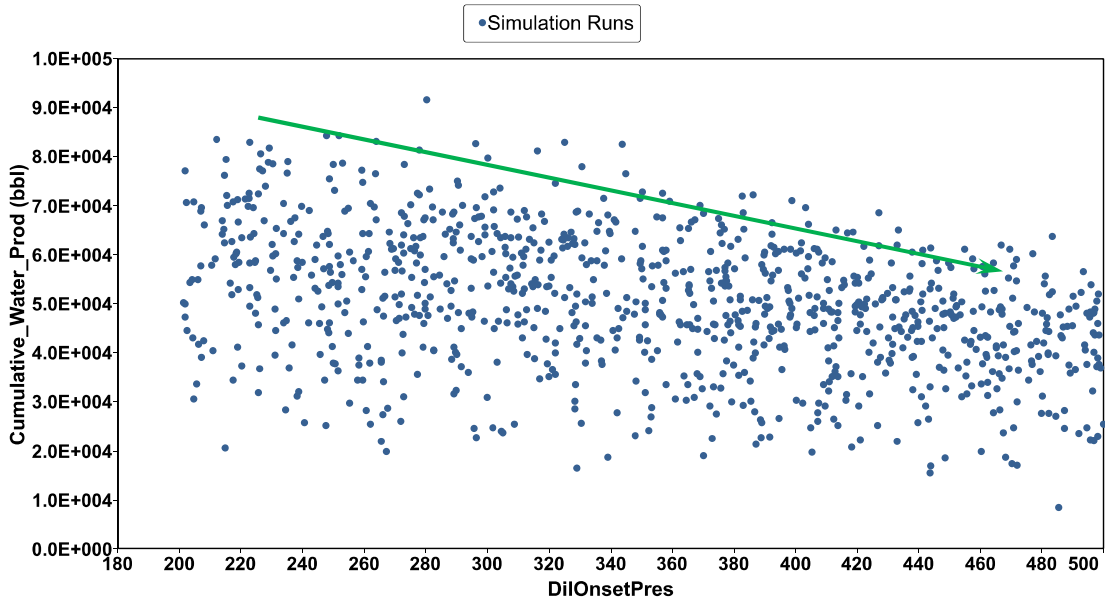


Figure 3.34: Cross plot of parameter 'Dilation Onset Pressure' vs. Cumulative Water Production objective function

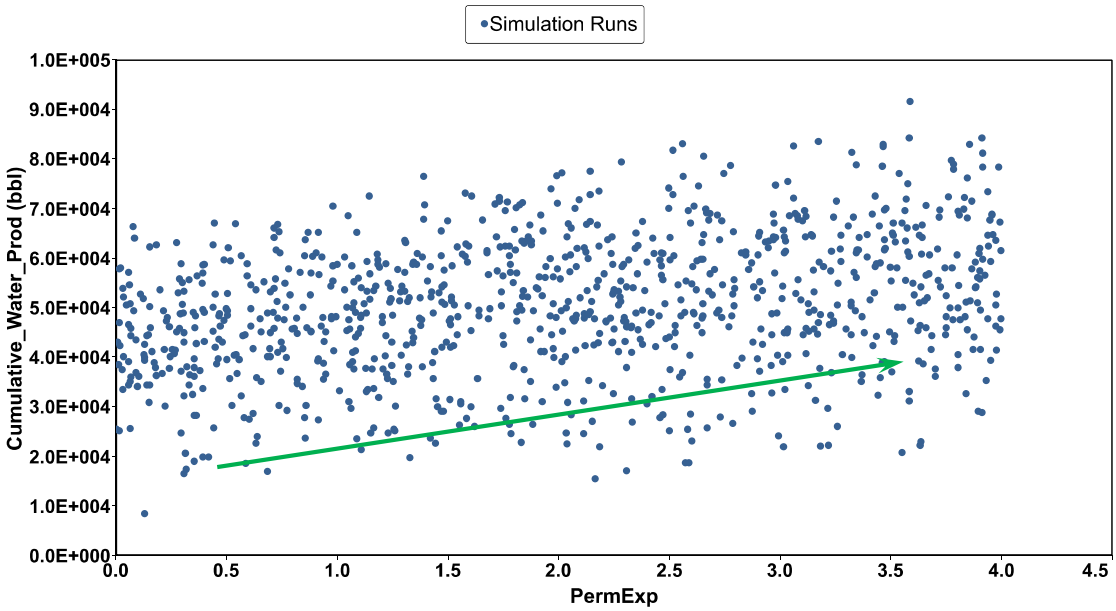


Figure 3.35: Cross plot of parameter 'Permeability Exponent' vs. Cumulative Water Production objective function

Cumulative Steam Injection:

The Sobol analysis results are displayed for the Cumulative Steam Injection objective function in Figure 3.36 and Table 3.15.

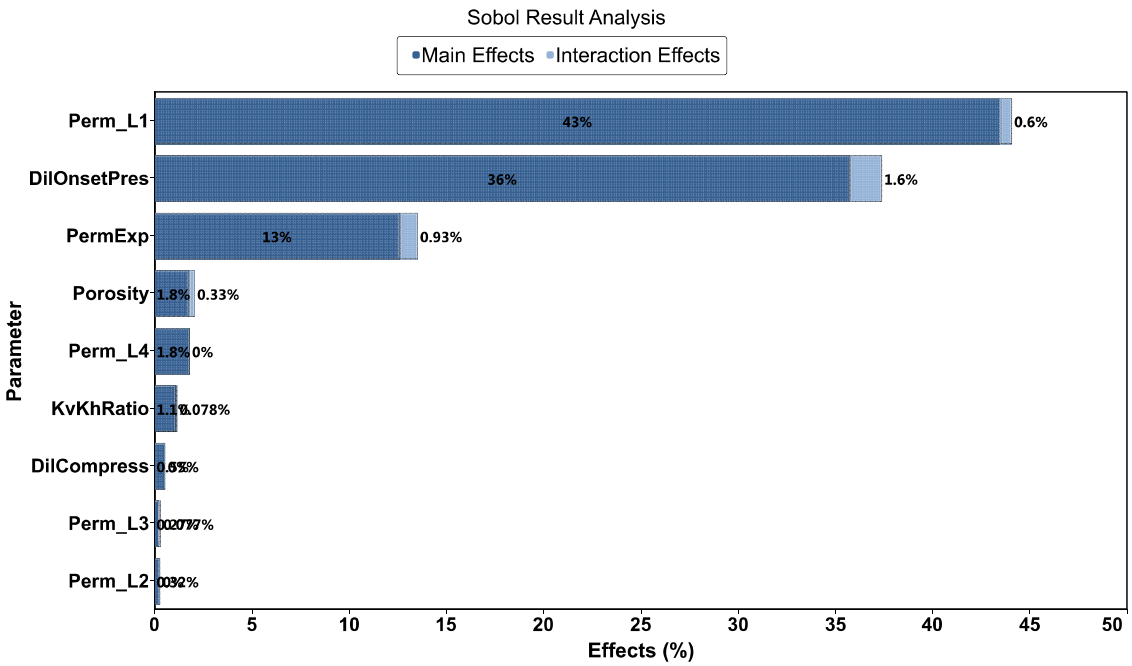


Figure 3.36: Sobol analysis results for Cumulative Steam Injection objective function

Table 3.15: Main, Interaction, and Total effects for Cumulative Steam Injection objective function

Parameter Name	Main Effects	Interaction Effects	Total Effects
Perm_L1	0.434971	0.00602888	0.441
DilOnsetPres	0.35787	0.0160079	0.373878
PermExp	0.126189	0.00932152	0.135511
Porosity	0.0177203	0.003277	0.0209973

Perm_L4	0.0181432	0	0.0181432
KvKhRatio	0.0110331	0.00077651	0.0118096
DilCompress	0.00550244	0	0.00550244
Perm_L3	0.00270592	0.000770401	0.00347633
Perm_L2	0.00318732	0	0.00318732

For the Cumulative Steam Injected objective function, the Permeability in Layer 1, Dilation Onset Pressure, and the Permeability exponential factor in the Carmen-Kozeny equation are also the most influential parameters. The parameter vs. objective function cross plots are plotted in Figures 3.37 to 3.39.

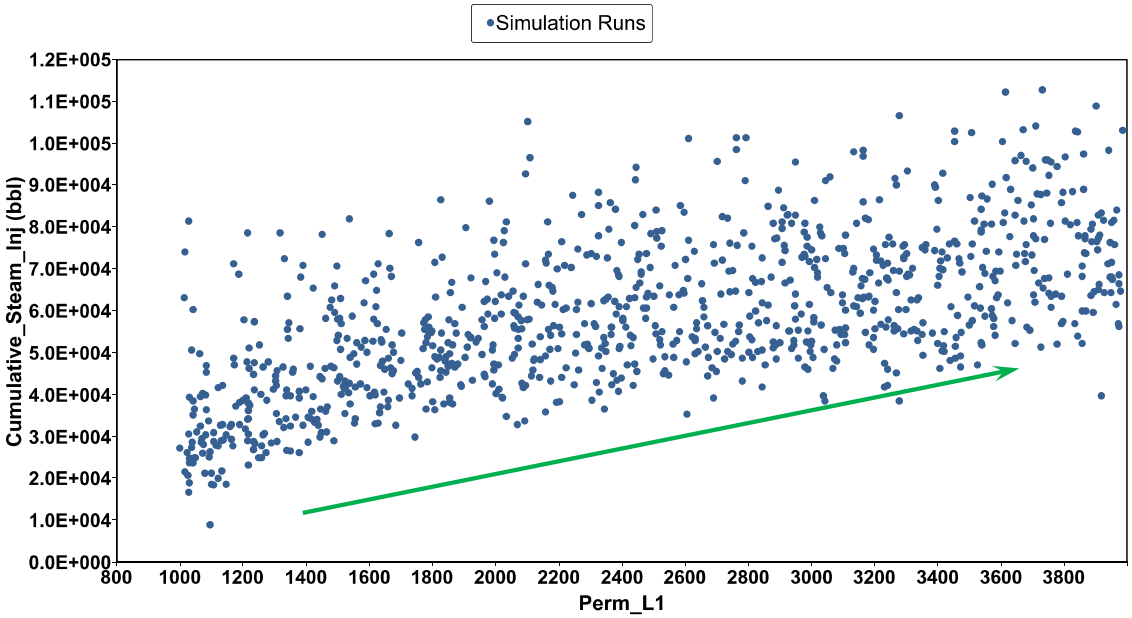


Figure 3.37: Cross plot of parameter 'Permeability in Layer 1' vs. Cumulative Steam Injection objective function

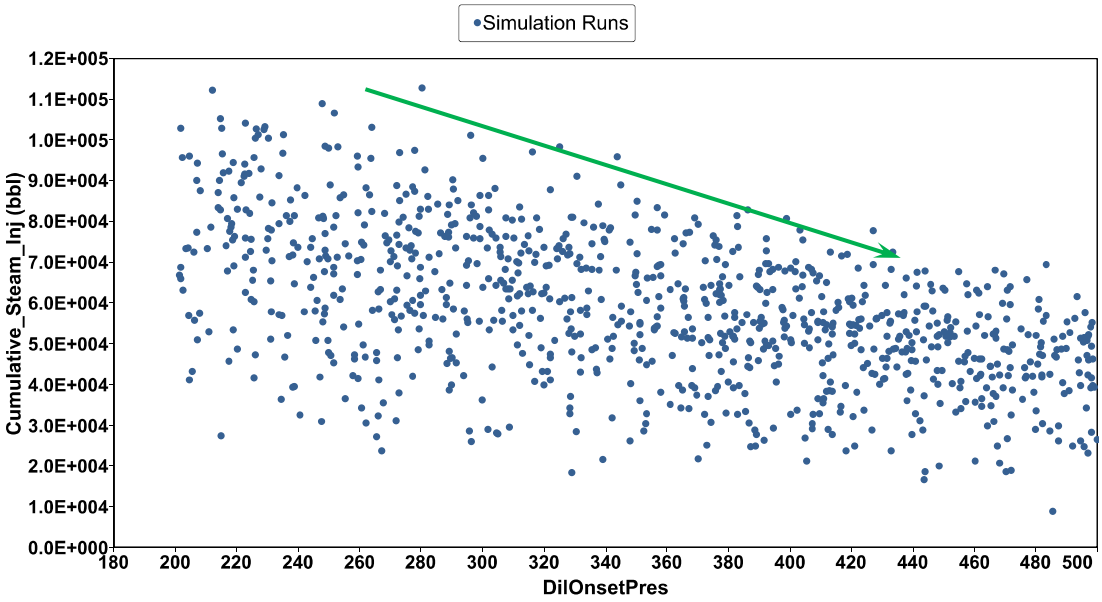


Figure 3.38: Cross plot of parameter 'Dilation Onset Pressure' vs. Cumulative Steam Injection objective function

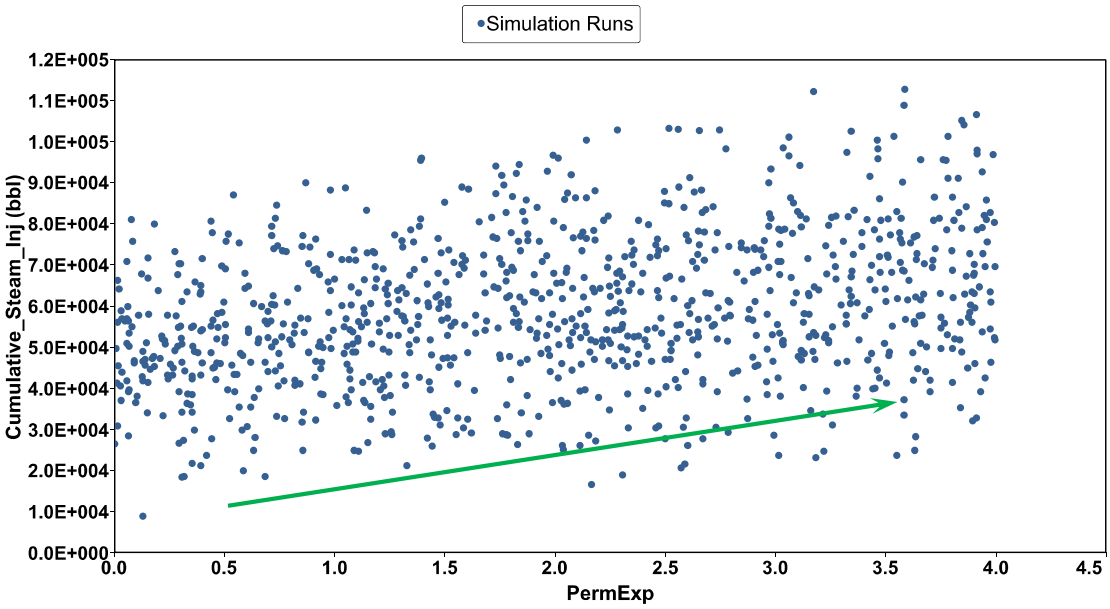


Figure 3.39: Cross plot of parameter 'Permeability Exponent' vs. Cumulative Steam Injection objective function

Applying Sobol global sensitivity analysis method for the cyclic steam stimulation pilot case study, we can conclude that parameter 5 (Permeability in Layer 1) has the most impact on the variability of all three objective functions, while parameter two, Dilation Onset Pressure, has a lot more contribution to the changes in Cumulative Water Production and Cumulative Steam Injected compared to Cumulative Oil Production. At the same time, all objective functions seem to be insensitive to Permeabilities in Layers 2 and 3, and Dilation Compressibility. Based on the numerical values obtained for the total and main effects of all parameters per objective function, it can be observed that interaction effects are not strongly present in our model.

3.6.5 Infill well drilling optimization

In this case study we will look at running a forecast along with the placement of 3 new infill wells (2 vertical injection wells & 1 horizontal producer well) in a conventional reservoir. The reservoir has been on primary production for 17 years and production rates have declined significantly. We would like to increase the oil recovery factor and maximize the Field NPV of this model using an optimization algorithm. Also, we would study the sensitivity of Field NPV to the variables of this model. The 3D representation of the reservoir is shown in Figure 3.40:

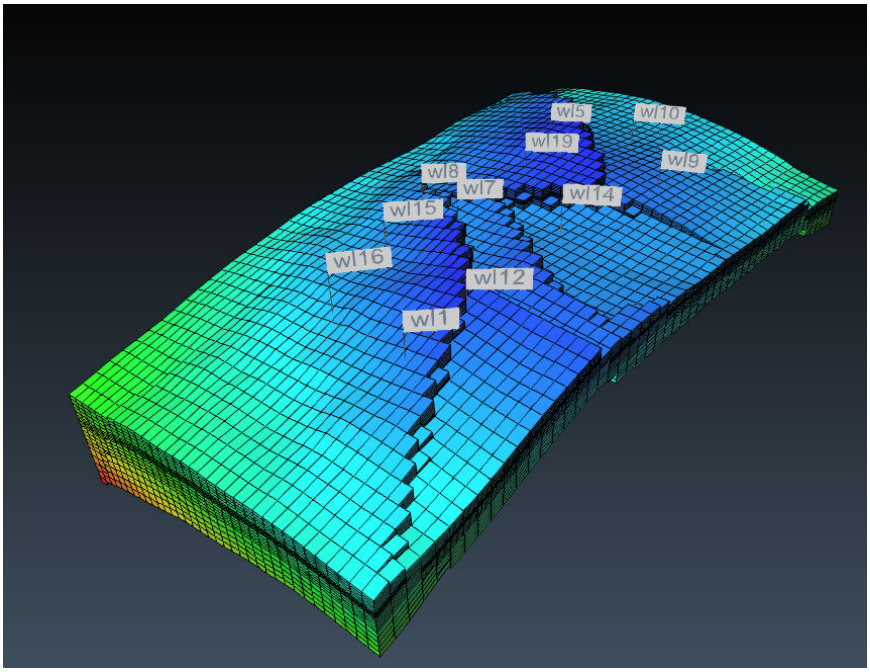


Figure 3.40: Reservoir model at the start of the forecast (Grid Top)

The Field NPV ($) objective function is calculated based on the following criteria in Table 3.16:

Table 3.16:Field NPV calculation criteria

	$/bbl	**$/m3**	**Rate of Increase per year**	**Rate of increase per month**
Oil Price (Start of Forecast)	$50.00	$251.59	3.000%	0.247%
Water Injection Price (Start of Forecast)	$0.50	$3.14	2.500%	0.206%
Infill Drilling Cost:	$6,000,000.00		**Discount Rate:**	10%

To conduct this study, we will incorporate the following group of parameters in the model:

- Shut-in/recompletion criteria for producers:
    - Minimum Oil Rate
    - Maximum Water Cut
    - Maximum GOR
- Decision to convert wells wl5 & wl16 to injectors
- Infill Well locations
- Group Water Injection Rate

The following parameters in Table 3.17 are used in the model accordingly:

Table 3.17: The parameters of the infill well drilling optimization study

Parameter number	Parameter Name	Unit	Lower Limit	Upper Limit	Type
1	GroupInjRate	m^3/day	1000	7500	Continuous
2	Infill_Hz_Prod_J	-	3	9	Discrete
3	Infill_Hz_Prod_K	-	5	18	Discrete
4	Infill_Inj1_I	-	31	38	Discrete
5	Infill_Inj1_J	-	2	8	Discrete
6	Infill_Inj2_I	-	25	35	Discrete
7	Infill_Inj2_J	-	27	34	Discrete
8	MaxGOR	-	400	1500	Continuous
9	MaxWCUT	-	0.75	0.995	Continuous
10	MinOilRate	m3/day	0	5	Continuous
11	wl5_type	-	1	2	Discrete
12	wl16_type	-	1	2	Discrete

Parameters 8, 9, and 10 are shut-in/recompletion criteria and monitoring constraints for the producers. The 'Infill' parameters are for specifying the location of the infill wells. We define the optimal placement by adjusting the J and K block addresses for parameters 2 and 3, and I & J block addresses for parameters 4 to 7. Parameter 1 will adjust the injection rate on the entire group of injector wells.

Parameters 11 and 12 are for whether or not to convert well 5 (wl5) and well 16 (wl16) to injector. For them, a value 1 will represent that the well will be a producer and the value 2 will represent that the well will be an injector. Both well 5 and well 16 default status is producers.

3.6.5.1 Optimization process

Now with the parameters set up, we optimize the model for maximizing the Field NPV. The focus of this case study is not on the optimization part of the modeling, but a brief description of the method is demonstrated in the following Figure 3.41:

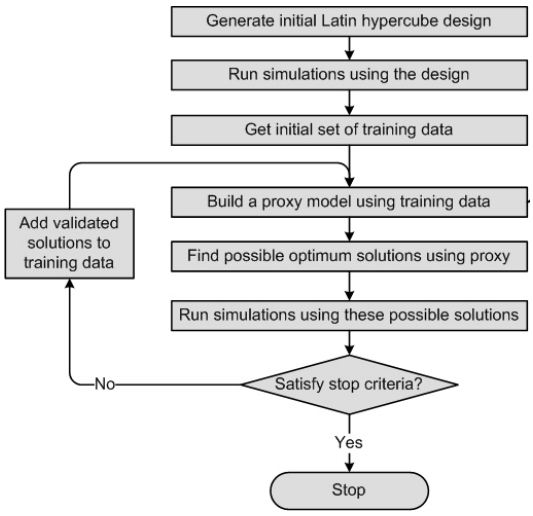


Figure 3.41: The optimization algorithm for the infill well study

Usage of this optimization algorithm involves the following four steps:

1. Latin Hypercube Design: As explained previously in this chapter, the purpose of Latin hypercube design is to construct combinations of the input parameter values so that the maximum information can be obtained from the minimum number of simulation runs. Latin hypercube design is chosen here because it can handle any number of input parameters with mixed levels.

2. Proxy Modeling: In this step, an empirical proxy model is built using training data obtained from Latin hypercube design runs.

3. Proxy-based Optimization: Due to the intrinsic limitations of a proxy model, it is generally recognized that they usually cannot give accurate predictions for highly nonlinear multidimensional problems. Therefore, the optimal solution obtained based on the proxy model may not be the true optimal for the actual reservoir model. This means that certain suboptimal solutions of the proxy model may become the true optimal solution for the actual reservoir model. To counteract false optimum predictions, a pre-defined number of possible optimum solutions (i.e., suboptimal solutions of the proxy model) are generated to increase the chance of finding the global optimum solution.

4. Validation and Iteration: For each possible optimum solution found through proxy optimization, a reservoir simulation needs to be conducted to obtain the true objective function value. To further improve the prediction accuracy of the proxy model, the validated solutions can be added to the initial training data set. The updated training data set can then be used to build a new proxy model. With the new proxy model, a new set of possible optimum solutions can be obtained. This iterative procedure can be continued for a given number of iterations or until a satisfactory optimal solution is found.

Applying this method of optimization for Field NPV with total number of 500 experiments, we can find the optimum solution as shown with the red diamond in Fig. 3.42:

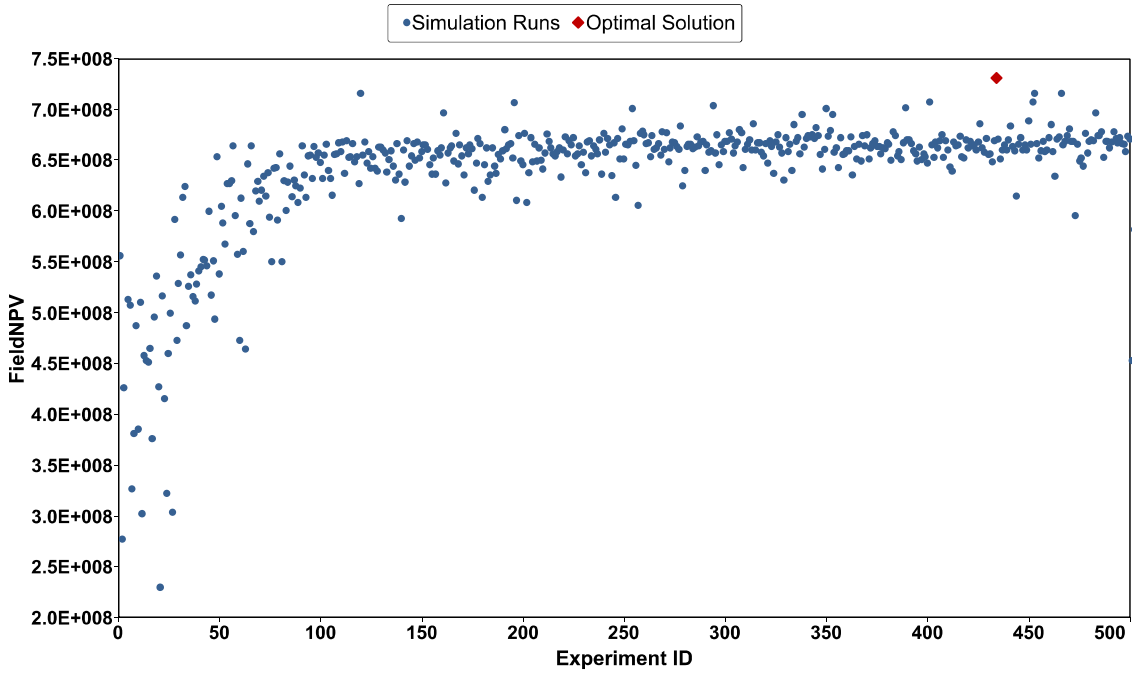


Figure 3.42: Run progress for optimization of Field NPV and the optimal solution

3.6.5.2 Sensitivity analysis using surrogate models

Now, let us study which of our input factors had the most impact on the Field NPV. We have the 500 experiments we created with the Latin Hypercube design, and we train the RBF NN proxy model with these experiments. We further add 38 more experiments to be used as verification data. The RBF NN quality control plot (Figure 3.43) is demonstrated below to check the ability of the RBF NN proxy to approximate the simulation results.

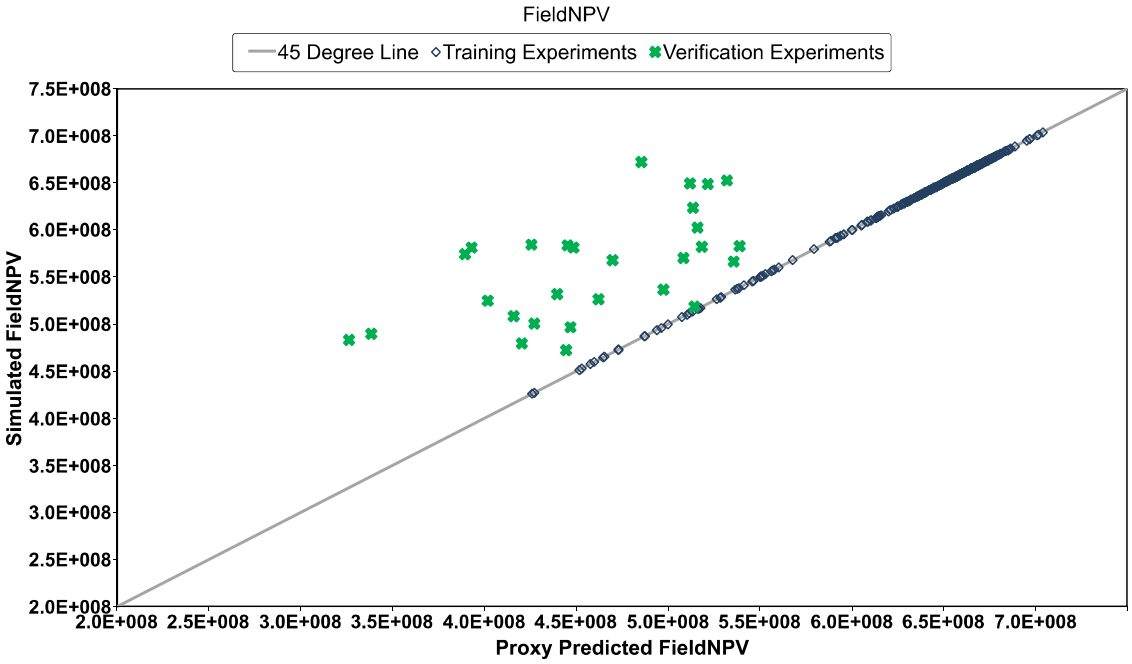


Figure 3.43: RBF proxy quality control plot for Field NPV objective function (R^2-training=1.000, R^2-verification=0.49)

As it can be observed the RBF NN proxy model is not predicting the results very closely to the actual simulation values. If we check the R^2 value for the verification points as well, it is equal to 0.49. (the R^2 for training data is 1 as usual). Therefore, the reliability of the

proxy prediction results for performing sensitivity analysis is doubted. We perform the Sobol SA method anyway and examine the results. Figure 3.44 shows the Sobol analysis result for the Field NPV objective function:

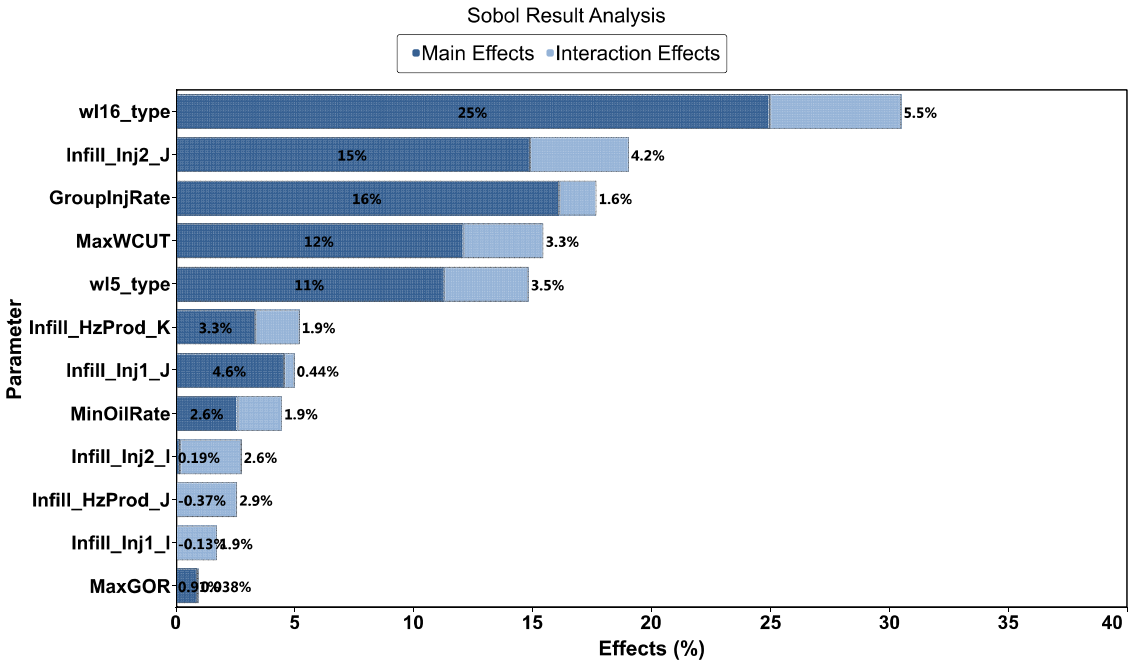


Figure 3.44: Sobol Analysis results based on RBF NN proxy model

The Sobol method finds parameters 12, 7, 1, 9, and 11 as the most significant to the variability of the Field NPV.

Now, we try to see if we can get a better approximation of our model using the configurable Multi-Layer NN proxy. Once the proxy is created, we first check the QC plot (Figure 3.45) to see if the prediction quality is improved:

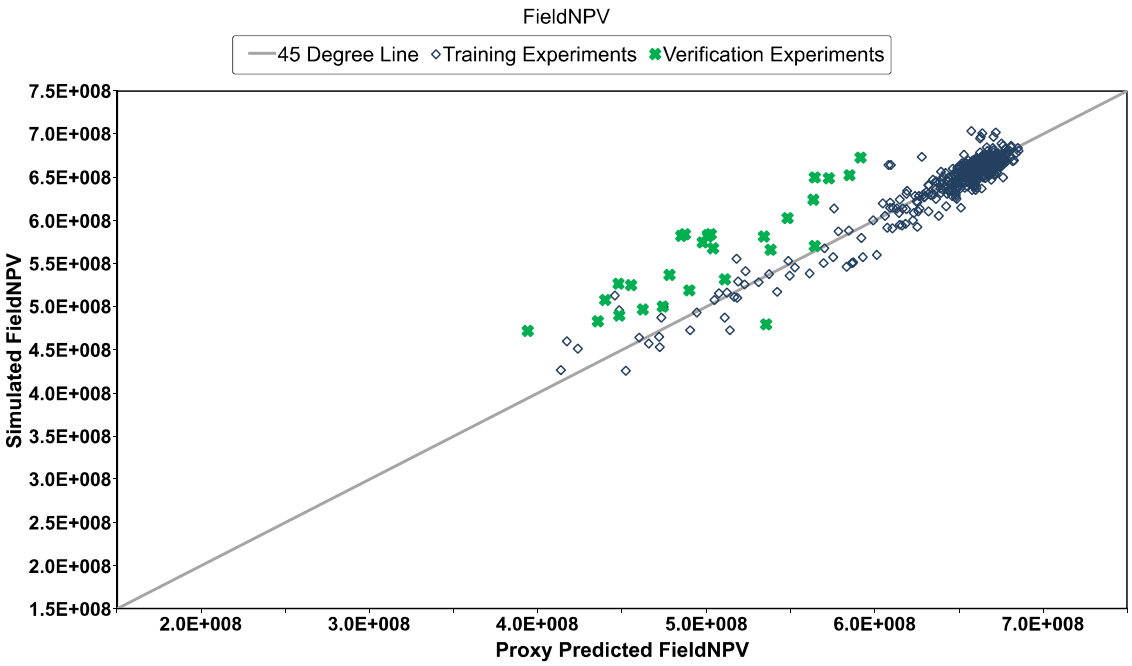


Figure 3.45: Multi-Layer proxy quality control plot for Field NPV objective function (R^2-training=0.947, R^2-verification=0.791)

We can clearly see the Multi-Layer NN proxy is providing a much better prediction quality compared to the RBF proxy previously used. Examination of the R^2 values also confirms this. For the configuration of 3 nodes, with 2,1,1 hidden layers, the R^2 value for training data equals to 0.947, and the R^2 for the verification points is 0.791 which is much more reliable than the values obtained for RBF proxy. Applying the Sobol SA method using Multi-Layer NN results in Figure 3.46:

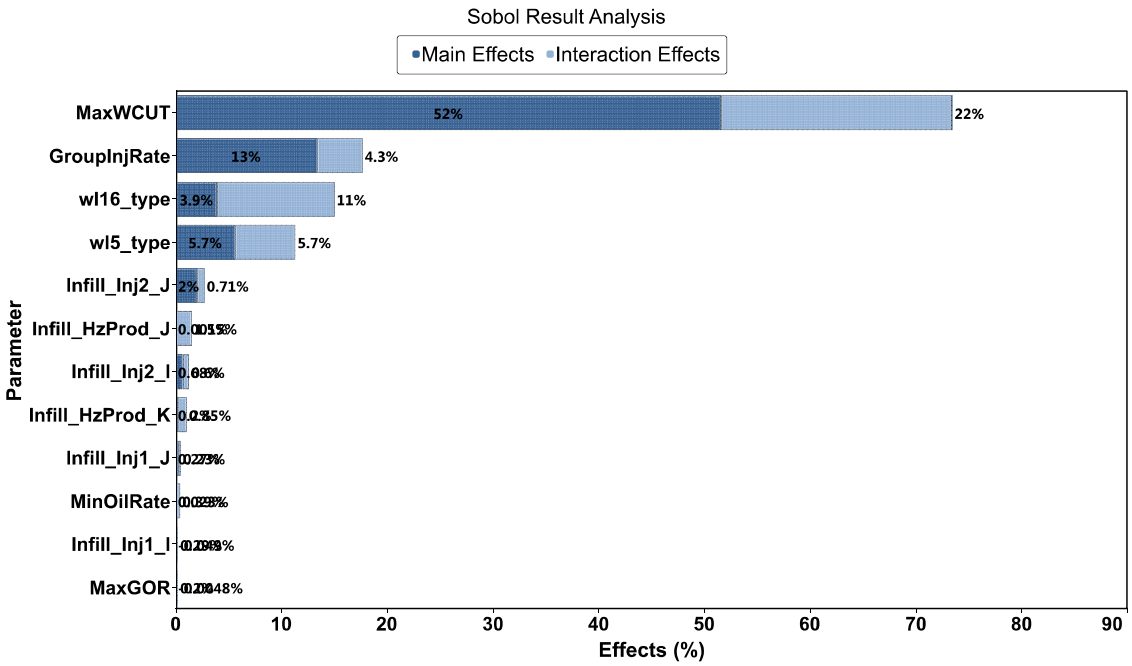


Figure 3.46: Sobol Analysis results based on Multi-Layer NN proxy model

As it can be observed, the maximum water cut constraint as well as the group injection rate, well 16 type, and well 5 type (parameters 9, 1, 12, and 11) are now the most significant factors contributing to the Field NPV objective function variability. By examining the cross plots of these parameters obtained from the simulation (Figures 3.47 to 3.50), we can confirm the Sobol results using Multi-Layer Neural Network is much more accurate than the case of RBF proxy.

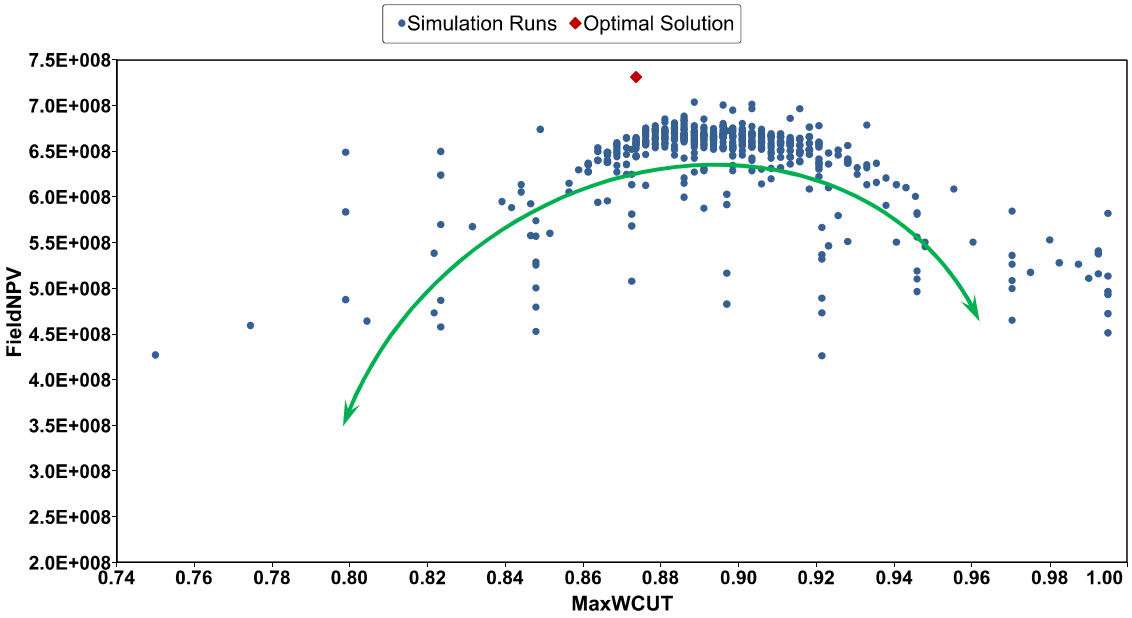


Figure 3.47: Cross plot of parameter 'MaxWCUT' vs. Field NPV

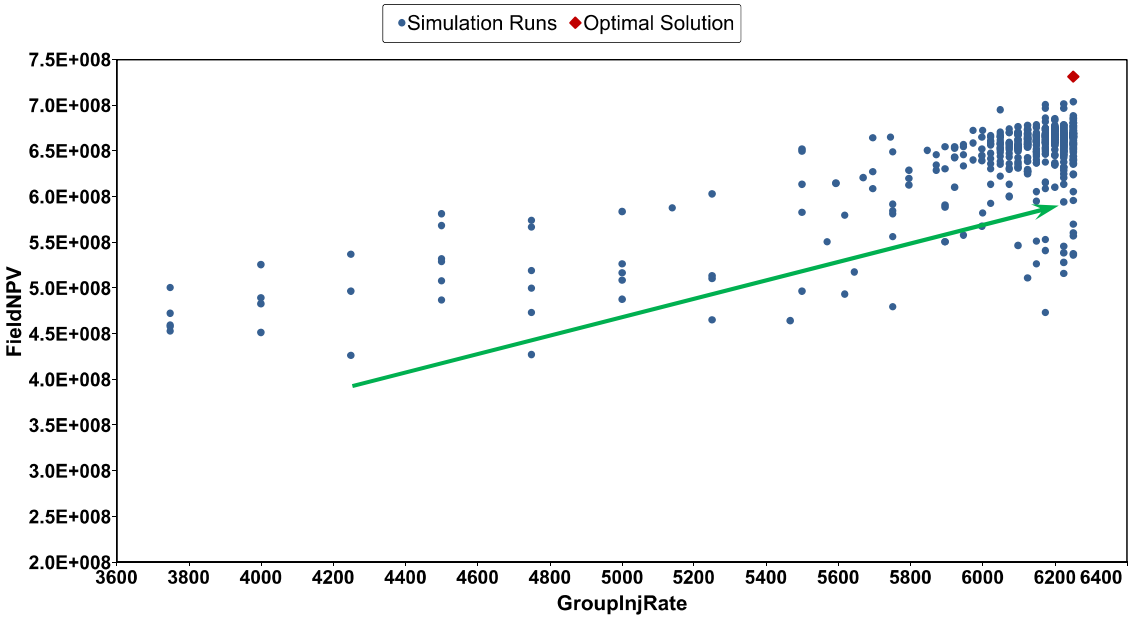


Figure 3.48: Cross plot of parameter 'GroupInjRate' vs. Field NPV

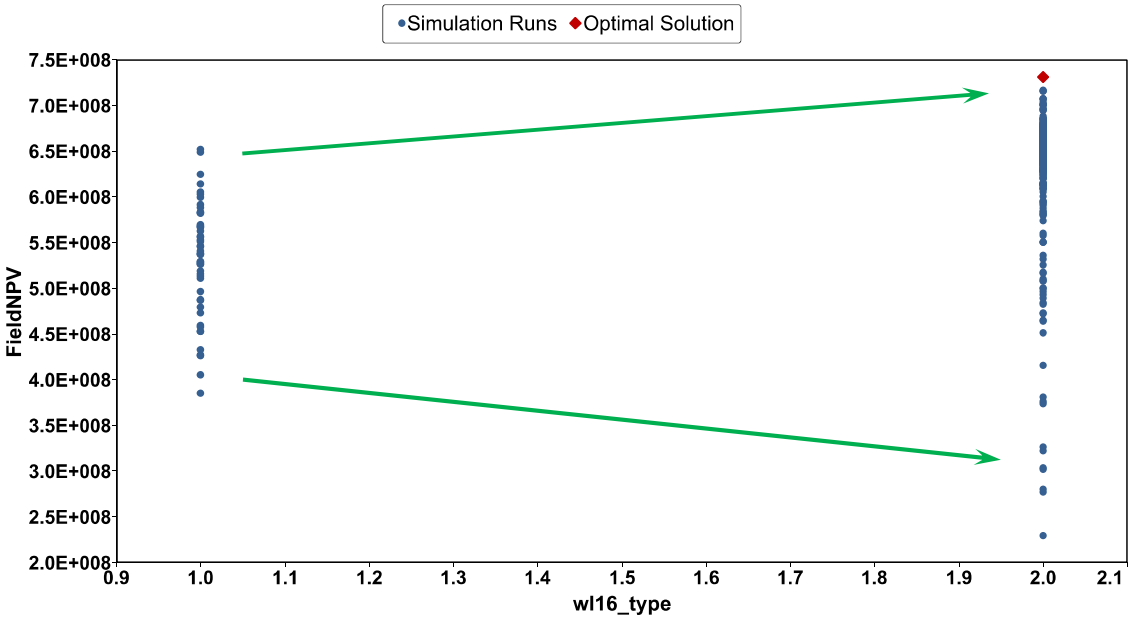


Figure 3.49: Cross plot of parameter 'wl16_type' vs. Field NPV

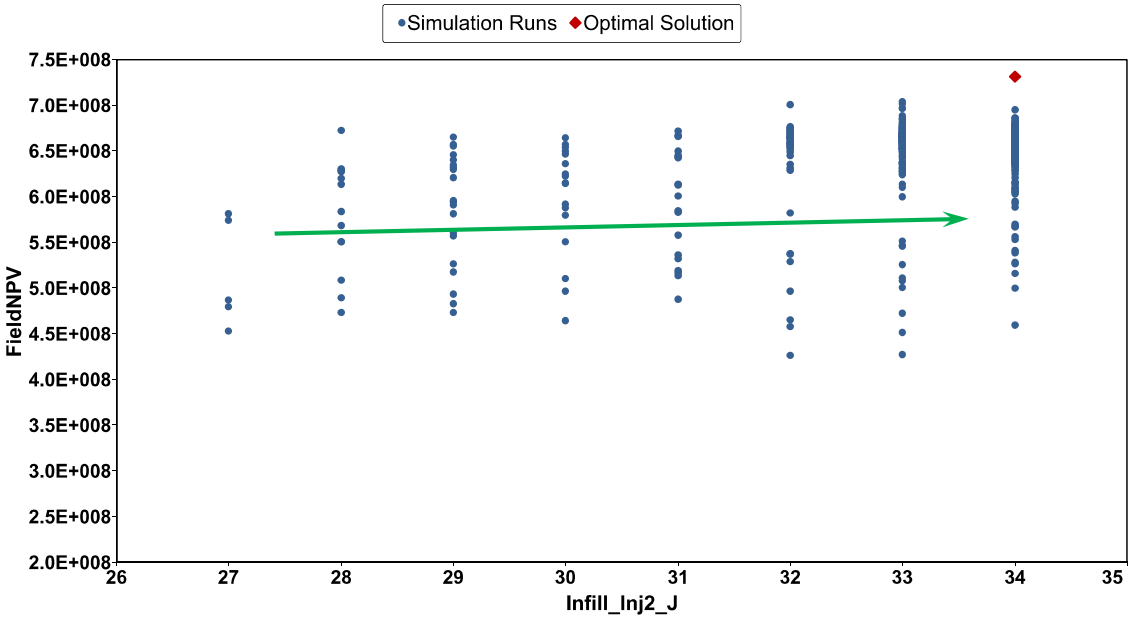


Figure 3.50: Cross plot of parameter 'Infill_Inj2_J' vs. Field NPV

In the case of using the RBF NN proxy, parameter 12 (wl16_type), followed by parameter 7 (Infill_Inj2_J) were identified as the 1st, and 2nd most significant parameters to Field NPV, however by observing the cross plots in Figure 3.49 and Figure 3.50 we can see that these were over estimations due to inaccuracy of RBF prediction.

This case study demonstrated the application of different proxy models incorporated in this research and the advantage of Multi-Layer NN proxy over RBF NN in a case where RBF NN was unable to accurately approximate the reservoir model.

3.7 Summary and Conclusions

In this chapter, we presented a critical review of Sobol variance based method for performing sensitivity analysis of complex, computationally expensive reservoir simulation models. This method provided quantitative measures of parameter effects and importance, and the relationship between the variances of the model inputs and outputs. This is useful and practical for reservoir engineering design purposes by providing a better understanding of how various parameters affect the response of a system so that design changes can be made. On the other hand, Sobol method provided information regarding which uncertain inputs should be better understood to most effectively reduce the uncertainty in the output.

The predominant limiting factor in successfully utilizing variance based sensitivity analysis methods is the enormous computational burden. As a result, integration of methods for approximating, or emulating, complex reservoir models was discussed in this chapter. We showed how to construct a simplified, probabilistic model, called a surrogate model, that was capable of approximating the output from the reservoir simulation model. Once constructed, the surrogate model served as a fast-running proxy to the reservoir simulation model and was used to predict outputs from a Monte Carlo simulation. While numerous different approaches to surrogate modeling have been proposed in the literature, each with unique advantages and disadvantages, in this thesis, we have restricted our attention to the RBF and Multi-Layer Artificial Neural Networks and proved their reliability within the framework of Sobol sensitivity analysis method.

We also presented a detailed discussion of sampling methods for including standard Monte Carlo simulation, Latin Hypercube sampling, and low-discrepancy sequences and compared the relative advantages and drawbacks of these methods to each other. In the following chapter, we will discuss a different approach to sensitivity analysis.

Chapter Four: Distance Based Sensitivity Analysis

In many engineering applications, understanding the model parameters and their independent and combined effects on the model is significant. Many types of parameters exist in reservoir modeling, which could come from geophysical, geological and engineering categories. Some of these parameters are continuous, others could be discrete, and others may have no numerical value and are scenario based (Fenwick et al., 2014). In this chapter, another sensitivity analysis method is introduced which focuses on being able to include any type of input parameter while also handling parameter interactions. It is then applied to complex reservoir systems with geological uncertainties to demonstrate its robustness. The sensitivity analysis method discussed in this chapter is primarily based on classifying the response/decision variables into a limited number of discrete categories. The results analysis is based on the fact that if the parameter frequency distribution in each class is the same, the model response would be considered insensitive to the parameter. At the same time, differences in the frequency distributions demonstrate the fact that the model response is sensitive to the parameter. Based on this structure, a new general measure of sensitivity is understood. This sensitivity measure is also able to quantify the model sensitivity to parameter interactions, and includes the possibility whether these interactions can be asymmetric in the context complex reservoir modeling (Fenwick et al., 2014). The approach is verified using the cases studies from the previous chapters and is

also illustrated with a case study of a large reservoir case while incorporating the reservoir's geological uncertainty in the study.

The primary objective of this method is to overcome the limitations of the other SA approaches while allowing the quantification of asymmetric parameter interaction. The main upside of this approach is the fact that no assumptions on the nature or prior distributions of the input parameters are made. The same is true for the response functions. This chapter reproduces the work of Caers (2011), Scheidt and Caers (2009) and Fenwick and Scheidt (2014) but within the same context of finite difference reservoir simulation and the other tools and sampling strategies developed in the previous chapters. The mentioned studies themselves are based on the work of Spear and Hornberger (1980), which they call Generalized Sensitivity Analysis, and further expanded by other authors (Beven and Binley 1992; Bastidas et al. 1999; Pappenberger et al. 2008).

4.1 Basic Concept

The basic concept of this method of global sensitivity analysis is to separate the models into discrete classes. Spear and Hornberger (1980) demonstrate that if the given models in their work were separated into distinct classes and they study the distributions of the parameter within each class, they can gain insight about the parameter sensitivities based on the principle that non-influential parameters will have no impact on the classification, and thus the distributions of the parameter values will be similar between classes. On the other hand, an influential parameter will separate the models into the different classes, and

will thus be evident when comparing the distributions of the sampled parameter values between classes. Therefore, this type of GSA method has the following advantages:

- It is applied to an ensemble of models which is created by Monte Carlo sampling of prior CDFs.
- It is solely performed by comparison of CDFs, which can be discrete, continuous or any form.
- It does not require any assumption about the state, behavior or the functional from of the response.

Scheidt and Caers (2009), extend Spear and Hornberger's work (1980) to include high dimensional outputs that are common in reservoir modelling studies. In addition, the classification of the output is based on definition of a "distance". Therefore, this approach is called Distance-based Global Sensitivity Analysis (DGSA). It is also further extended by quantifying the influence of asymmetric parameter interactions. In the later sections of this chapter, we will apply this methodology and demonstrate its application to data from actual field reservoirs as well as sensitivity analysis test functions.

4.2 Distance-Based Classification of Responses

Scheidt and Caers (2009) extend the work of Spear and Hornberger (1980) to high-dimensional responses by analyzing responses using distance-based classification. First, classifying the responses into a limited set of classes (k being the number of classes), we would have:

$$c = \{c_1, c_2, \dots, c_k\} \quad (4.1)$$

meaning that, based on some measure of similarity/dissimilarity (a distance) that we define, responses r_i would be attributed uniquely to a class c_k . It must be highlighted that the classification is performed on the model responses r_i, and not on the input parameters p of the models. For each class the number of responses is denoted as

$$n_c = \{n_{c1}, n_{c2}, \dots, n_{ck}\} \quad (4.2)$$

The objective of the methodology is efficient selection of a subset of responses in the model ensemble, such that they exhibit the same statistical properties such as densities, quantiles, mean, variance, etc. as the entire set of models. It is important to note that since this method relies on the assumption that after the classification a few selected realizations have the same statistical characteristics in terms of response as the entire set, thus, no prediction outside of the existing set of responses is done with this method, and the key is to select properly the subset of realizations. Since in reservoir modeling time-varying responses over several wells are considered, standard distances can be employed. The following section describes the methodology used in this approach.

4.2.1 Definition of Distance

The first step of the methodology is defining what a similarity/dissimilarity distance means between any two realizations. The concept of similarity between geostatistical model realizations was introduced by Arpat (2005), and Suzuki and Caers (2008). The distance specifies how similar two models are in terms of spatial properties and transfer function response. The distance between two realizations can be determined by classical distances

that measure difference in geometry such as the Hausdorff distance (Dubuisson and Jain 1994). However, the Hausdorff distance does not consider connectivity between wells which is necessary for our cases here to evaluate flow differences in different models. Connectivity-based distances (Park and Caers 2007), or streamline based simulation are other examples of possible distances that could be used. When talking about distance for flow response, it is important to note that in order to measure similarity or dissimilarity, it is not necessary to determine statistical measures for each realization. This distance can be measured in any way as long as it could allocate a reasonable correlation with the difference in response of the same two realizations.

The distance is defined as a single scalar value that quantifies the difference between two models, *a* and *b*. It is usually denoted by $\delta_{a,b}$. By definition, we have that $\delta_{a,a} = \delta_{b,b} = 0$ which means the distance of a model to itself is zero; and that $\delta_{a,b} = \delta_{b,a}$, meaning the distance is symmetric.

The choice between distances provides an opportunity to choose a distance that is related to the response differences between models. This will allow structuring uncertainty with a particular response in mind and create better insight into what uncertainty affects the response most. In making this distance a function of the desired response, the clustering of responses becomes effective for the response uncertainty question we are trying to address. For example, for a transfer phenomenon a distance measuring the connectivity difference from source to well between any two models would be considered a good choice of distance. A simple explanation of the distance is to view it similar to an objective function

in history matching in which the field data is replaced by data coming from another model instead (Thiele et al., 2016). For example, the distance in the field oil rate response between two models, a and b, can be written as:

$$\delta_{a,b}^{oil} = \frac{\sum \Delta t_i \left| \mathrm{Q}_{a,i}^{oil} - \mathrm{Q}_{b,i}^{oil} \right|}{\sum \Delta t_i} \qquad (4.3)$$

This is simply a difference measure in the oil rate over time between model a and b. The distances might be defined for any output response, grid properties or values such as OIP. If there are *N*-models, then there are $\frac{N \times (N-1)}{2}$ distances to compute and the dissimilarity matrix can be constructed and will be a symmetric $N \times N$ matrix with zero diagonal entries. If historical data is available, then the same formula can also be used for filling-in an additional column/row containing the objective function for all the models. In this case, the dissimilarity matrix will be of size $(N+1) \times (N+1)$.

It is difficult to get insight from the dissimilarity matrix directly. As result, after we construct the distance matrix D, all the realizations are mapped into an Euclidean space R using multidimensional scaling (MDS). This will allow getting the position of each model relative to all the others in a lower-dimensional space, and also visualizing the distance matrix in a way that the relationship between models is preserved as a Euclidean distance (Thiele et al., 2016). In this case, the models that are very different from each other will exhibit a larger separation in the lower dimensional MDS space than models that are similar and we would be able to visually observe these differences.

4.2.2 Multidimensional Scaling (MDS)

MDS is used as a method to map the dissimilarity matrix into a specific layout of points in nD Euclidean space. The spatial representation of the points is arranged such that their Euclidean distances correspond as much as possible to the dissimilarities of the actual models (Scheidt and Caers, 2009). A successful MDS procedure would result in a good correlation between the dissimilarity distance and the Euclidean distance. The classical MDS algorithm rests on the fact that the coordinate matrix X of the points can be derived by eigenvalue decomposition from a matrix A obtained by converting the dissimilarity matrix D into a scalar product. We should note that because the map of points obtained by MDS is derived only by the dissimilarity distances in the matrix, the absolute location of the points is irrelevant. It is possible to subject the MDS map to translation, rotation, and reflection, without impacting the methodology. That's because only the distances in mapping space R are of interest (Caers, 2011).

To define the notion of distance mathematically, a single model i is represented by a vector x_i, which contains a list of variables that quantify that model uniquely. The "size" or "dimension" N of the model is then the length of this vector (Park, 2011). N is typically very large. L denotes the number of alternative models generated with typically $L \ll N$. All models are collected in the matrix X:

$$X = [x_1 x_2 \dots x_L]^T \qquad (4.4)$$

One of the most studied distances is the Euclidean distance, which is defined as

$$d_{ij} = \sqrt{(x_i - x_j)^T (x_i - x_j)} \quad (4.5)$$

if this distance were applied to a pair of models x_i, x_j (Caers, 2008). A distance, such as a Euclidean distance, defines a metric space M, which is a space only equipped with a distance; hence it does not have any axis, origin, nor direction as, for example, a Cartesian space has. This means that the location of any x in this space cannot be uniquely defined, only how far each x_i is from any other x_j, since only their mutual distances are known. Even though locations for x in M cannot be uniquely defined, some mapping or projection of these points in a low dimensional Cartesian space can, however, be presented. To construct such maps, a traditional statistical technique termed multidimensional scaling (MDS) is employed. The MDS procedure works as follows. The distances are centered such that the maps origin is 0. It can be shown that this can be done by the following transformation of the distance d_{ij} into a new variable b_{ij},

$$b_{ij} = -\frac{1}{2}\left(d_{ij}^2 - \frac{1}{L}\sum_{k=1}^{L} d_{ik}^2 - \frac{1}{L}\sum_{l=1}^{L} d_{lj}^2 - \frac{1}{L^2}\sum_{k=1}^{L}\sum_{l=1}^{L} d_{kl}^2\right) \quad (4.6)$$

This scalar expression can be represented in matrix form as follows. Firstly, construct a matrix A containing the elements

$$a_{ij} = -\frac{1}{2} d_{ij}^2 \quad (4.7)$$

then, center this matrix as follows

$$B = (HAH) \quad (4.8)$$

With

$$H = I - \frac{1}{L}\,\mathbf{1}\mathbf{1}^T \quad (4.9)$$

With $\mathbf{1} = [1\ 1\ 1\ ...\ 1]^T$ a row of L ones, I, the identity matrix of dimension L. B (of size $L \times L$) can also be written as:

$$B = (HX)(HX)^T \quad (4.10)$$

Considering the eigenvalue of decomposition B as:

$$B = V_B \Lambda_B V_B^T \quad (4.11)$$

In our case, $L \ll N$ and the distance is Euclidean, hence all eigenvalues are positive. We can now reconstruct (map onto a location in Cartesian space) any x in X in any dimension from a minimum of one dimension up to a maximum of L dimensions, by considering that

$$B = (HX)(HX)^T = V_B \Lambda_B V_B^T \Rightarrow X = V_B {\Lambda_B}^{0.5} \Rightarrow X_d = V_{B,d} {\Lambda_{B,d}}^{0.5} : \quad X \rightarrow X_d\ (MDS) \quad (4.12)$$

if we take the d largest eigenvalues. $V_{B,d}$ contains the eigenvectors belonging to the d largest eigenvalues contained in the diagonal matrix $\Lambda_{B,d}$. The solution X_d retained by MDS is such that the mapped locations contained in the matrix $X_d = \left[x_{1,d} x_{2,d} \ldots x_{L,d}\right]^T$ have their centroid as origin and the axes are chosen as the principal axes of X. The length of each vector $x_{i,d}$ is d, the chosen mapping dimension.

Traditional MDS was developed for Euclidean distances, but as an extension the same operations can be done on any distance matrix. Assembling the dissimilarity matrix and

building the MDS space results in a display of the relationships among all the models as shown.

4.3 Clustering

Clustering techniques are commonly used tools in computer science and some other areas like Earth Sciences. In clustering, the goal is to divide a set of objects into mutually exclusive classes based on information provided on that object. What is unknown and need to be defined are the number of classes and what features or attributes of the object should be used to make such division. The number of classes, clusters or groups can either be decided on:

- how many response evaluations could be made based on computational power

  or

- how many response evaluations are necessary in order to obtain the desired accuracy for a realistic assessment of uncertainty on the response.

4.3.1 K-Means clustering

There are several different clustering techniques available in the literature. A simple technique is termed *k*-means clustering. Similar to other clustering methods, the objective of the *k*-means clustering method is to cluster n objects into k classes. The value of k is specified by the user based on the criteria explained above or can also be suggested by methods like the Davies–Bouldin index (1979), or the Silhouette index (Rousseeuw 1987). In traditional *k*-means clustering, such objects are characterized by m attributes. The

objects are then plotted in *m*-D Cartesian space. The following algorithm summarizes the *k*-means algorithm:

1. Randomly selected *m* cluster centers are chosen for initialization.
2. The distance between each object and the cluster centers is calculated.
3. Objects are assigned to the center closest to it.
4. A new cluster center is calculated for each cluster based on the assigned objects.
5. The process is repeated from step 2 again until no changes in the cluster centers are observed.

The number *k* needs to be specified and may not be easy to know a priori. One advantage is that the *k*-means clustering algorithm works as long as the distance between objects is known.

k-means works well for a lot of cases, but may go wrong in more difficult cases where the variation of points in the 2D plot is quite nonlinear. Also, the *k*-means method is sensitive to the initialization of the algorithm in step 1, and a "bad" initialization can result completely wrong grouping of objects sometimes. A solution that works well is to make the distribution of dots in this 2D plot more linear or at least more aligned in a specific direction. Performing a kernel transformation is a way to achieve this that is explained in the following section.

4.3.2 Kernels and Feature space

The selection of representative models can prove difficult if the ensemble of models in a 2D or 3D projection Cartesian space has a complex shape. In computer science kernel

techniques are used to transform from one metric space into a new metric space such that, after projecting in 2D, 3D and so on, the cloud of models displays a simpler arrangement. To achieve this, we can consider a model x_i and its transformation using some multivariate function φ:

$$x_i \rightarrow \varphi \text{ or } \begin{pmatrix} x_1 \\ . \\ . \\ . \\ x_L \end{pmatrix} \rightarrow \begin{pmatrix} \varphi(x_1) \\ . \\ . \\ . \\ \varphi(x_L) \end{pmatrix} \qquad (4.13)$$

Now, consider that we are trying to achieve Figure 4.1.

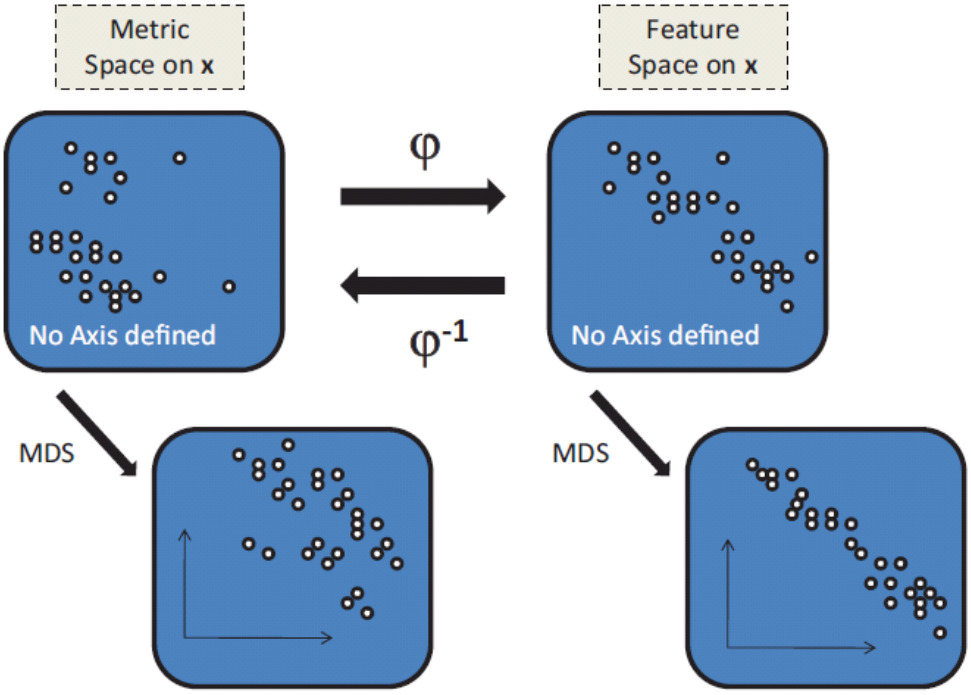


Figure 4.1: Metric and feature space and the projections with MDS (Caers, 2011)

If we want the points of the ensemble on the metric space to be simpler, by knowing the distance between any two x_i, we do not need to change x_i individually, we only need to transform the distances between the x_i, in order to rearrange the cloud. We also do not need a function to change x_i but a function that changes the distances between them. This is practical, since x is of very large dimension N and finding such a multivariate function may be hard, however the distance is a simple scalar and there are only $L \times L$ distances to transform. A function that transforms distances or transforms from one metric (or distance) space to another is termed in technical literature a kernel function. There are many kernel functions, and we will just discuss one that was previously also discussed in chapter 3, the radial basis kernel (RBF) which in all generality is given by:

$$K_{ij} = k(x_i, x_j) = \exp(-\frac{(x_i - x_j)^T - (x_i - x_j)}{2\sigma^2}) \quad (4.14)$$

This RBF is function (a single scalar) of two reservoir models x_i and x_j and of parameter σ, called the bandwidth, which needs to be chosen by the modeler. The bandwidth acts like a scalar of length. Modelers have found that choosing the bandwidth equal to the standard deviation of values in $L \times L$ distance matrix is a reasonable choice (Caers, 2011). This RBF is function of the Euclidean distance $(x_i - x_j)^T - (x_i - x_j)$. If we want to make this RBF a function of any distance and not just the Euclidean distance, it can be done simply by means of MDS according to Caers (2011) as follows:

1) Specify the distance $d(x_i, x_j)$

2) Use MDS to plot these locations in a low dimension, call these locations $x_{d,i}$ and $x_{d,j}$ with the dimension of d in the MDS plot.

3) Calculate the Euclidean distance between $x_{d,i}$ and $x_{d,j}$.

4) Calculate the kernel function with given σ using

$$K_{ij} = k(x_i, x_j) = \exp(-\frac{(x_{d,i} - x_{d,j})^T - (x_{d,i} - x_{d,j})}{2\sigma^2}) \quad (4.15)$$

Because there is a new indicator of distance, K_{ij}, we also have a new metric space, which is traditionally called the "feature space" (Caers, 2011). The eigenvalue decomposition of *K* is calculated and projections are mapped in any dimension, for example in 2D using

$$\varphi_{f=2} = V_{K,f=2}\Lambda_{K,f=2}{}^{0.5} \quad (4.16)$$

where $V_{K,f=2}$ contains the eigenvectors of *K* belonging to the two largest eigenvalues of *K* contained in the diagonal matrix $\Lambda_{K,f=2}$. Some examples to illustrate this are provided in Figures 4.2 and 4.3 from Caers (2011). In the figures, it can be seen how MDS is different with and without kernel transformation.

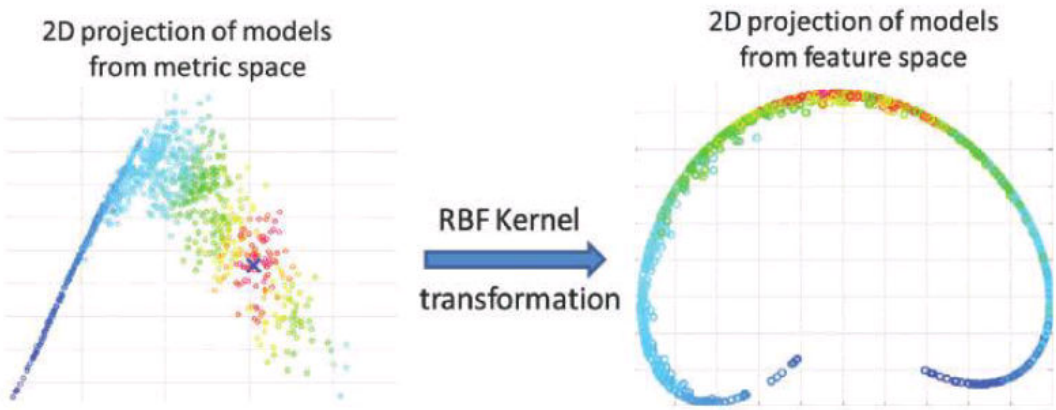


Figure 4.2: RBF kernel transformation (Caers, 2011)

If we first transform the dots using a kernel transformation, as shown in Figure 4.2, then perform k-means clustering and back-transform the dots back into the original space. The kernel transformation requires only the specification of a distance and, as shown in Figure 4.3, would reposition the points into a more organized distribution.

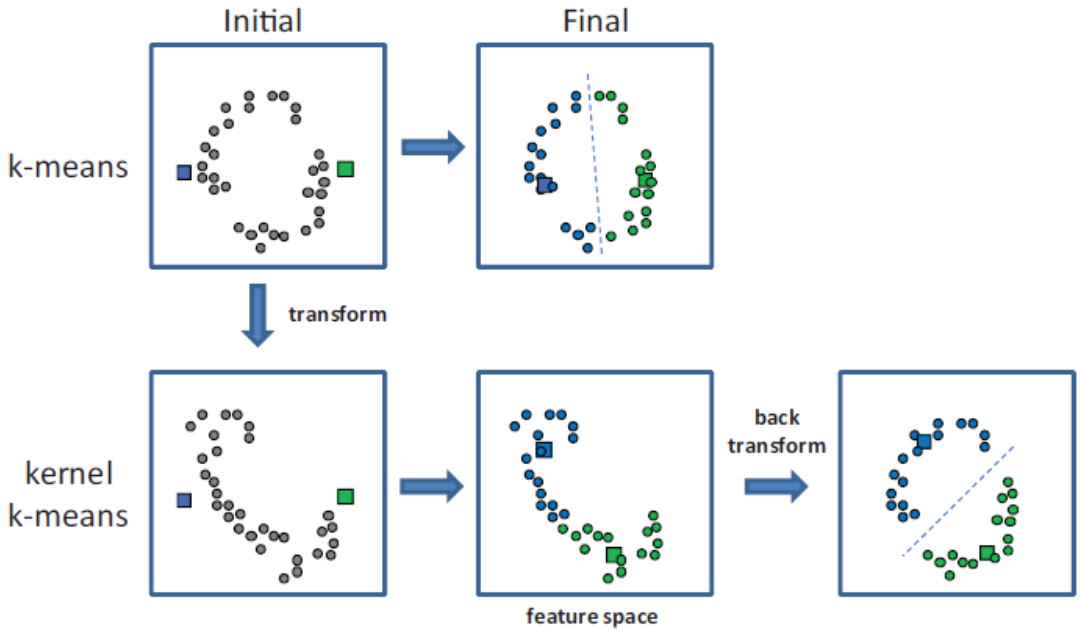


Figure 4.3: k-means and kernel k-means clustering comparison (Caers, 2011)

The following algorithm, named kernel k-means, summarizes these steps:

1) A distance between the n objects is specified.

2) The objects are transformed into a feature space.

3) K-means algorithm presented above is applied.

4) The results are back-transform into the original Cartesian axis system.

Since reservoir models typically include high non-linearity of the responses, we can use the kernel k-means to clusters the models more accurately for the sensitivity analysis purposes described in the next sections.

4.4 Distance-Based Sensitivity Analysis

4.4.1 Main Effects

The clustering result from the previous step separates the models into distinct classes based upon the dissimilarity distance definition and the selected model responses. As explained previously, the appropriate number of classes can be discovered using the Davies–Bouldin index (Davies and Bouldin 1979), or the Silhouette index (Rousseeuw 1987). Alternatively, the user can select the number of classes based upon specific domain criteria. For reservoir engineering applications, it is typical to use the P10, P50, P90 values. For higher number of classification P10, P25, P50, P75 and P90 in ensembles are usually used, so a value of 3 to 5 for the number of clusters is common for k in the reservoir modelling context. If a smaller number of clusters are defined, the clustering would result in differentiating models with large differences in model response. If it is practical to define a bigger number of clusters, subtler differences could be identified in the response, and parameters which impact their variations. But it should be noted that the defined number should be practical in terms of computation as well. In terms of the number of models in each cluster, Fenwick and Scheidt (2013) suggest that at least ten models per cluster should be used. Obviously, the selection of the numbers is completely dependent of the complexity of the models being classified. The more complex the modelling is; the larger number of model evaluations must be done.

Once classification of the output responses is obtained, we would list the values of the parameters responsible for creating the responses belonging to that class. This is done per

each class. Now if the statistical distribution of the sample values of a parameter is similar amongst all clusters then the response would be considered insensitive to the value of that parameter, hence the parameter is non-significant to the variability of the model response. If the opposite is true, then the response is sensitive to the parameter values and the parameter is considered influential. Therefore, some measure of dissimilarity between statistical distributions of a parameter among different clusters would also be a measure of sensitivity for that parameter in relevance to the response (Fenwick et al., 2014). According to this concept, we can define a sensitivity measure as follows:

The prior distribution of each parameter is denoted as $F(p_i)$ (CDF) and its empirical distribution function (from sampling) as $\hat{F}(p_i)$. After classification, we obtain a class-conditional empirical distribution function $\hat{F}(p_i|c_k), \forall i, k$. (Fenwick et al., 2014)

Now in order to evaluate how the classification of the responses impacts the distribution of a parameter p_i, the distance between the prior empirical distribution function and the class-conditional empirical distribution function is calculated. This is done by a calculating the area between the curves numerically:

$$\hat{d}_{k,i} = f_{\Delta cdf}(\hat{F}(p_i), \hat{F}(p_i|c_k)) \quad (4.17)$$

Where $k = 1, 2, \dots, K$.

This distance is denoted as the CDF distance. Equation (4.17) defines the measure of sensitivity for parameter p_i, used in this chapter. It is a single-way sensitivity measure, because it measures sensitivity for a single parameter. Similar to the previous chapters, this

measure is also called the Main Effect as it shows the singular impact of parameter p_i on the response of choice (objective function).

4.4.2 Testing statistical significance using bootstrap

We should be careful to test for statistical significance since the number of points in each cluster is always smaller than the total population size and therefore there will always be a difference because of the resolution of the respective CDFs.

Using the bootstrap results, the CDF distance (Eq. 4.17) can be standardized as follows:

$$s(p_i) = \frac{1}{K}\sum_{k=1}^{K} \hat{d}_{k,i}^{S} \qquad (4.18)$$

Where

$$\hat{d}_{k,i}^{S} = \frac{\hat{d}_{k,i}}{\hat{\hat{d}}_{k,i}^{(q)}} \qquad (4.19)$$

Here $s(p_i)$ defines a standardized sensitivity measure for parameter p_i on the response r. Each term $\hat{d}_{k,i}^{S}$ in the summation in Eq. (4.18) is defined as the standardized CDF distance for class k and parameter p_i. In order to study the results, a Pareto plot can be used to rank these sensitivity measures as we will see in the case study results.

4.5 Parameter Interactions

We can express parameter interactions through conditional distributions. For example, if we consider two parameters p_i and p_j, the same approach used for a single-way sensitivity could be used to quantify sensitivity for two-way parameter interactions (interaction between a pair of parameters); response is insensitive to the parameter interaction under

the condition that there is no difference in the class-conditional distribution of a single parameter and the class-conditional distribution of that parameter additionally conditioned to a second parameter (Fenwick et al., 2014). Therefore, for each pairwise interaction, two parameters p_i and p_i would be considered in addition to the various indicators values $i(p_i; t_m)$ defined as follows for continuous parameters:

$$\begin{aligned} &i(p_i; t_m) = 1 \; if \; t_{m-1} \leq p_i \leq t_m; \\ &otherwise \; i(p_i; t_m) = 0, m = 1, 2, \dots, M \end{aligned} \tag{4.20}$$

with $t_0 = min(p_i)$ and M the number of bins. The thresholds are a subjective and could be chosen by the modeler, with a natural choice being evenly spaced quantiles (Fenwick et al., 2014). For discrete parameters, an indicator value can be defined

$$\begin{aligned} &i(p_i; t_m) = 1 \; if \; p_i = t_m; \\ &otherwise \; i(p_i; t_m) = 0, m = 1, 2, \dots, M \end{aligned} \tag{4.21}$$

where t_m is the m_{th} value of the discrete parameter, and M is the total number of discrete categories. For each class c_k and for each threshold t_m the conditional distribution $F(p_i|i(p_j; t_m) = 1, c_k)$ can be estimated based on the parameter values obtained from the classification. Similar to Eq. (4.17) for individual parameters, a CDF distance is calculated:

$$\hat{d}_{k,i|j,m} = f_{\Delta cdf}\left(\hat{F}\left(p_i|i(p_j; t_m) = 1, c_k\right), \hat{F}(p_i|c_k)\right) \forall k \; \forall m \tag{4.22}$$

Equation (4.22) is used to determine whether any interaction of p_i on p_j is significant enough which influences the response r. Here, we can also first perform a bootstrapped-based hypothesis test to check whether there is a statistically significant interaction and,

then continue to define a measure of conditional interaction sensitivity per class, similar to the approach for single-way sensitivity:

$$s_k(p_i|p_j) = \frac{1}{M}\sum_{m=1}^{M} \hat{d}^S_{k,i|j,m} \;\forall k \qquad (4.23)$$

Where

$$\hat{d}^S_{k,i|j,m} = \frac{\hat{d}_{k,i|j,m}}{\hat{\hat{d}}^{(q)}_{k,i|j,m}} \qquad (4.24)$$

We can also define an average measure of sensitivity over all classes:

$$s(p_i|p_j) = \frac{1}{K}\frac{1}{M}\sum_{k=1}^{K}\sum_{m=1}^{M} \hat{d}^S_{k,i|j,m} \qquad (4.25)$$

Now this measure of sensitivity based on conditional interaction is asymmetric because conditional distributions are asymmetric. Interested readers can check Fenwick et al. (2014) for a more detailed description of the distance based approach.

4.6 Case Studies

4.6.1 Polynomial function

Similar to the previous chapters, we first validate the results of the distance based sensitivity analysis method with the following polynomial function

$$Y = X_1 + 10X_2 + X_3 + 50X_1X_2 \quad (4.26)$$

Using Latin Hypercube Design sampling, we generate 512 experiments from within the X_1, X_2 and X_3 parameter ranges and calculate the model responses accordingly. The next step is to calculate the pairwise distances between pairs of models. Once the pairwise distances are available, we construct the dissimilarity matrix and perform the multidimensional scaling for 2 dimensions. Assembling the dissimilarity matrix and building the MDS space result in a display of the relationships among all the models. Here, this MDS space is only used for visualization purposes, and all necessary calculations required to perform the next stages can be done by directly using the original dissimilarity matrix. Next, we apply the k-means clustering algorithm described in the previous section to categorize these model responses into 3 clusters. The medoids of these 3 groups are also defined at this step. The clusters are displayed in MDS space for easier interpretation of the clustering visualization and may at times be counterintuitive since MDS can only be displayed using a maximum of three dimensions.

The final clustering result is shown in Figure 4.4:

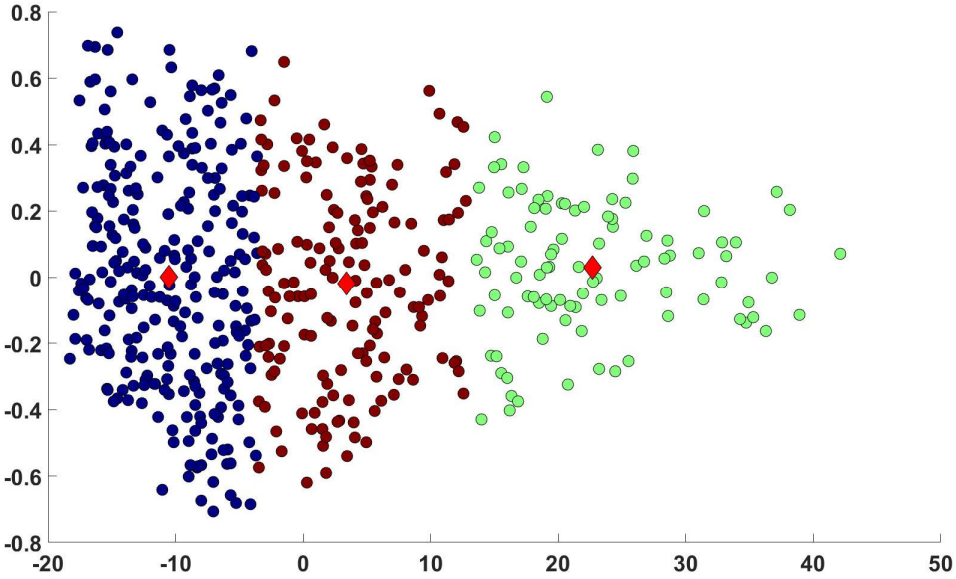


Figure 4.4: 2D clustering result for polynomial function model ensemble

The medoids of the clusters are visualized by a diamond shaped red point in this figure. We can also visualize the result by applying the multidimensional scaling in 3 dimensions as shown in Figure 4.5:

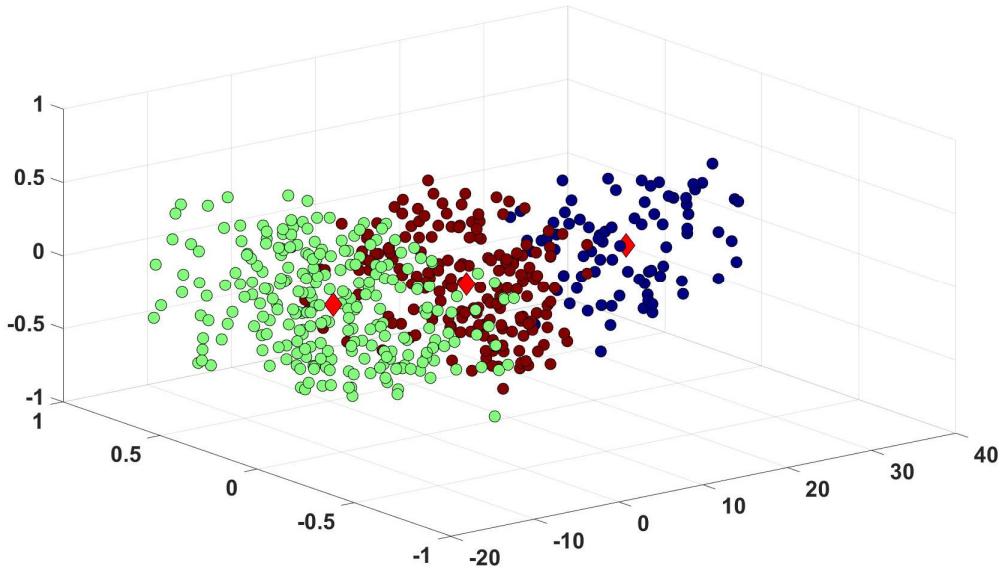


Figure 4.5: 3D clustering result for polynomial function model ensemble

However, the visualization of the clustering result in 3D is counterintuitive in this case as the 3 groups can be more easily identified in the 2D Figure 4.5 shown previous to it.

Once the dissimilarity is clustered into 3 groups as visualized above, the CDF of each parameter in each cluster is compared to the distribution of the entire population (black line in the following figures). The following figures (Figures 4.6 to 4.8) show the empirical CDF curves for the three continuous parameters. The prior CDF (the distribution of the entire population) is in black and the CDFs for the three clusters are in blue (c_1) and red (c_2) and green (c_3).

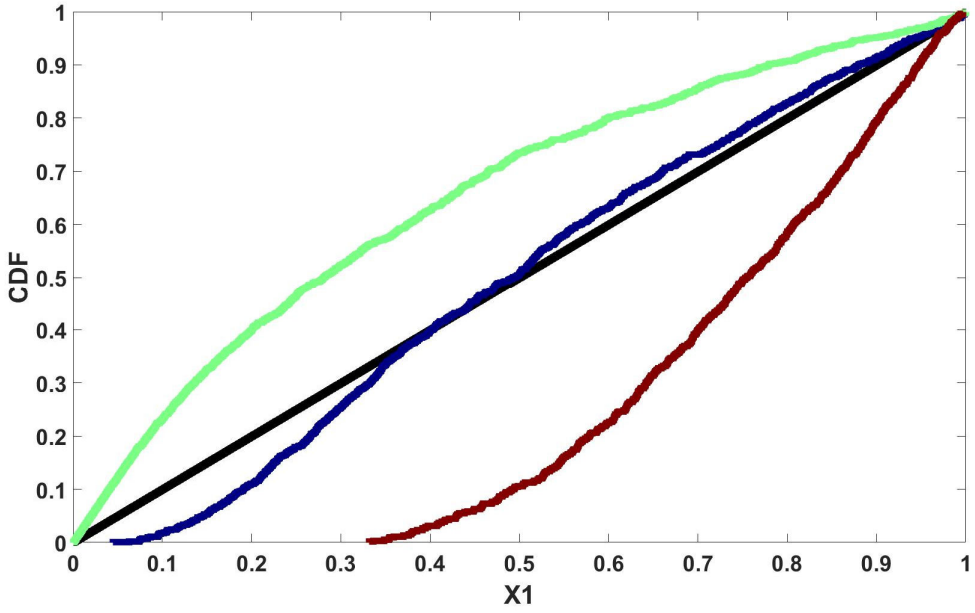


Figure 4.6: Empirical CDF curves for parameter X_1 in polynomial function

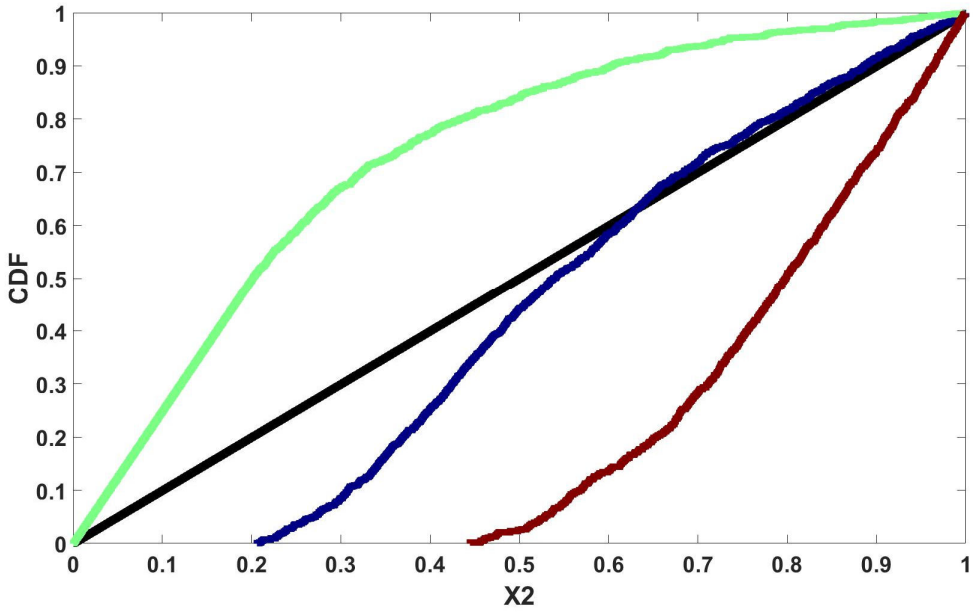


Figure 4.7: Empirical CDF curves for parameter X_2 in polynomial function

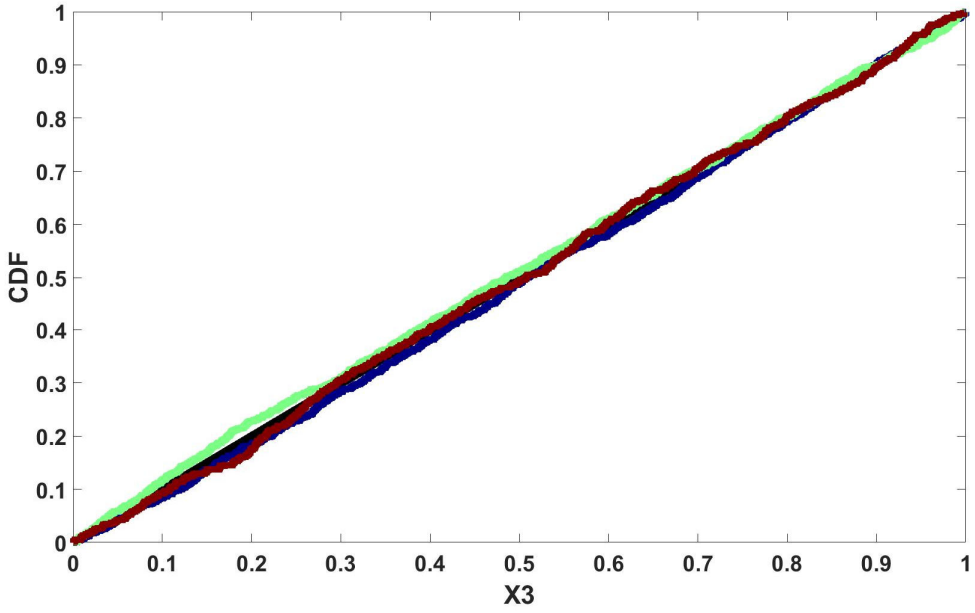


Figure 4.8: Empirical CDF curves for parameter X_3 in polynomial function

It is possible to quantify the extent of sensitivity by the value of the normalized CDF difference, meaning the larger the deviation from the whole population CDF, the more/less sensitive the parameter is considered to be. As it can be observed, the parameters X_1 and X_2 show the largest deviation.

Figure 4.9 shows the results of a SA using the standardized CDF distance calculated from the CDFs in Fig. 4.6 to Fig. 4.8 for each class. If the difference is statistically relevant, the parameter is deemed sensitive and a bar with a value greater than one (normalized difference in CDF) is displayed. If the parameter is insensitive the bar length will be less than one.

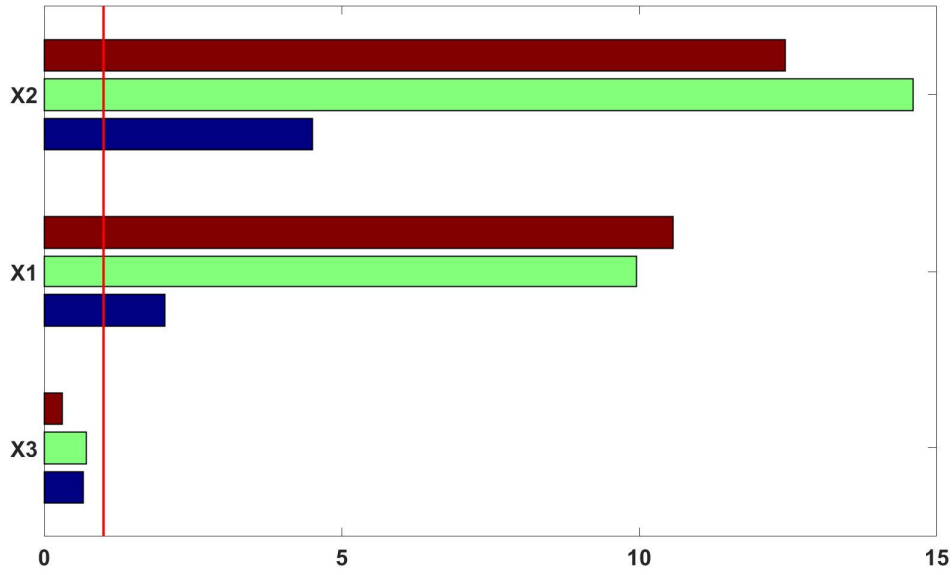


Figure 4.9: Standardized CDF distances for each cluster of polynomial function

The standardized measure of sensitivity which is the average distance $s(p_i)$ for each parameter is displayed in the following Figure 4.10:

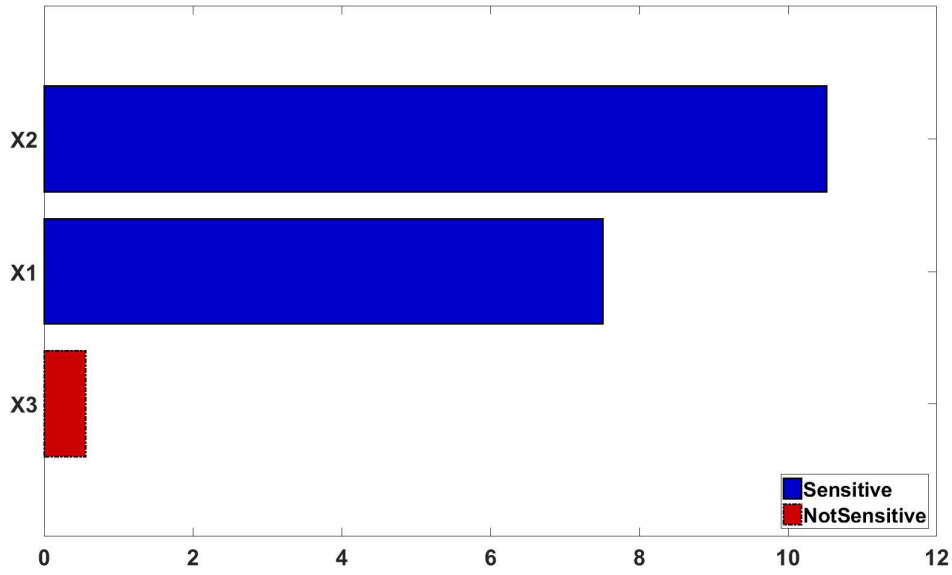


Figure 4.10: Average distance for each parameter of polynomial function

Figure 4.10 matches the analytic sensitivities of the polynomial function and also equate to the results of the previous chapters, and is therefore deemed valid. There are two parameters, X_1 and X_2, for which the distances are significant, and therefore are judged to be influential based on the classification. The other parameter, X_3 has a small distance and is determined to be non-influential to the classification\response of the function.

For the two-way interactions, each parameter is divided into 3 bins; low, medium and high; using evenly spaced quantile values for the thresholds t_m. Figures 4.11 to 4.16 display the class conditional CDFs (for class c_2) for the interactions from which the standardized distances are calculated.

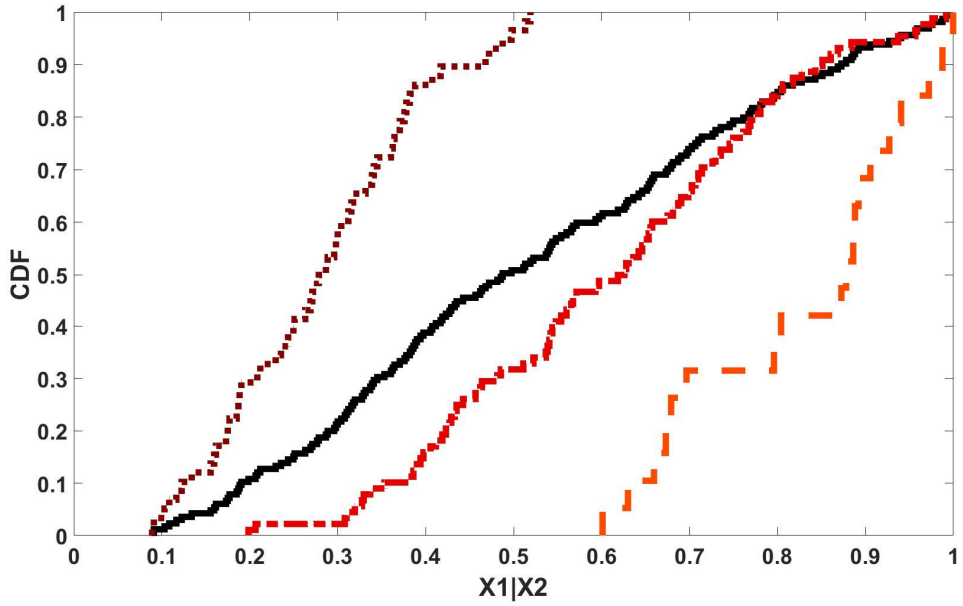


Figure 4.11: Class conditional CDF curves for interaction $X_1|X_2$ in polynomial function (cluster c_2)

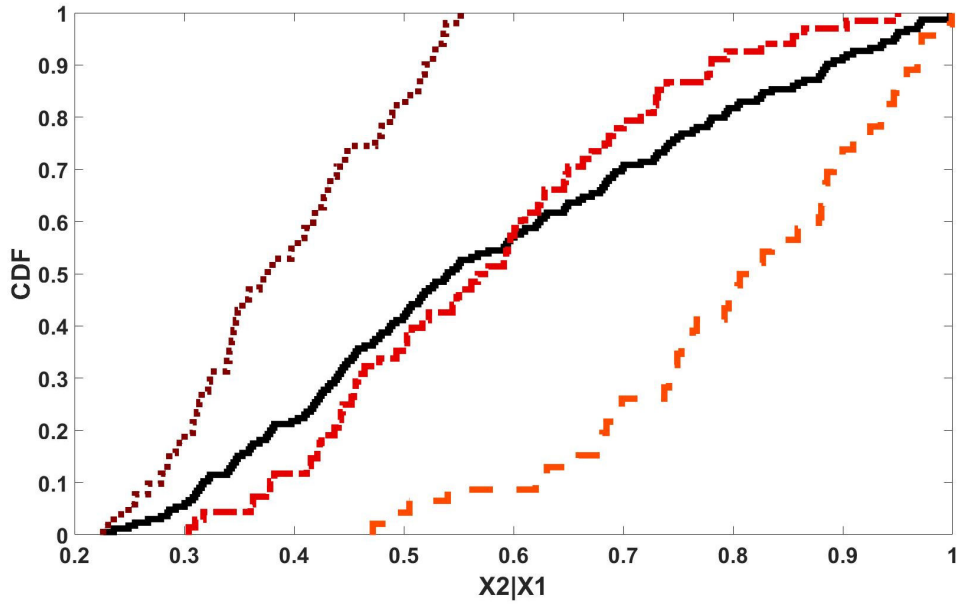


Figure 4.12: Class conditional CDF curves for interaction $X_2|X_1$ in polynomial function (cluster c_2)

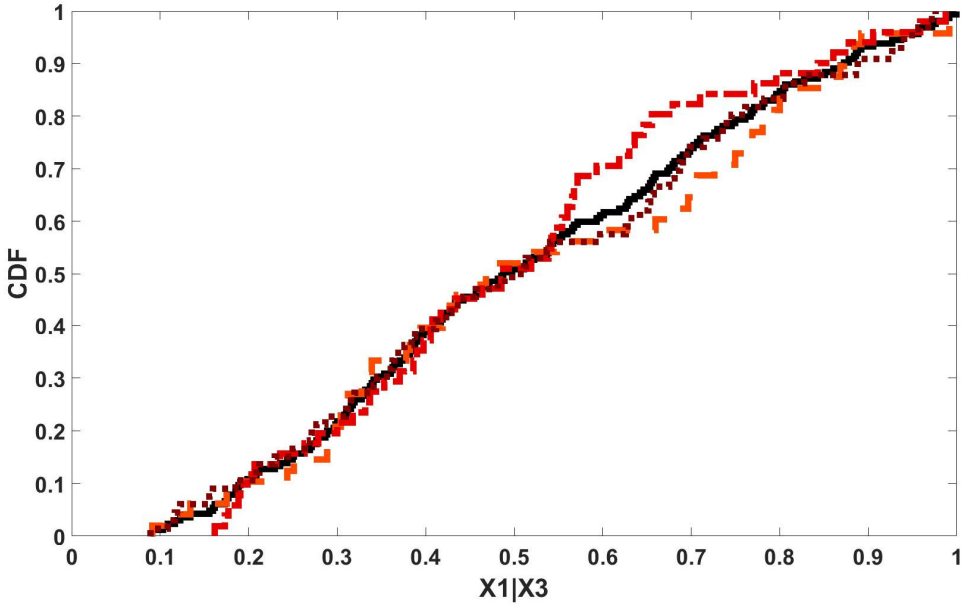


Figure 4.13: Class conditional CDF curves for interaction $X_1|X_3$ in polynomial function (cluster c_2)

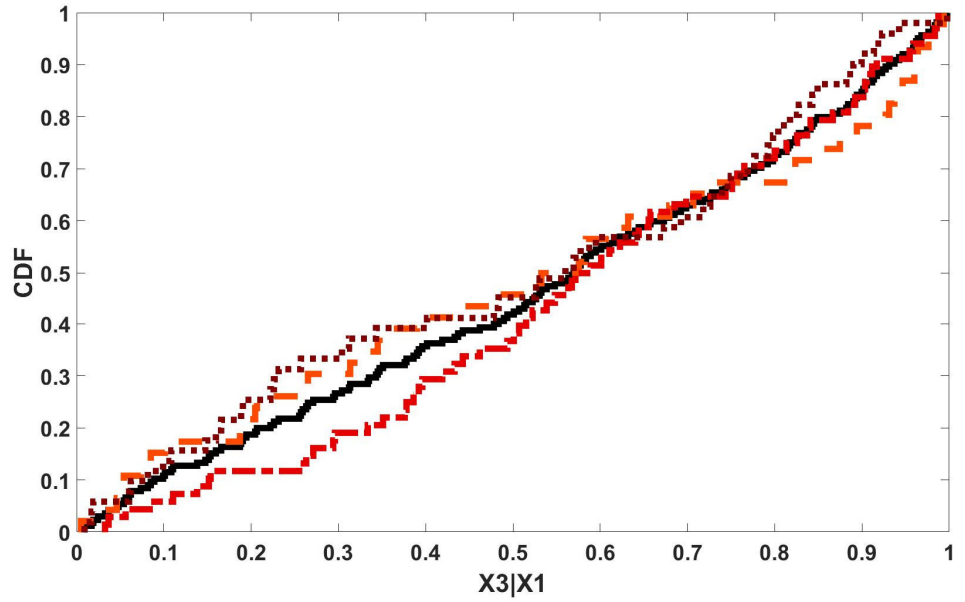


Figure 4.14: Class conditional CDF curves for interaction $X_3|X_1$ in polynomial function (cluster c_2)

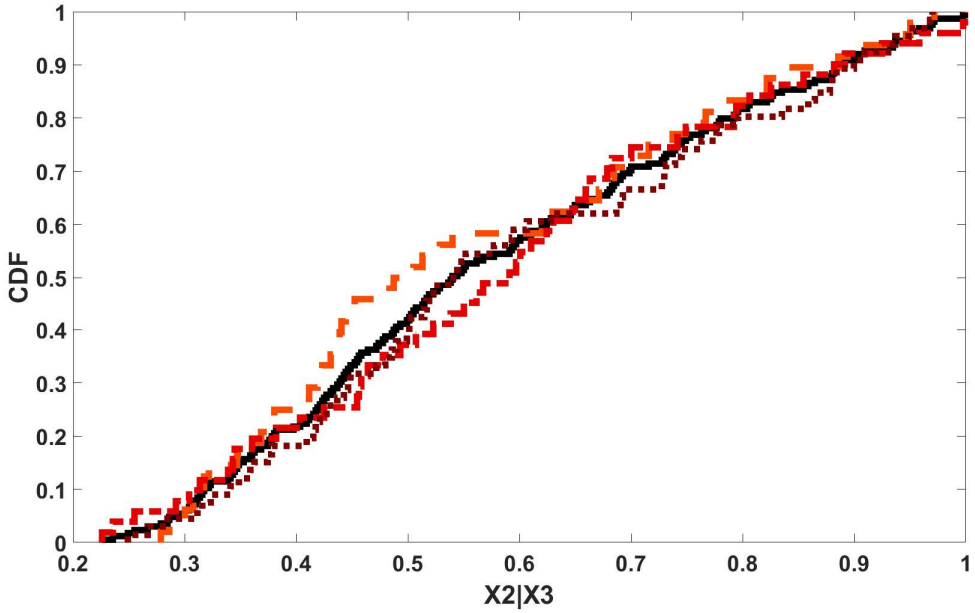


Figure 4.15: Class conditional CDF curves for interaction $X_2|X_3$ in polynomial function (cluster c_2)

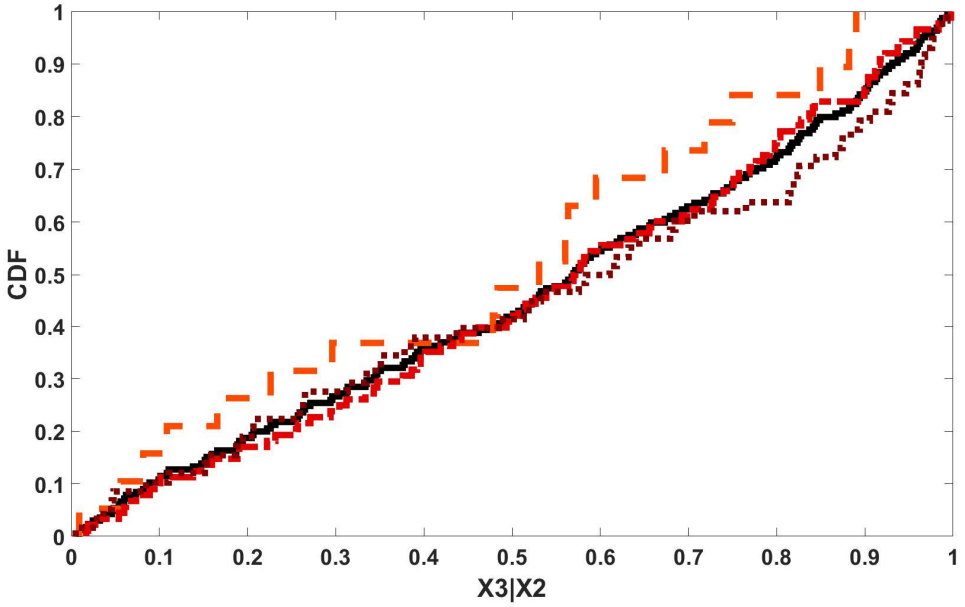


Figure 4.16: Class conditional CDF curves for interaction $X_3|X_2$ in polynomial function (cluster c_2)

By simply reviewing the above figures, it could be observed that the two way interaction between parameters X_1 and X_2 exist in the model. One possible way to display the results for analysis is to stack the bars (from Figures 4.11 to 4.16) belonging to a single cluster. Since the number of bins might not be the same for different parameters, each bar is standardized by the number of bins for the given interaction. As a consequence, the total length of the bar is $s_k(p_i|p_j)$ where each sub-bar (bars of different hue) has a length of $\frac{\hat{d}_{k,i|j,m}}{M}$. We can also display the measure of sensitivity averaged over all classes $s(p_i|p_j)$. Based on this, Figure 4.17 displays the CDF distance measure per class (red, blue, or green bars), $s_k(p_i|p_j)$ and the contribution $\frac{\hat{d}_{k,i|j,m}}{M}$ of each distance per threshold (segments within the red, blue and green bars).

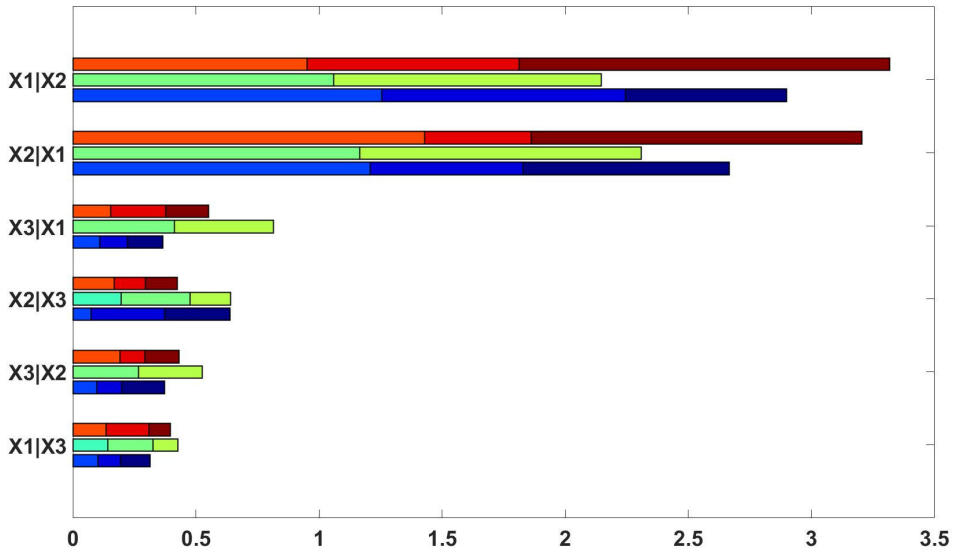


Figure 4.17: Two way conditional interactions and CDF distance measure per class for polynomial function

The average measure of interactions $s(p_i|p_j)$ is then displayed in Figure 4.18:

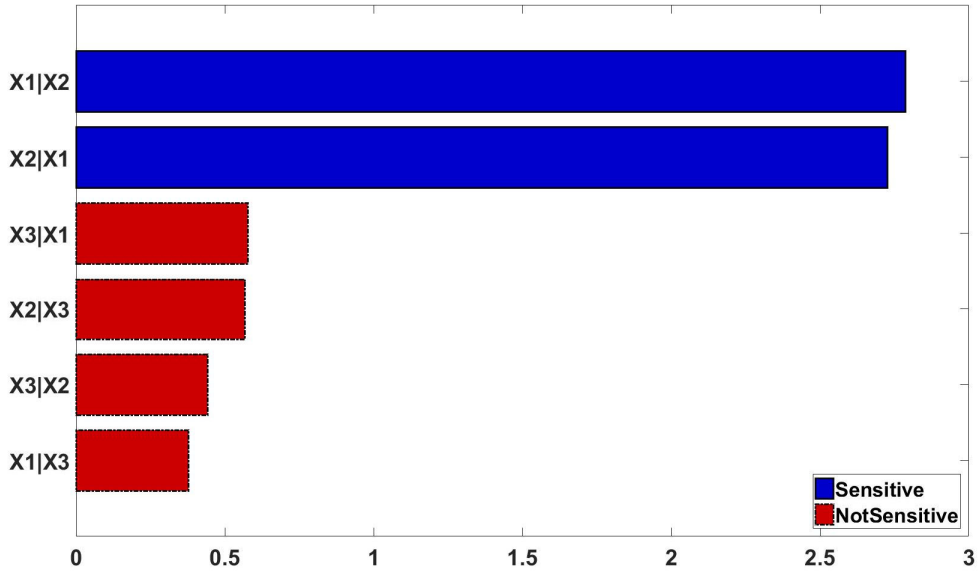


Figure 4.18: Average measure of sensitivity for two way conditional interactions of polynomial function

The interaction results are also as expected and show the interactions existing between parameters X_1 and X_2. The other two way interactions are negligible for this function.

4.6.2 Ishigami function

We learned about the Ishigami function from the previous chapters as a commonly used test function for sensitivity and uncertainty analysis that exhibits strong non-linearity and non-monotonicity. It is a very good function to test the reliability of the sensitivity analysis

methods since the analytical sensitivities are available and here we use the distance based sensitivity analysis approach to compare the results.

$$f(x) = Sin(X_1) + a\,Sin^2(X_2) + bX_3^4\,Sin(X_1) \qquad (4.27)$$

with $a = 7$ and $b = 0.1$.

Similar to the previous case of the polynomial function, as the first step, we perform a Latin Hyper Cube (LHC) sampling with 2048 experiments and compute the output of the Ishigami function. Next, the pairwise distances are calculated and the MDS space is constructed to apply the clustering method. Models are categorized in 3 clusters and the result is plotted in Figure 4.19:

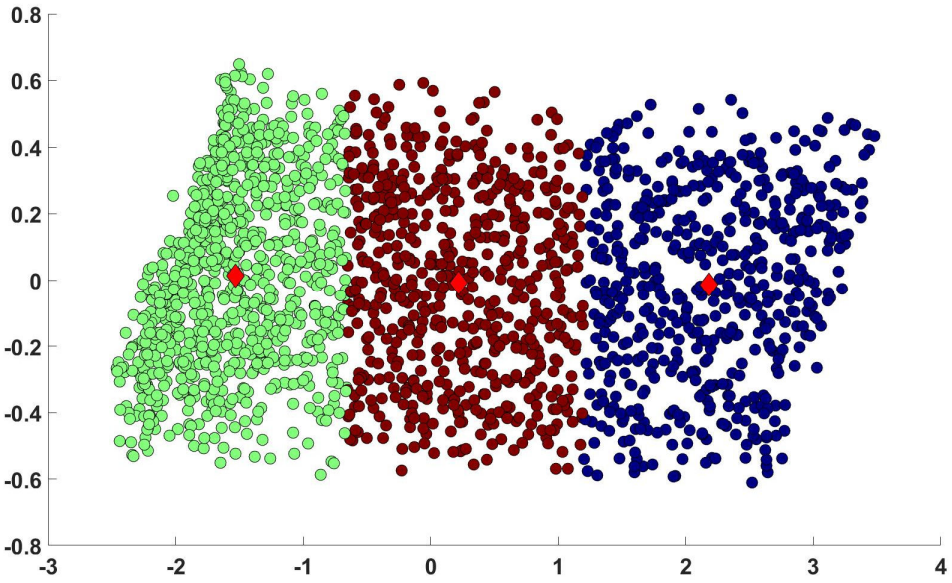


Figure 4.19: 2D clustering result for Ishigami function model ensemble

The following Figures 4.20, 4.21, and 4.22 show the class conditional CDFs as well as the CDF for the overall population for the 3 clusters:

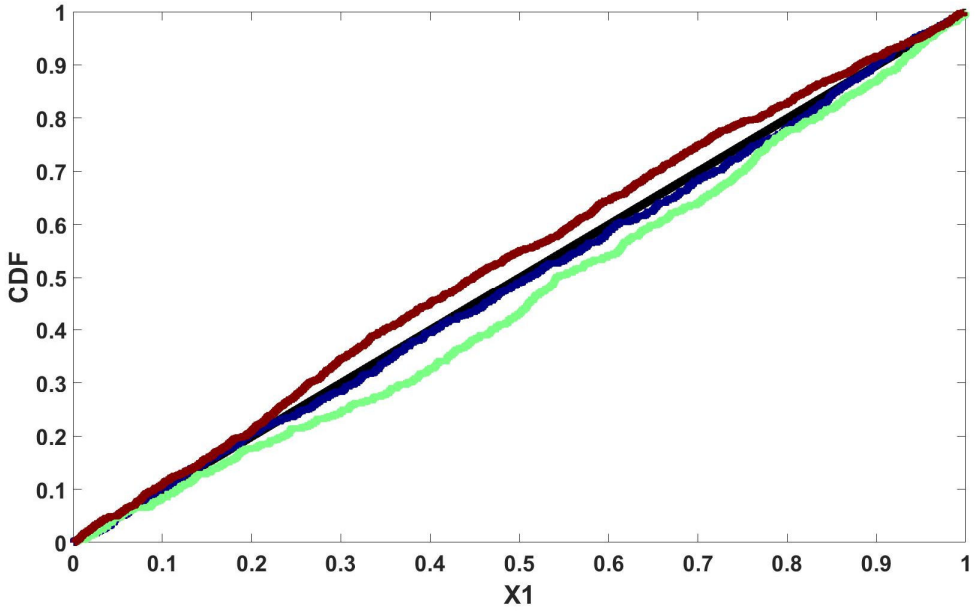


Figure 4.20: Empirical CDF curves for parameter X_1 in Ishigami function (LHC sampling)

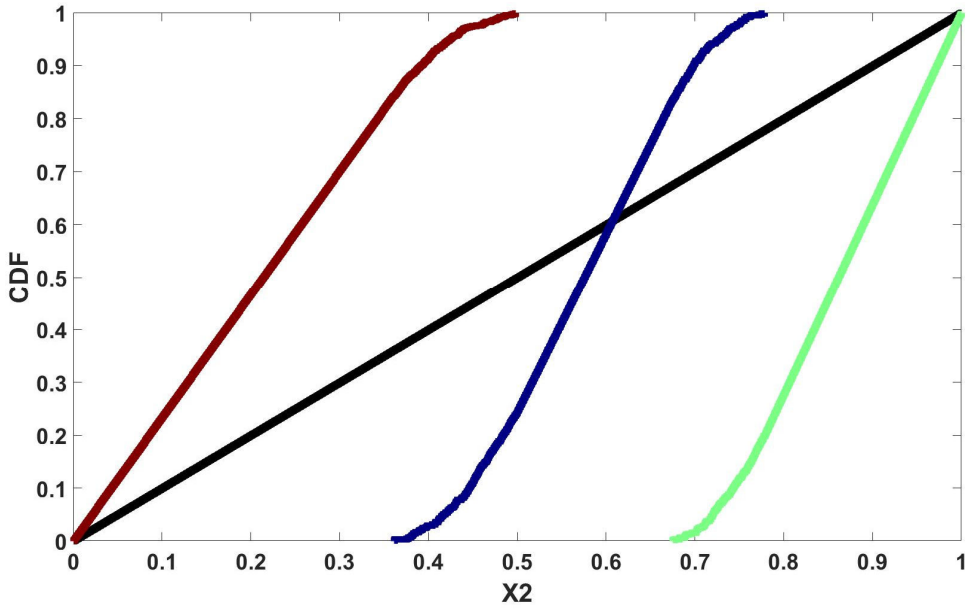


Figure 4.21: Empirical CDF curves for parameter X_2 in Ishigami function (LHC sampling)

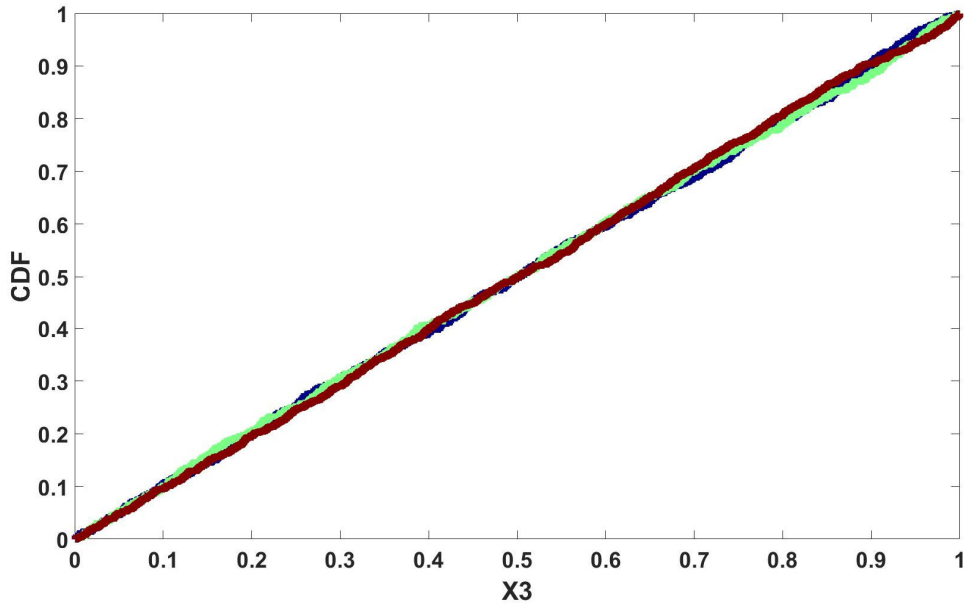


Figure 4.22: Empirical CDF curves for parameter X_3 in Ishigami function (LHC sampling)

We can display the standardized CDF distance calculated based on the previous step as well as the average measure of sensitivity (main effect) for each parameter in the following Figures 4.23 and 4.24:

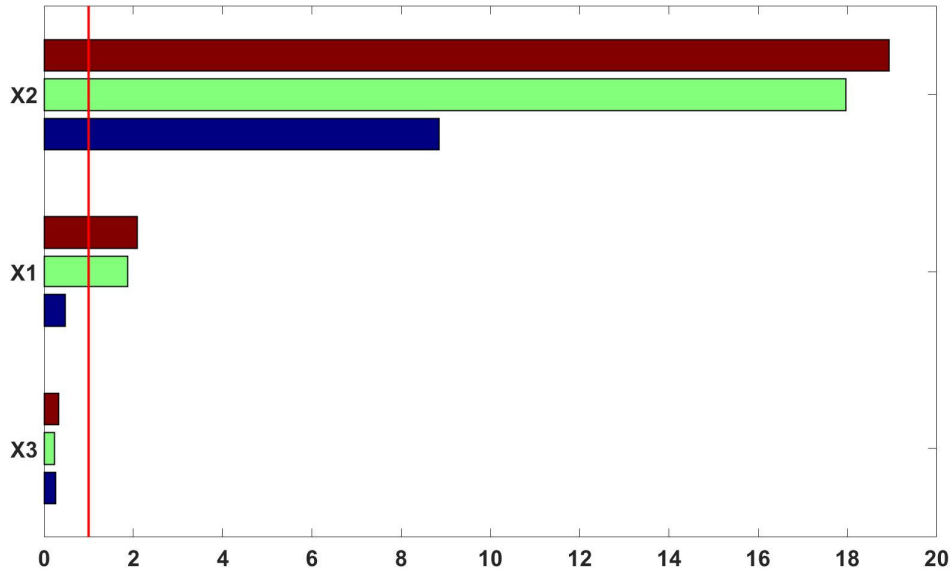


Figure 4.23: Standardized CDF distances for each cluster of Ishigami ensemble (LHC sampling)

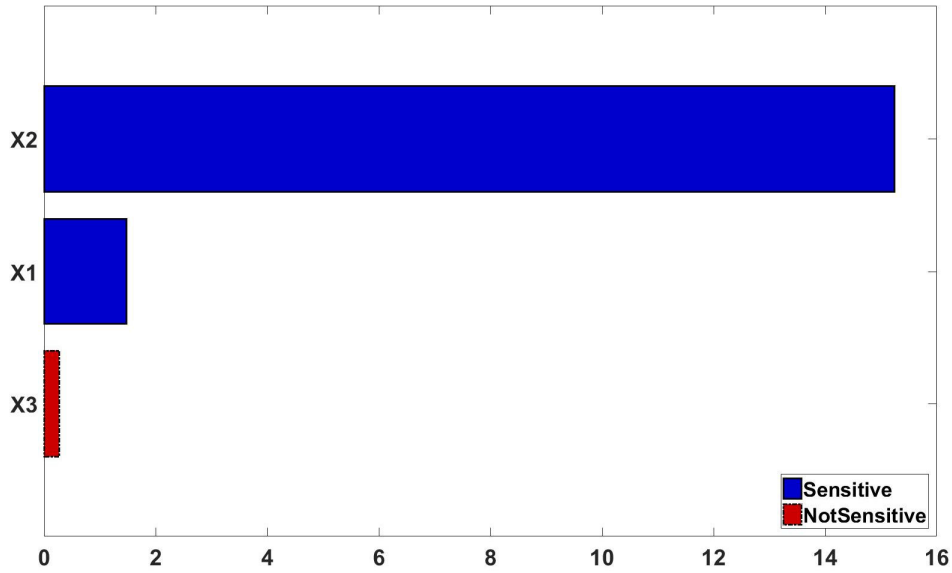


Figure 4.24: Average distance (sensitivity measure) for each parameter of Ishigami function (LHC sampling)

The analysis of the result shows the method is able to correctly identify parameter X_3 as non-influential, but the result for the other parameters are not exactly the true sensitivities. Here parameter X_2 is identified as the most important parameter by a large margin, where in fact we know from the analytical sensitivities as well the results of the Sobol method in the previous chapter that this is not the case. By looking back at the result, we see a peculiar behavior in Figure 4.21 where the CDF of parameter X_2 is calculated in each of the clusters and compared to the prior CDF. In the next section we try to investigate this and possibly address the issue.

4.6.2.1 Change of sampling method with Low discrepancy sequence

In our first try, we performed a Latin Hyper Cube (LHC) sampling in order to sample the input parameter space and subsequently calculate the responses. In doing so, we saw some skewed behavior for the class conditional CDFs of parameters X_2 which contributed to overestimation of its sensitivity in the model. Since we have the analytical sensitivities of the Ishigami function available in the literature, we know for a fact that this estimation was incorrect.

In the previous chapter we introduced low-discrepancy sampling sequence where we needed more orthogonality when trying to estimate the value of the Monte-Carlo integrals for Sobol method. Although using LHC ensures in many cases that the sampling ensemble of random numbers is representative of the real variability in the model, we observed that this was not enough in cases where the complexity of the estimation is high. Due to the highly nonlinear and non-monotone nature of the Ishigami function, we can assume that using the low discrepancy sequence may help improving the estimation of the distance based sensitivities measure. In order to verify this hypothesis, we repeat the procedure exactly as before but instead of LHC sampling, we use the low discrepancy sequence that we developed in the previous chapter and compare the results. The same number of experiments (2048) are generated and the class conditional CDFs are calculated and plotted in Figures 4.25 to 4.27 as follows:

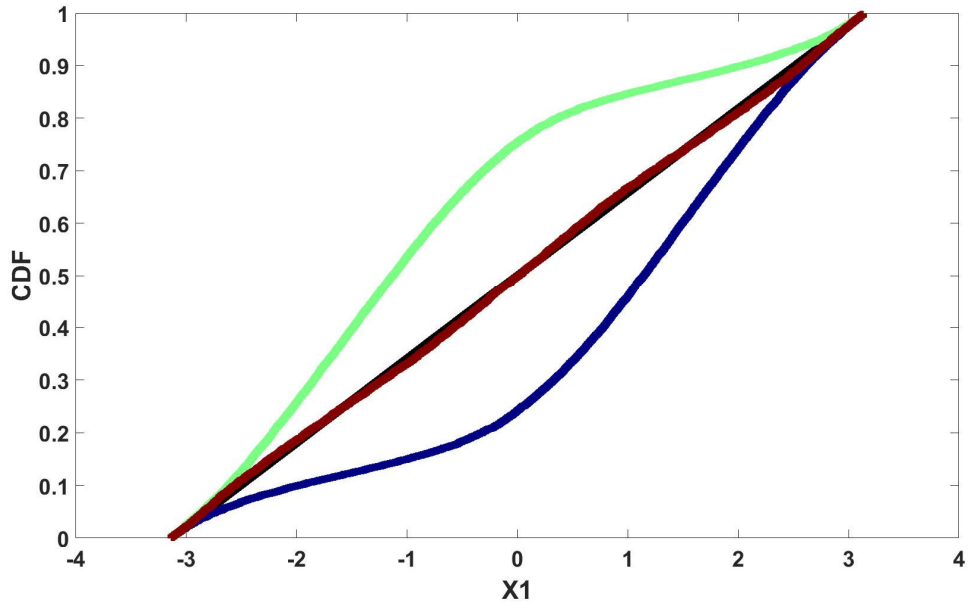


Figure 4.25: Empirical CDF curves for parameter X_1 in Ishigami function (low-discrepancy sampling)

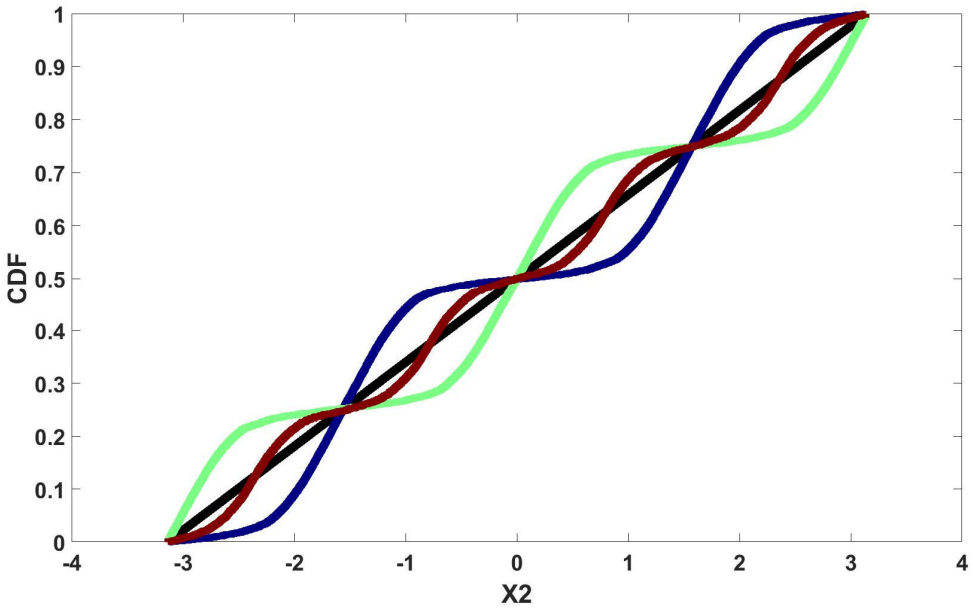


Figure 4.26: Empirical CDF curves for parameter X_2 in Ishigami function (low-discrepancy sampling)

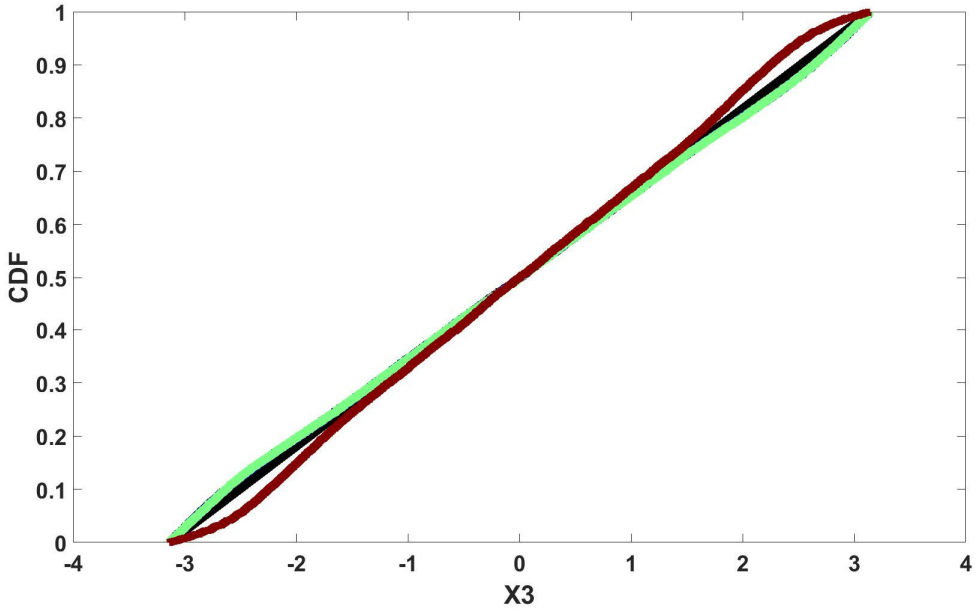


Figure 4.27: Empirical CDF curves for parameter X_3 in Ishigami function (low-discrepancy sampling)

We can see that we get a completely different set of CDFs for each cluster for parameter X_2 compared to before (Figures 4.20, 4.21, and 4.22). Calculating the standardized CDF distance and the average measure of sensitivity (main effect) for each parameter based on the new CDF values result in the following Figures 4.28 and 4.29:

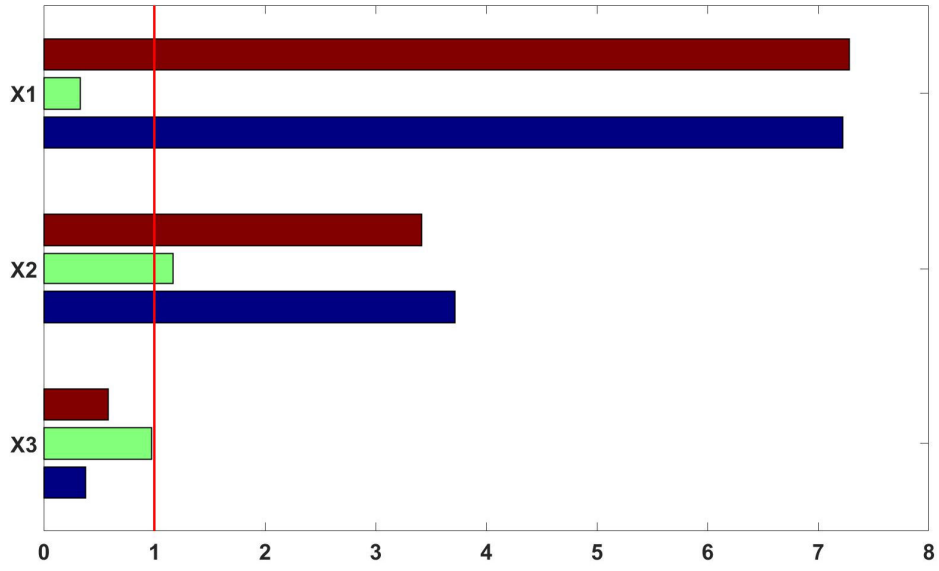


Figure 4.28: Standardized CDF distances for each cluster of Ishigami ensemble (low-discrepancy sampling)

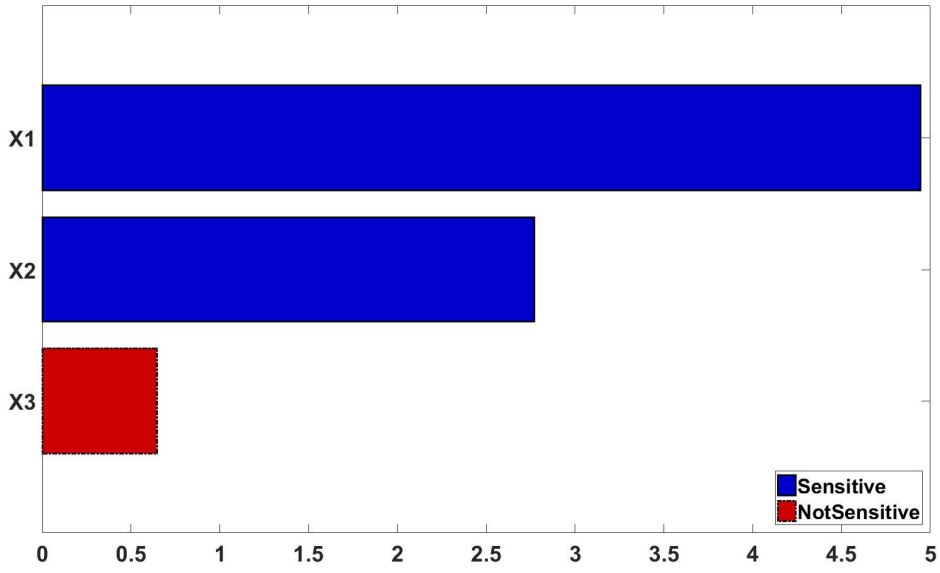


Figure 4.29: Average distance (sensitivity measure) for each parameter of Ishigami function (low-discrepancy sampling)

It can be observed that using the low discrepancy sequence sampling results in matching the expected true values for parameter sensitivities in the Ishigami function.

Now that we could match the expected sensitivities for the main effects, we evaluate how the distance based sensitivity analysis method can handle the parameter interactions.

4.6.2.2 Ishigami Interactions

By defining 3 bins for each parameter we create evenly spaced quantile values for the thresholds t_m and calculate the class conditional CDFs per all clusters from which the standardized distances are calculated. In the following graphs class conditional CDFs for pairwise interaction of $X_1|X_2$ and $X_2|X_1$ are displayed for cluster c_1 (Figures 4.30 and 4.31), $X_2|X_3$ and $X_3|X_2$ for cluster c_2 (Figures 4.32 and 4.33), and $X_1|X_3$ and $X_3|X_1$ are displayed for cluster c_3 (Figures 4.34 and 4.35).

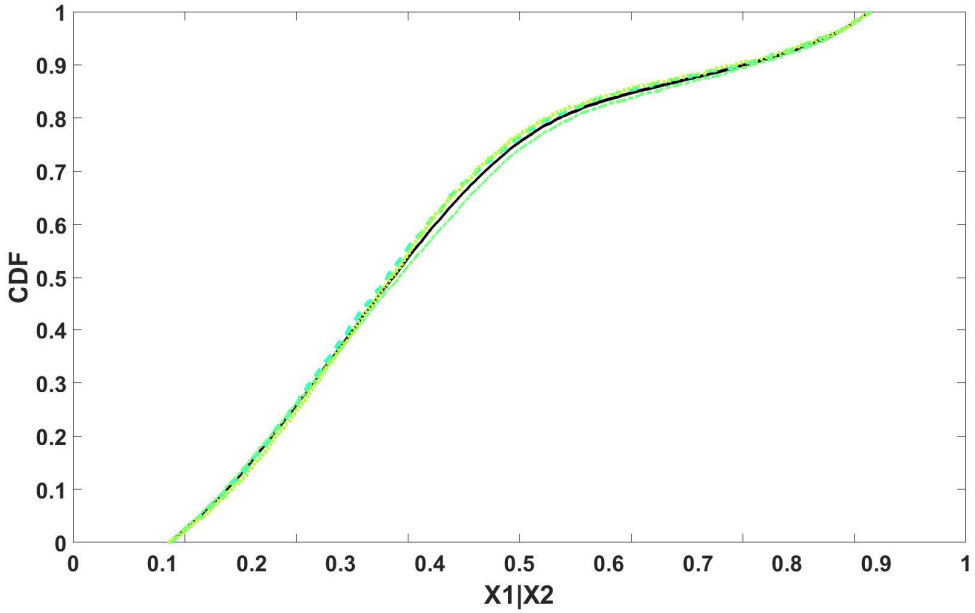


Figure 4.30: Class conditional CDF curves for interaction $X_1|X_2$ in Ishigami function (cluster c_1)

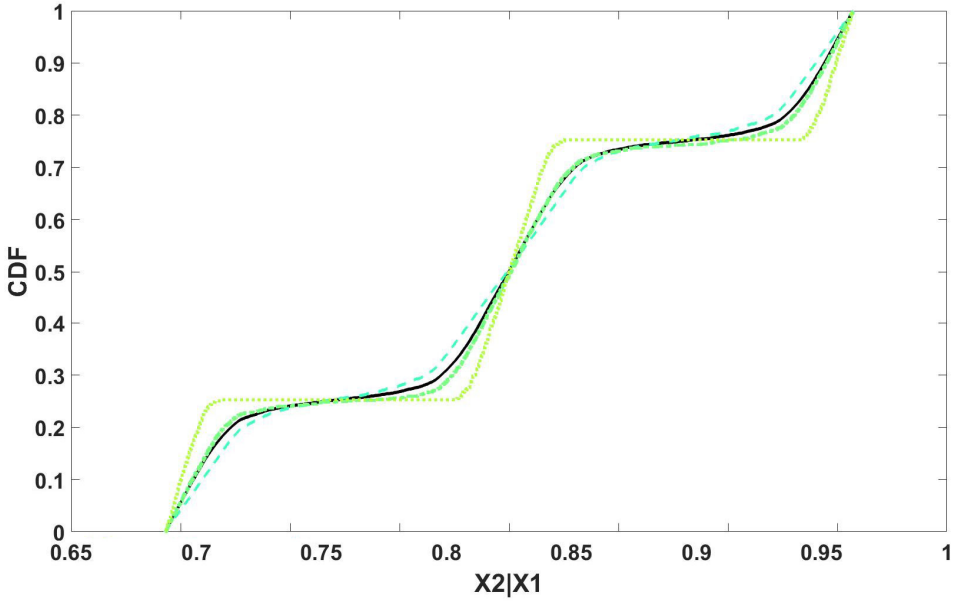


Figure 4.31: Class conditional CDF curves for interaction $X_2|X_1$ in Ishigami function (cluster c_1)

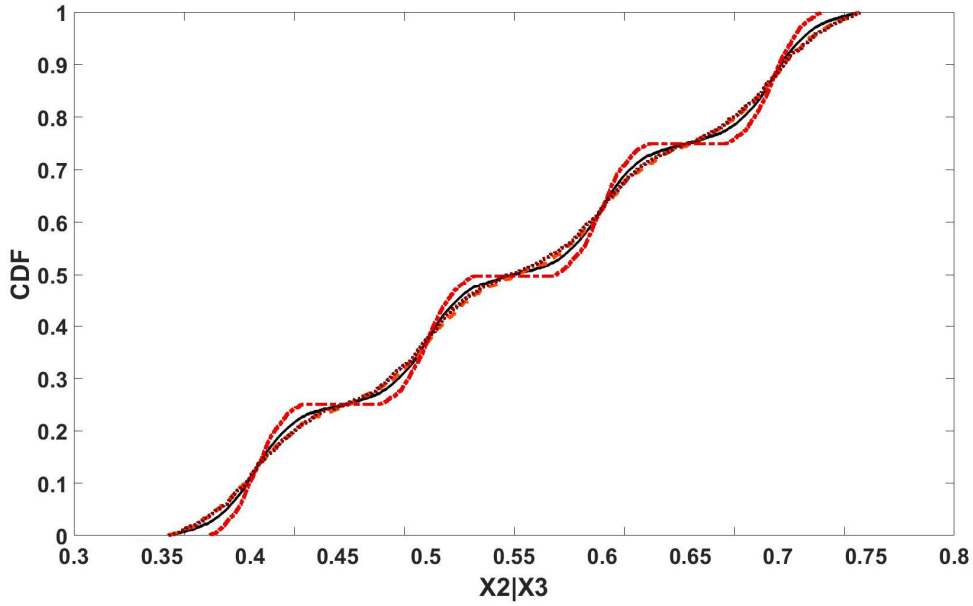


Figure 4.32: Class conditional CDF curves for interaction $X_2|X_3$ in Ishigami function (cluster c_2)

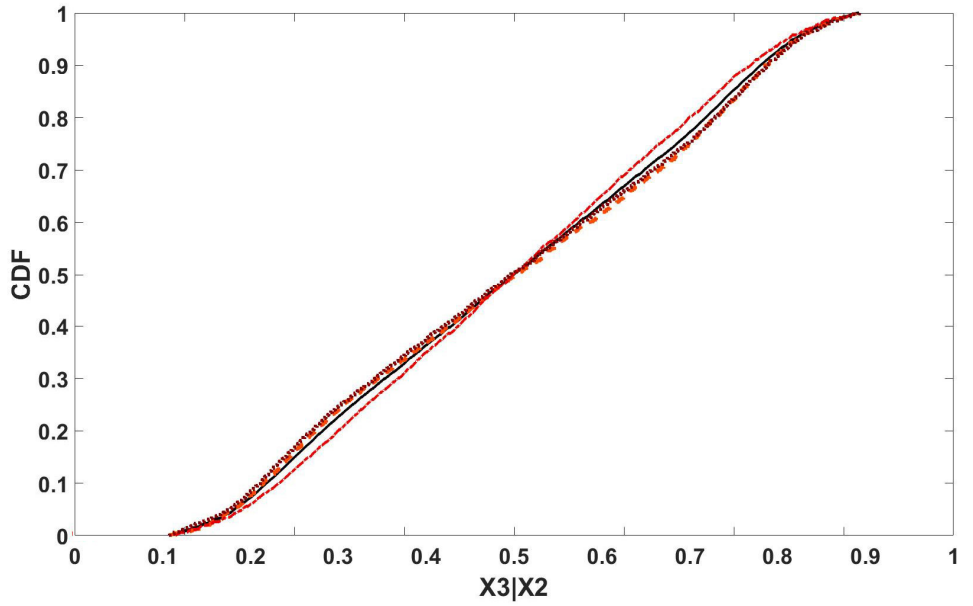


Figure 4.33: Class conditional CDF curves for interaction $X_3|X_2$ in Ishigami function (cluster c_2)

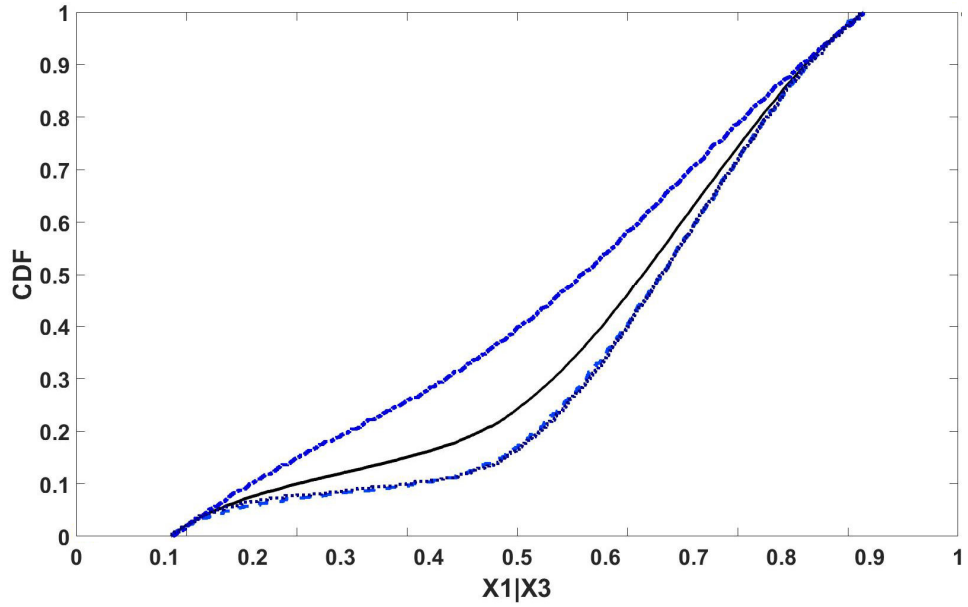


Figure 4.34: Class conditional CDF curves for interaction $X_1|X_3$ in Ishigami function (cluster c_3)

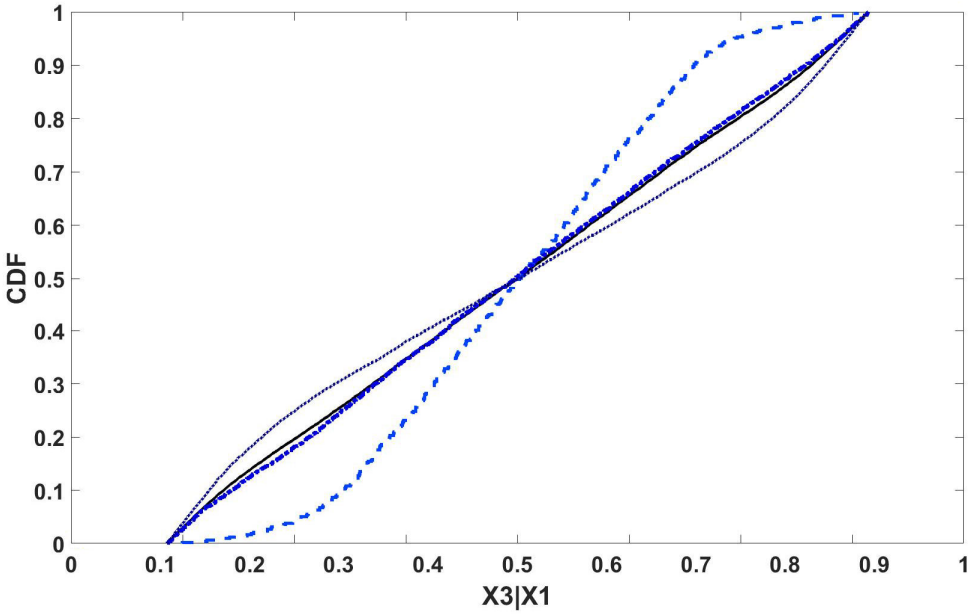


Figure 4.35: Class conditional CDF curves for interaction $X_3|X_1$ in Ishigami function (cluster c_3)

The CDF distance measures per cluster and average measure of interactions are plotted next in Figures 4.36 and 4.37:

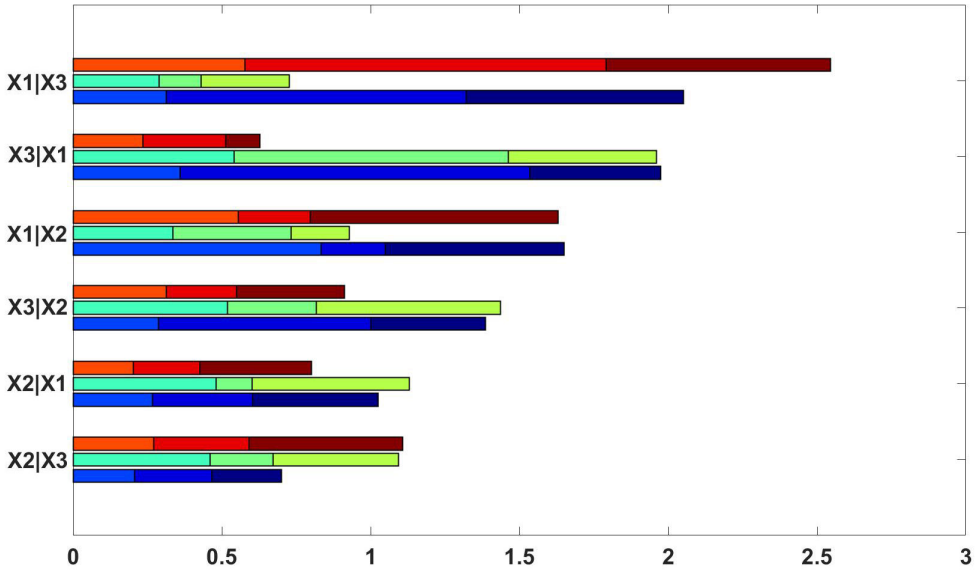


Figure 4.36: Two way conditional interactions and CDF distance measure per class for Ishigami function

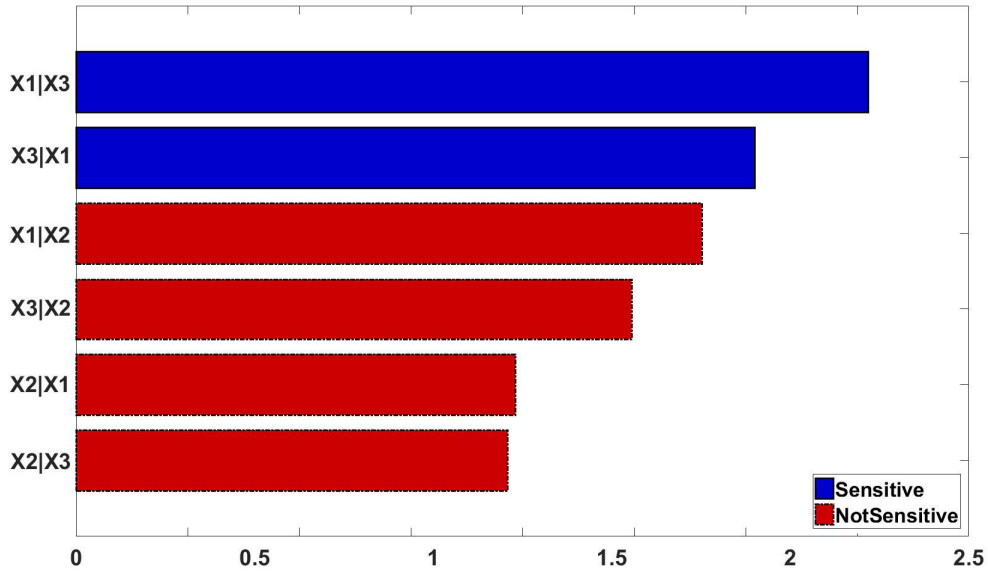


Figure 4.37: Average measure of sensitivity for two way conditional interactions of Ishigami function

The interaction results are also as expected and show pairwise interactions between $X_1|X_3$ and $X_3|X_1$which proves the robustness of the method. This is important because parameter X_3 in itself is a non-influential input but its pairwise interaction together with parameter X_1 contributes to the variability of the response.

4.6.3 Sobol G function

Sobol G function is another common test function for numerical estimation and sensitivity analysis methods that we used in the previous chapters to evaluate the developed sensitivity analysis methods.

$$f(X) = \prod_{i=1}^{d} \frac{|4X_i - 2| + a_i}{1 + a_i} \qquad (4.28)$$

Where the values of a_i coefficients displayed in the Table 4.1 are used.

Table 4.1: a_i values for Sobol G function

a_i	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}
Value	0.001	89.9	5.54	42.10	0.78	1.26	0.04	0.79	74.51	4.32	82.51	41.62

The sensitivity indices for Sobol G function are expressed analytically and we know the lower the value of a_i is, the more important parameter X_i becomes. So in terms of parameter importance, we expect a ranking of $X_1 > X_7 > X_5 > X_8 > \ldots$ and so on.

Applying the MDS procedure and k-means clustering for 3 clusters yields in the following clustering of responses in Figure 4.38. In this case we have generated 2048 experiments using LHC sampling.

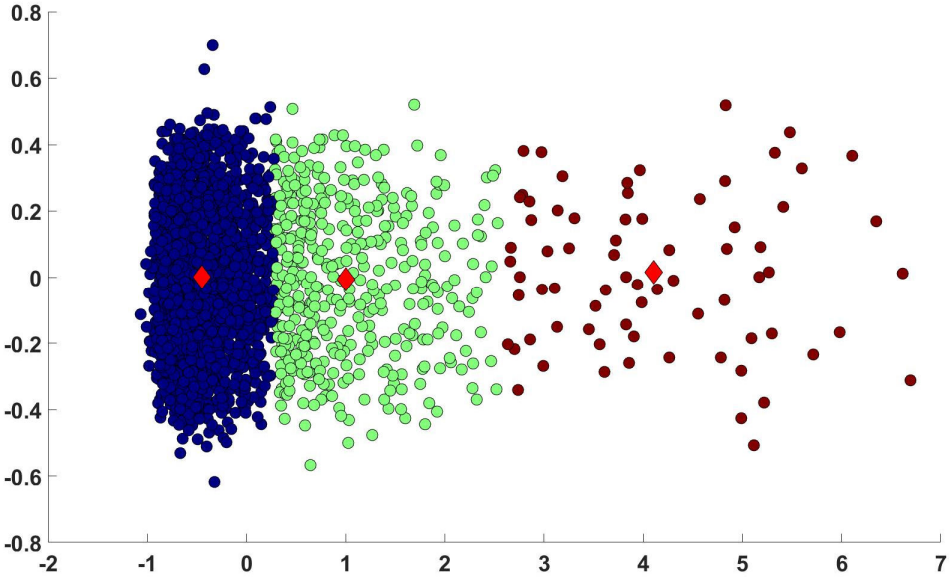
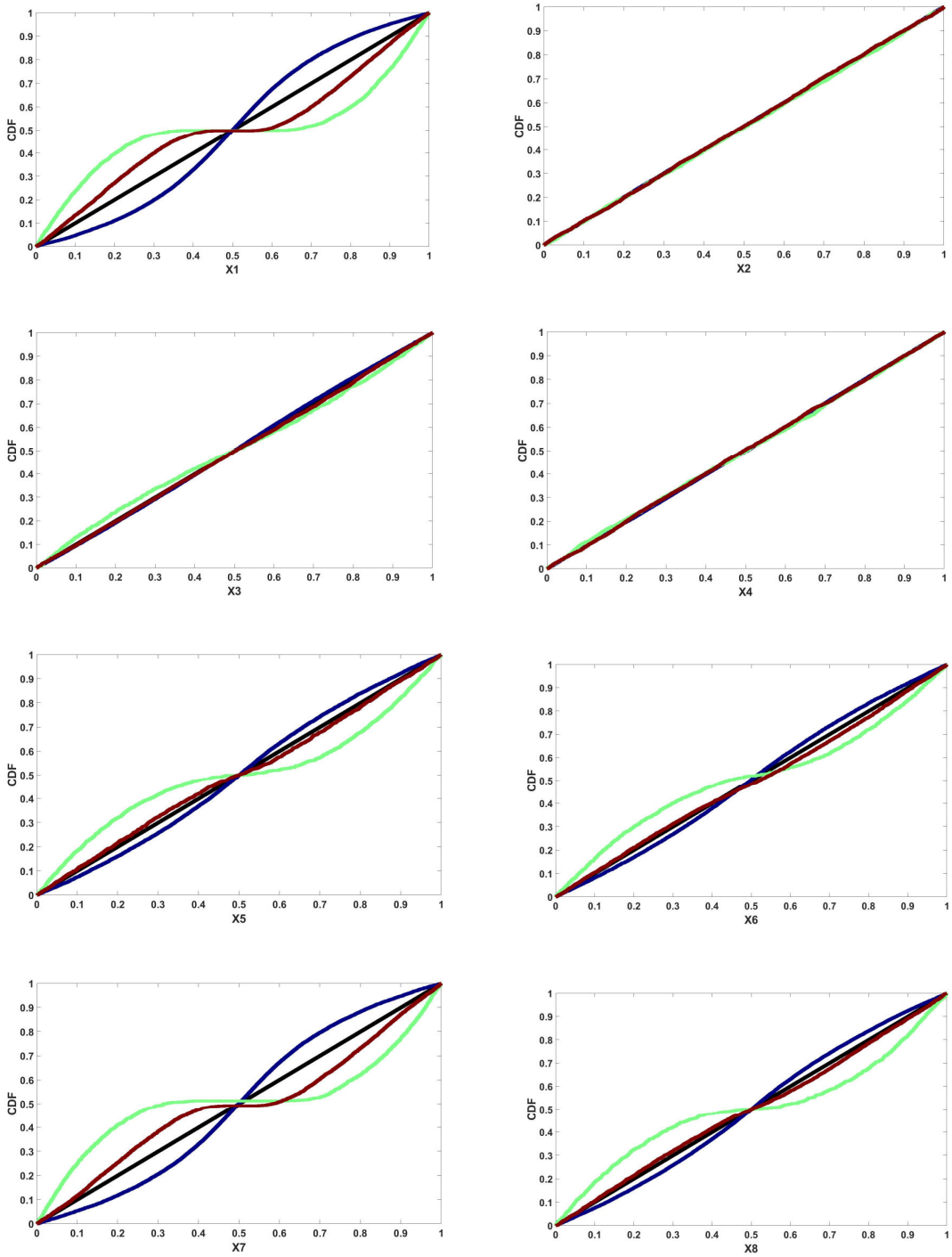


Figure 4.38: 2D clustering result for Sobol G function model ensemble

The class conditional CDFs for parameters X_1 to X_{12} are shown in Figure 4.39 as follows:


CDF
X1
CDF
X2
CDF
X3
CDF
X4
CDF
X5
CDF
X6
CDF
X7
CDF
X8

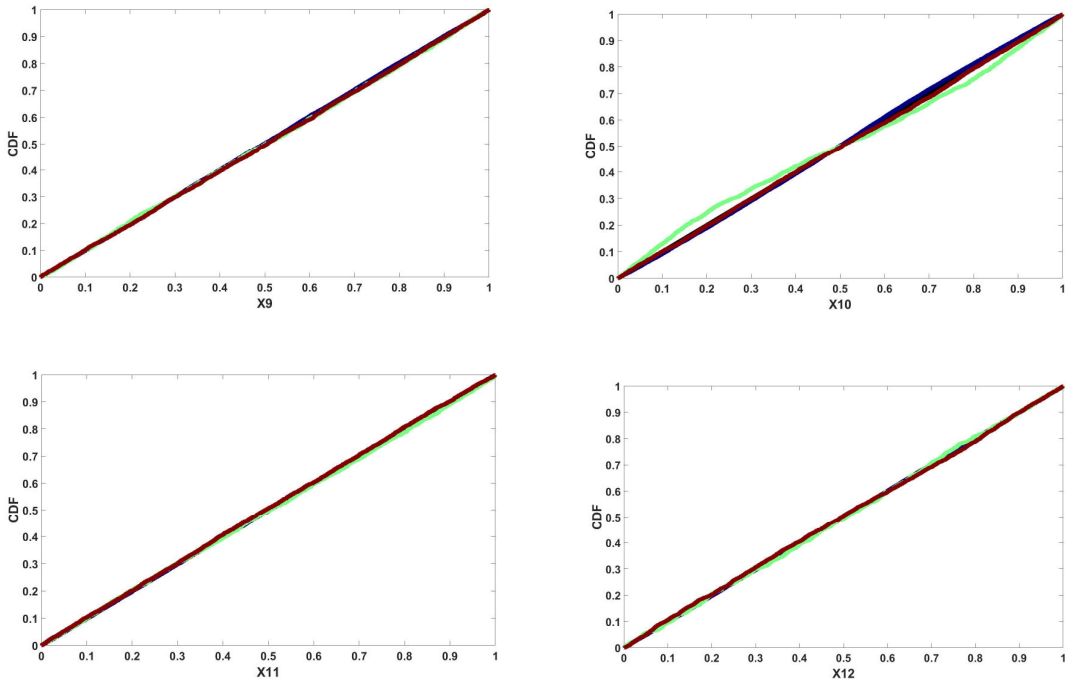


Figure 4.39: Empirical CDF curves for parameter X_1 to X_{12} in Sobol G function

Analyzing the plots shows the largest distance between the class conditional CDFs of the parameters X_1 and X_7 and the total population CDF. This is the indication of the impact of these parameters based on the selected distance.

4.6.3.1 Sobol G Main Effects

Using the standardized CDF distance calculated from the CDFs in Fig. 4.39 for each class, again, if the difference is statistically relevant, as we observed in the previous figures for parameters X_1 and X_7 for example, the parameter is deemed sensitive and a bar with a value greater than one (normalized difference in CDF) is displayed. If the parameter is insensitive

the bar length will be less than one as it is the case for parameters X_2, X_4, X_9, etc. We can check that for this function in Figure 4.40:

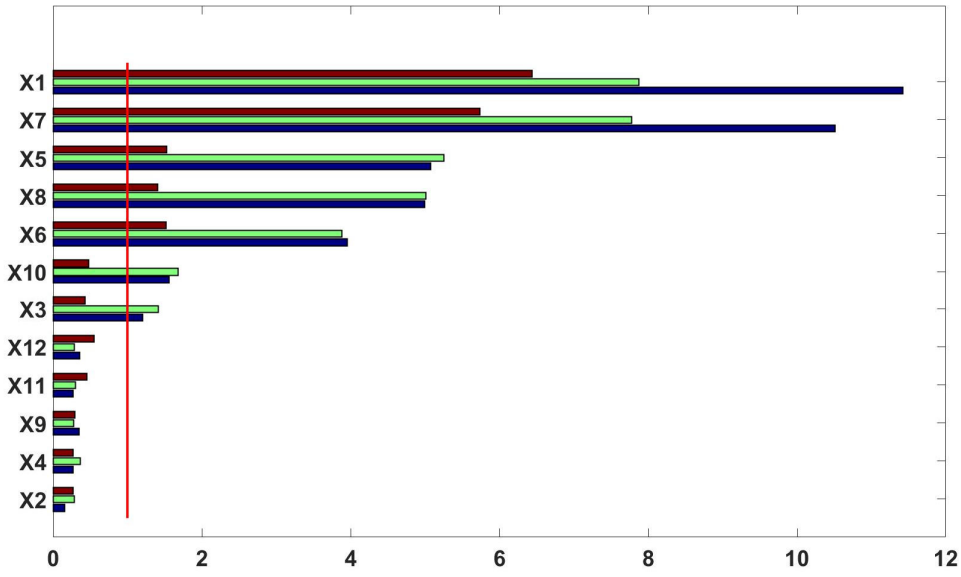


Figure 4.40: Standardized CDF distances for each cluster of Sobol G function

The standardized measure of sensitivity (the average distance $s(p_i)$ for each parameter) is displayed in the following Figure 4.41:

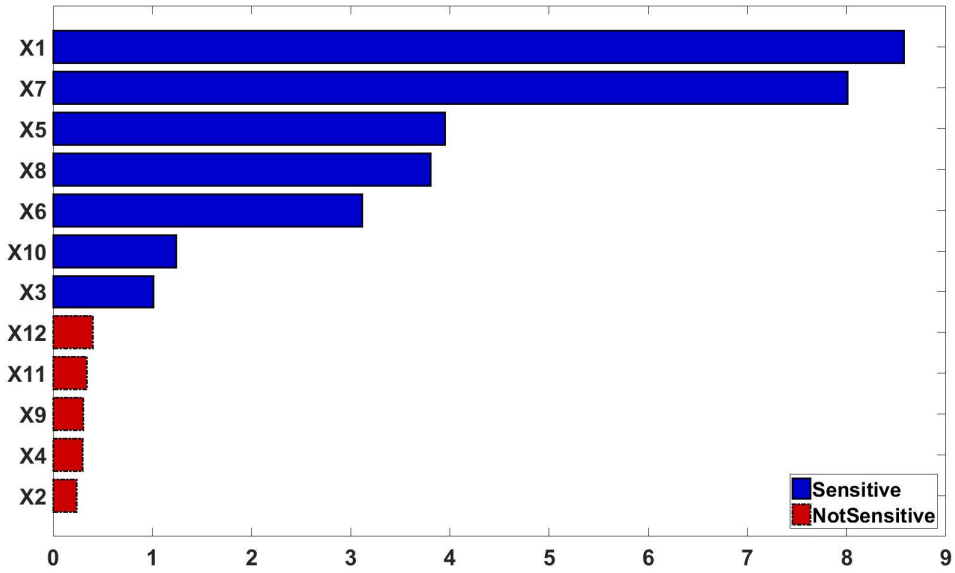


Figure 4.41: Average distance (sensitivity measure) for each parameter of Sobol G function

The order of parameter importance from the above graphs matches the parameter ranking obtained from the available analytic sensitivities of the Sobol function in the literature and also equate to the results of the previous chapters. This confirms the method could now be applied to real reservoir modeling cases for analysis.

4.6.4 CSS pilot

In the previous chapters, we studied a single well cyclic steam stimulation (CSS) pilot and tried to determine the importance of reservoir and operating variables, for later history

matching the model so that it can be used to develop an optimum ongoing strategy for future wells. The CSS model consisted of 9 uncertain parameters and 3 objective functions. In chapter 3, since the CPU cost of running Sobol sensitivity analysis method directly on the model was prohibitively high, we created a LHC design with 642 experiments in order to train a surrogate model and run the SA method through this approximation of our model. Now, we utilize this created model ensemble from before for use in the distance based approach introduced in this chapter.

4.6.4.1 Cumulative Oil Production objective function

The time series plot of these 642 runs for our first objective function, Cumulative Oil Production is presented in Figure 4.42.

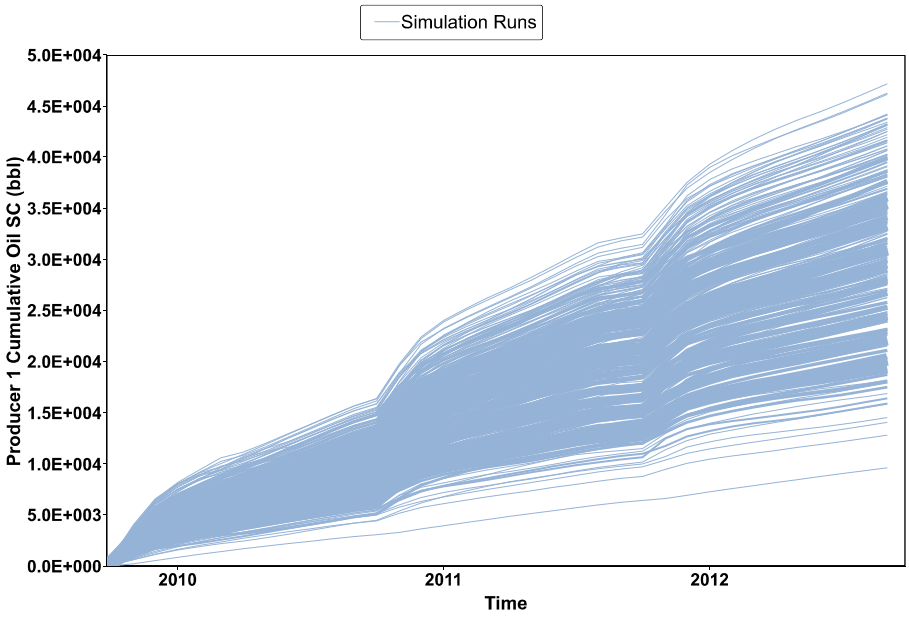


Figure 4.42: Time series model ensemble for the producer Cumulative Oil SC

The next step is to calculate the pairwise distances for Cumulative Oil SC of our producer well for each time step between each two pairs of models. Once the pairwise distances are available, we construct the dissimilarity matrix and perform the multidimensional scaling for 2 dimensions. Then, the clustering algorithm is applied to group the runs in 3 clusters and find the medoids. The clustering results are plotted in Figure 4.43.

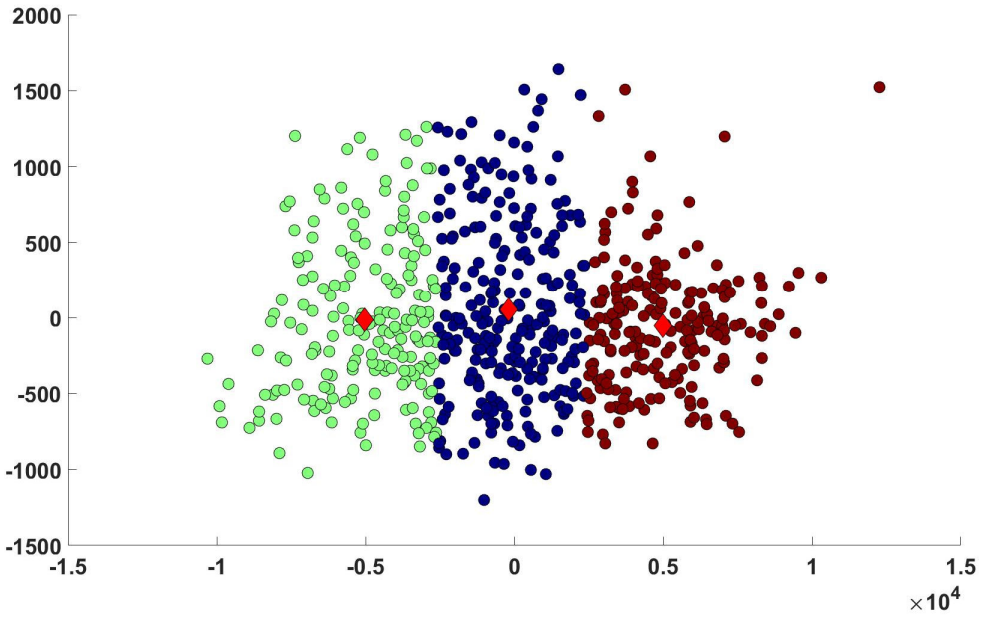


Figure 4.43: 2D clustering result for CSS pilot Cumulative Oil Production model ensemble

The selected 3 medoids by the clustering algorithm are highlighted within the model ensemble in Figure 4.44.

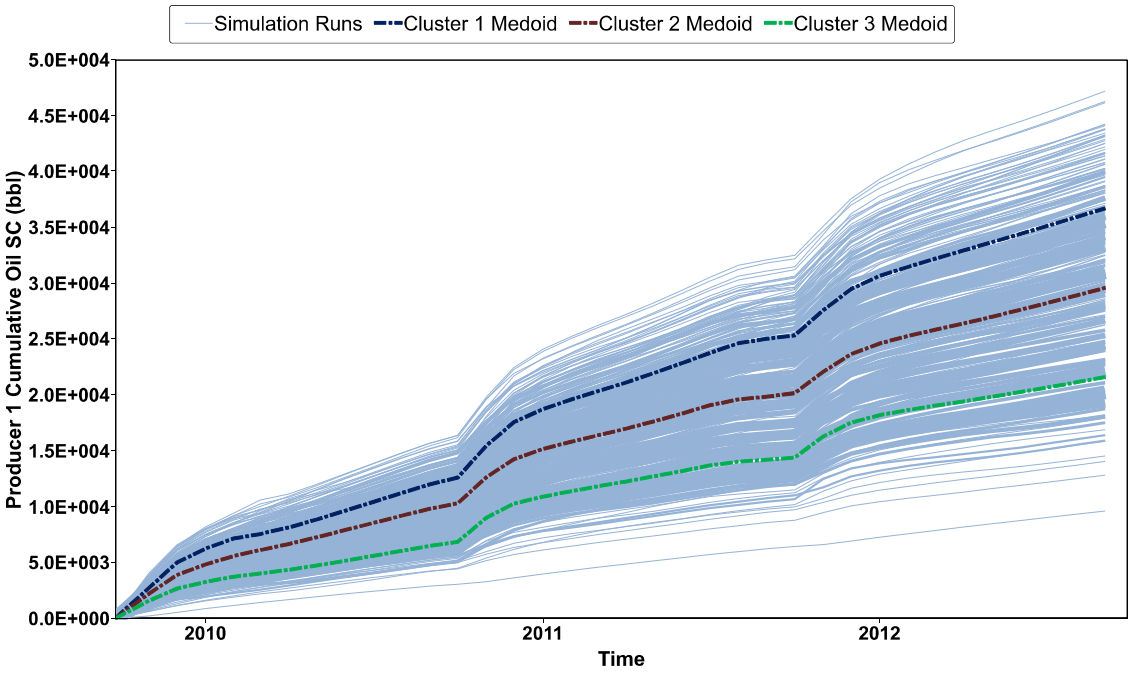


Figure 4.44: 3 cluster medoids for CSS pilot Cumulative Oil SC model ensemble

Now that we have the clusters available, the CDF of each input parameter in each cluster is compared to the distribution of the entire population. These CDF plots are available for verification in Appendix B.

Based on the results of this stage, the CDF distance measure per cluster (red, blue, or green bars) is calculated for each parameter and the results are displayed in the Figure 4.45.

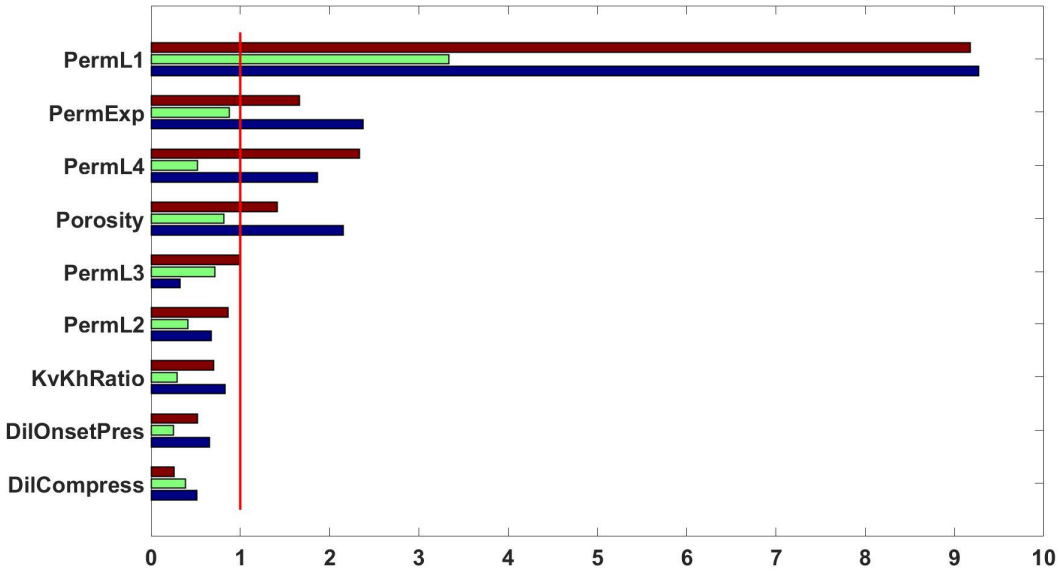


Figure 4.45: Standardized CDF distances for each cluster for Cumulative Oil Production objective function

This plot, as well as Figure 4.46 that displays the average distance measure for each parameter, provide the final ranking of inputs in terms of their main effect contribution to the objective function uncertainty. As it can be observed, the results are in very good agreement with the Sobol SA method presented in chapter 3.

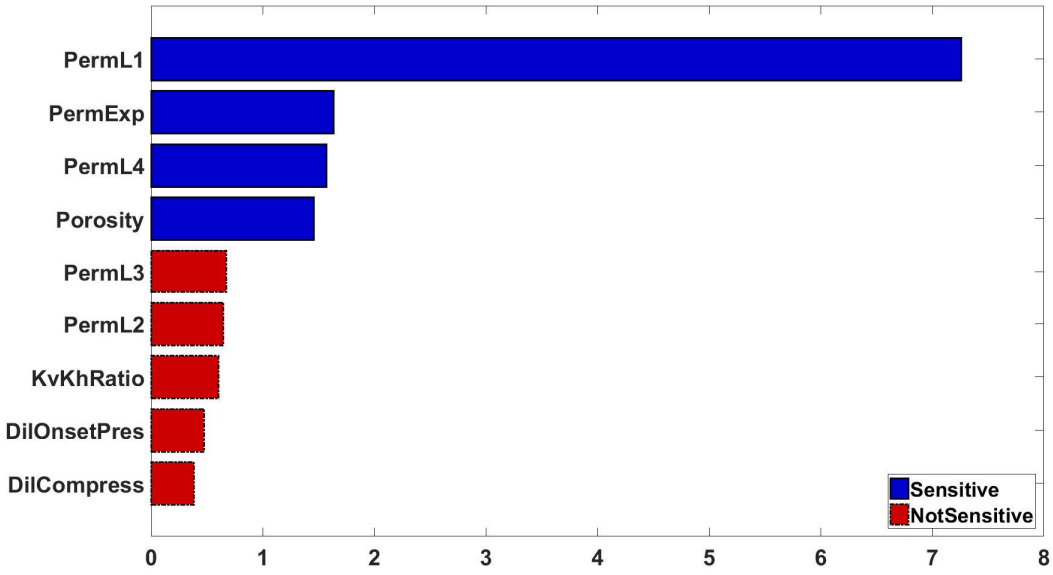


Figure 4.46: Average distance (sensitivity measure) for each parameter of Cumulative Oil Production objective function

4.6.4.2 Cumulative Water Production objective function

The same procedure is repeated for Cumulative Water Production objective function. The following model ensemble is used to obtain the clustering in Figure 4.47.

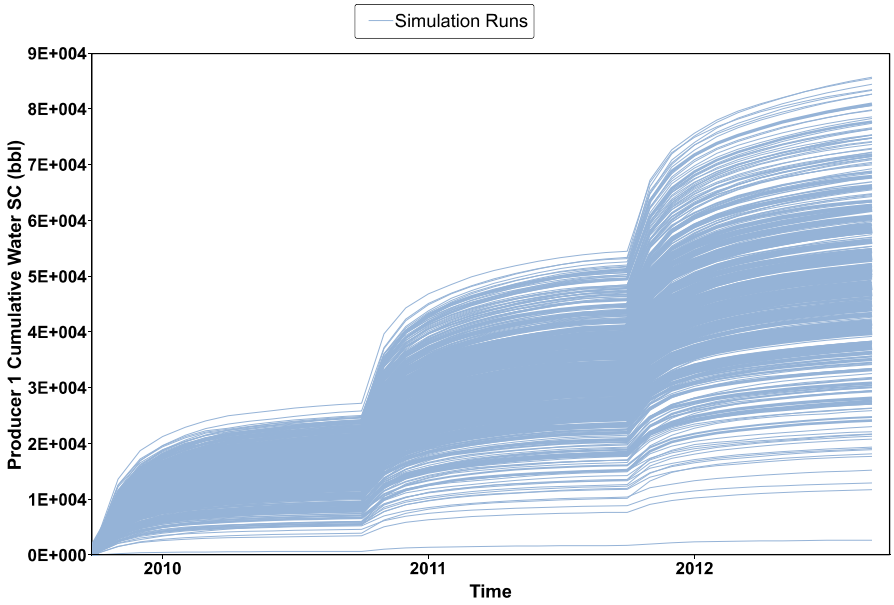


Figure 4.47: Time series model ensemble for the producer Cumulative Water SC

Figure 4.48 displays the result of clustering the model ensemble in 3 classes:

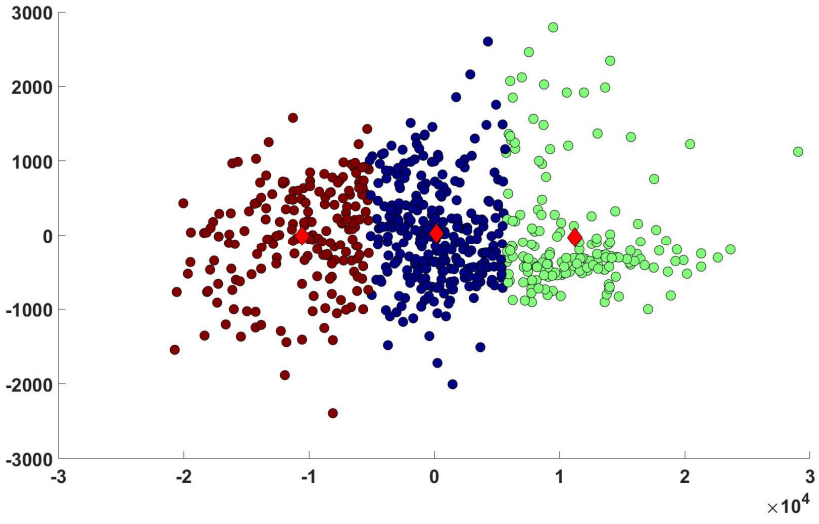


Figure 4.48: 2D clustering result for CSS pilot Cumulative Water Production model ensemble

The cluster medoids are highlighted within the ensemble in Figure 4.49 and the CDF distances per cluster per parameter are calculated. The plots are available in Appendix B.

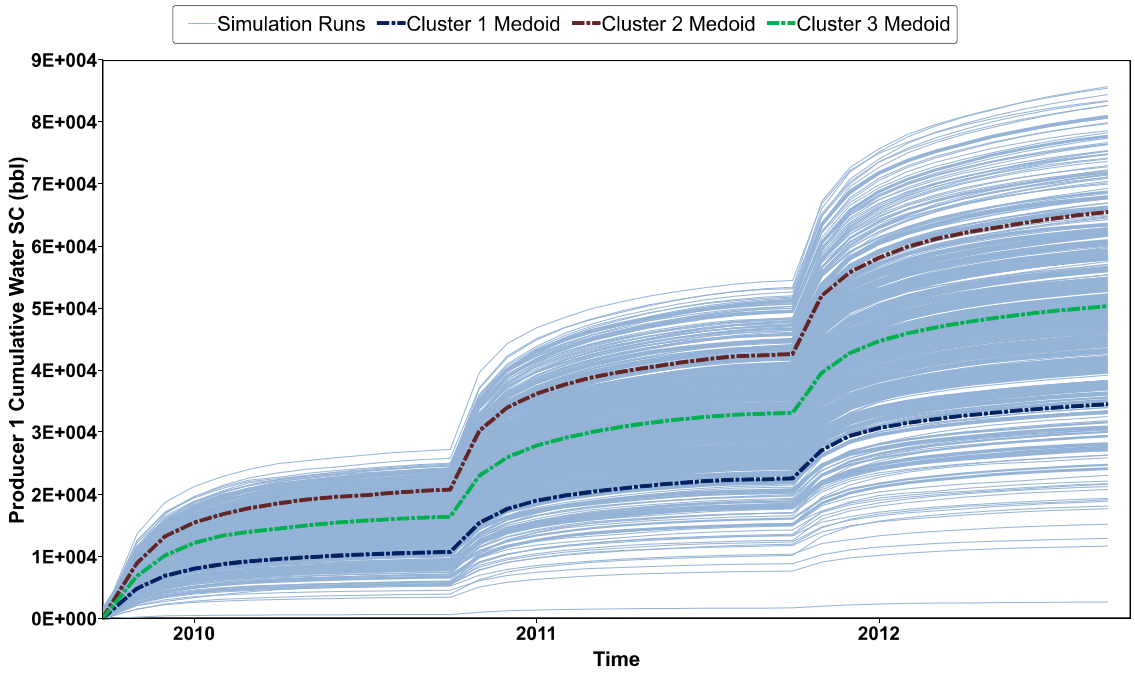


Figure 4.49: 3 cluster medoids for CSS pilot Cumulative Water SC model ensemble

Using the standardized CDF distance for each class and the average distance for each parameter (Figures 4.50 and 4.51), we get the parameter ranking for Cumulative Water Production objective function. Once again the results match the much more computationally expensive Sobol SA method.

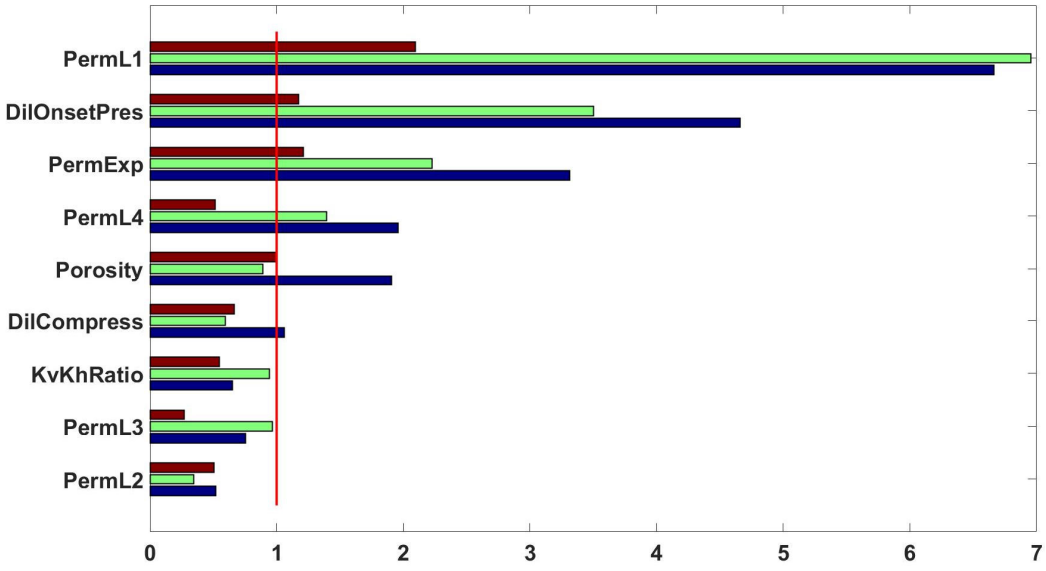


Figure 4.50: Standardized CDF distances for each cluster for Cumulative Water Production objective function

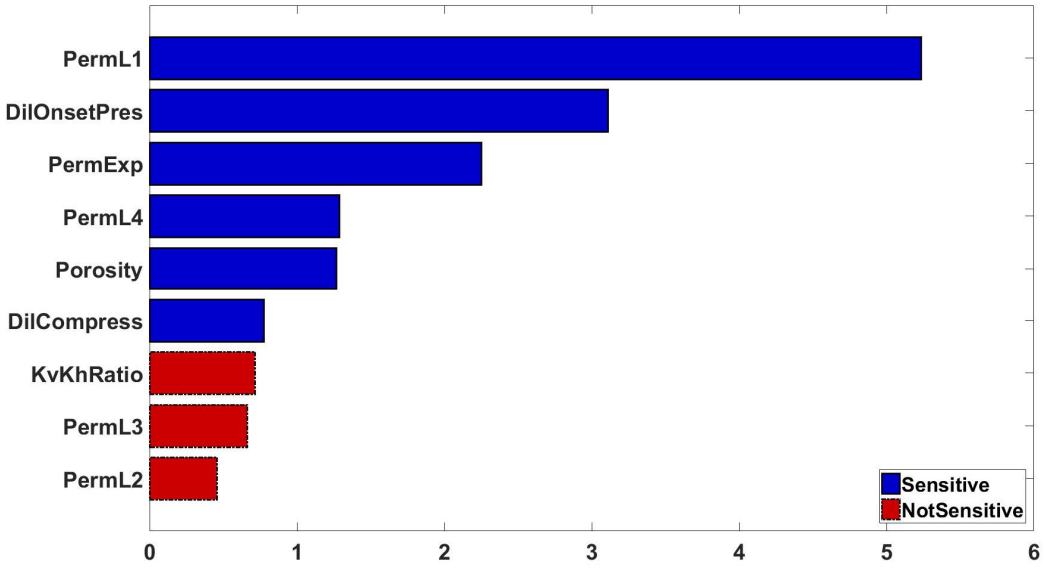


Figure 4.51: Average distance (sensitivity measure) for each parameter of Cumulative Water Production objective function

4.6.4.3 Cumulative Steam Injection objective function

Using the same setup and procedure as the previous two objective functions, the following Figures 4.52 to 4.56 display the model ensemble, clustering results, the standardized CDF distance for each class, and finally the average distance for each parameter.

The CDF plots of each input parameter within each cluster are available for verification in Appendix B.

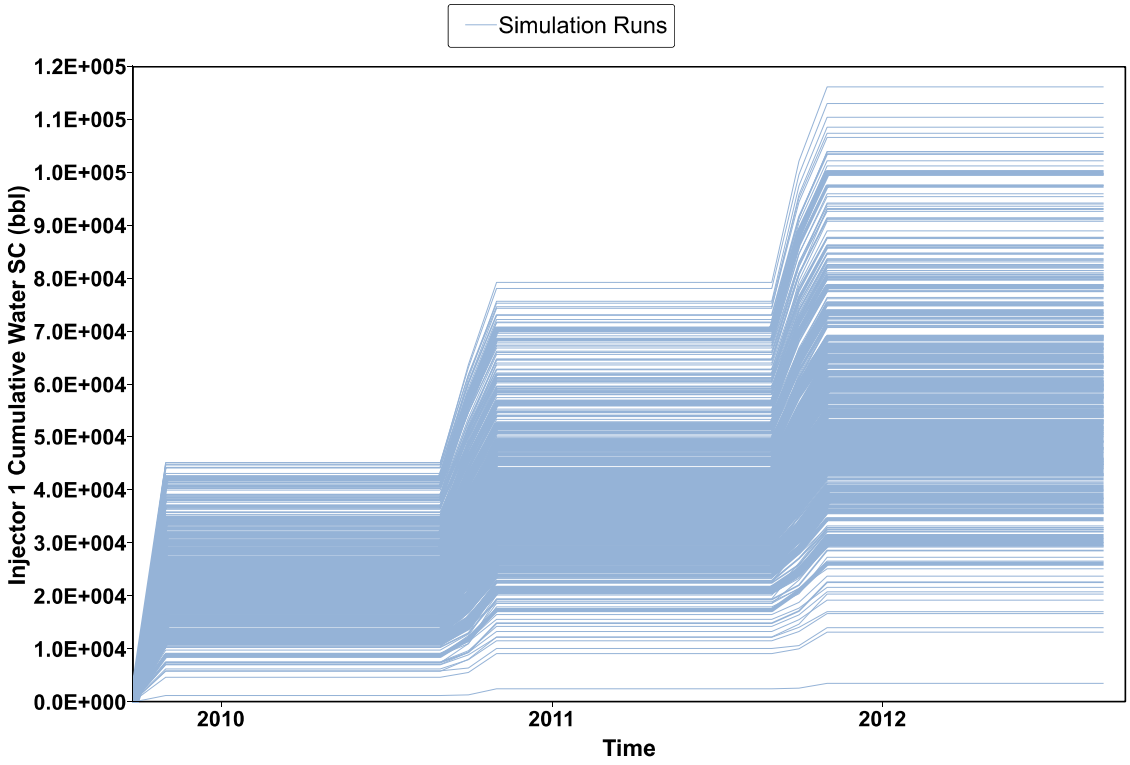


Figure 4.52: Time series model ensemble for the injector Cumulative Steam

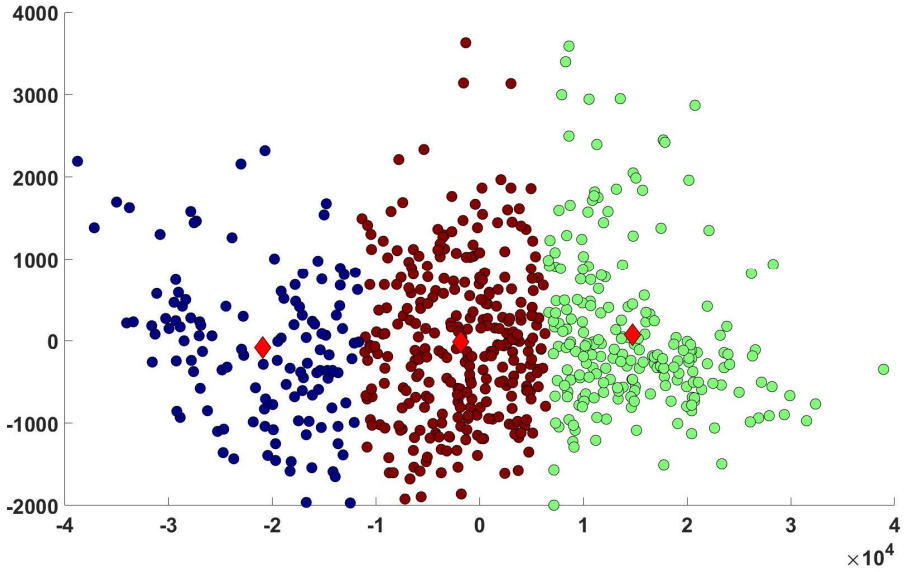


Figure 4.53: 2D clustering result for CSS pilot Cumulative Steam Injection model ensemble

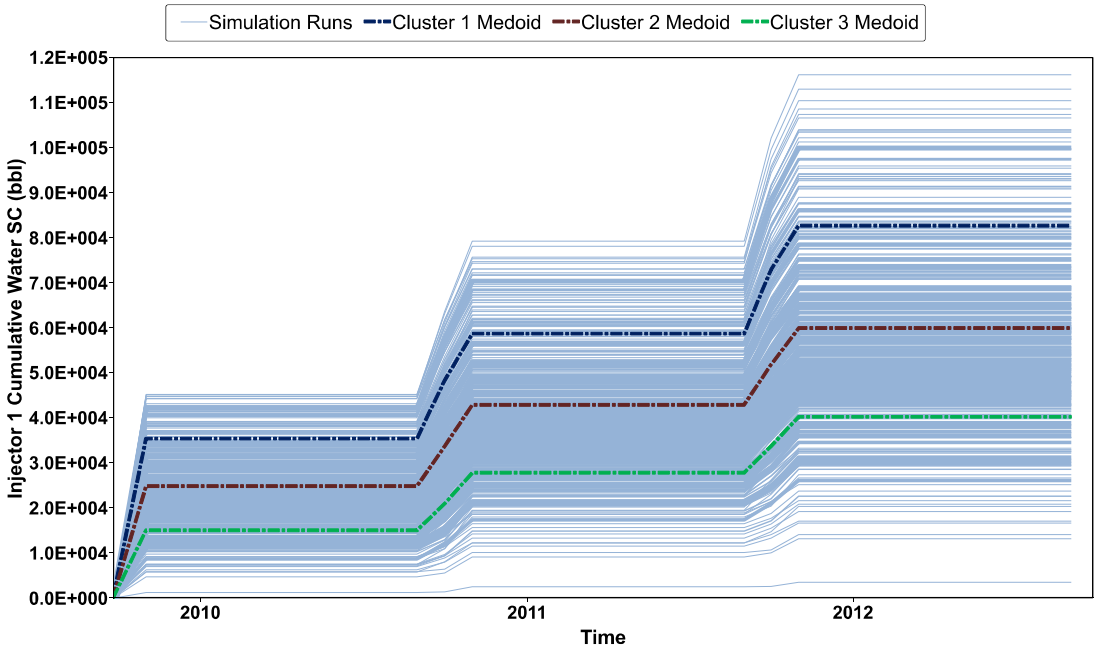


Figure 4.54: 3 cluster medoids for CSS pilot Cumulative Steam Injection model ensemble

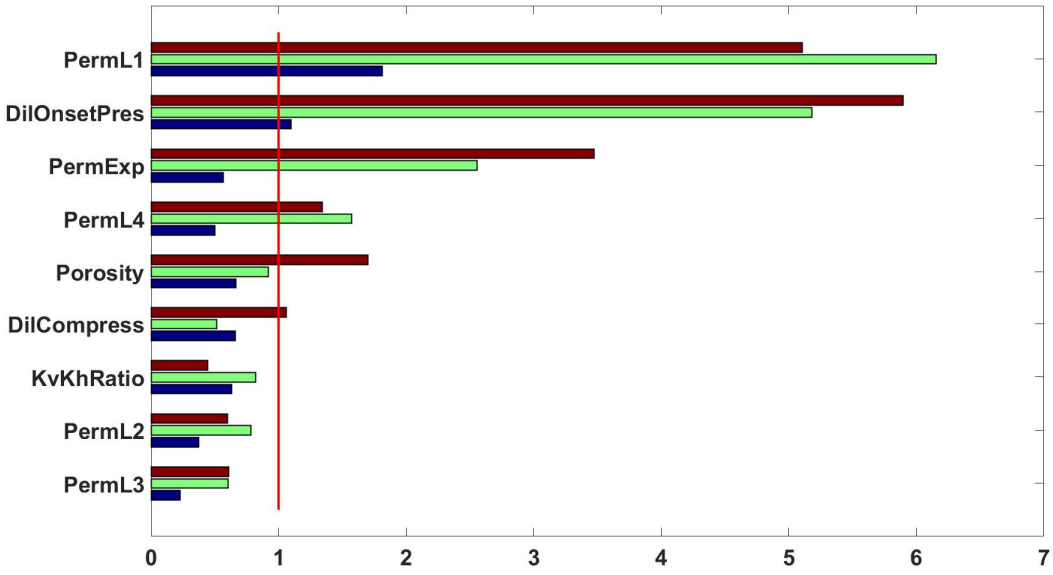


Figure 4.55: Standardized CDF distances for each cluster for Cumulative Steam Injection objective function

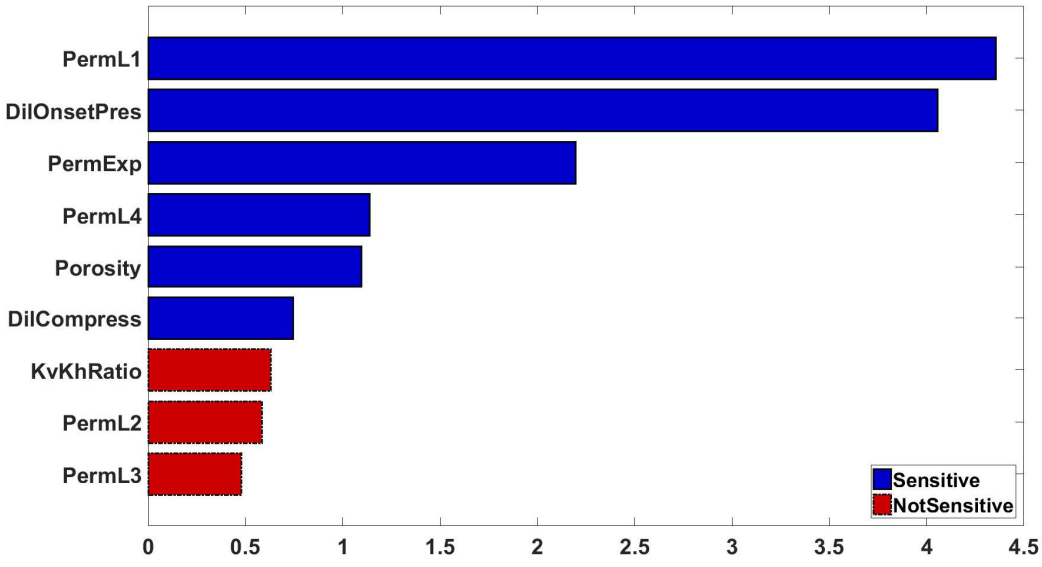


Figure 4.56: Average distance (sensitivity measure) for each parameter of Cumulative Steam Injection objective function

The final result identifies permeability in layer 1, dilation onset pressure, and the permeability exponent as the most significant parameters for cumulative steam injected objective function.

With the results available for all three objective functions, we can see that the distance based sensitivity analysis method applied to this case study was successful in ranking the input parameters in terms of their importance with great agreement with the much more computationally expensive variance based Sobol method. Besides that, since this method is based on clustering the model ensemble it could also be used in cases where multiple realizations of the same model is incorporated in the study to include the geological uncertainty. In the next section, we will examine such case and study the impact of geological uncertainty using the distance based sensitivity analysis method.

4.6.5 Brugge field

In reservoir engineering, several realizations of petrophysical and/or lithological models of the subsurface reservoir are generated using a geostatistical algorithm. The transfer function in this application is a numerical flow simulator, and the realizations are submitted to flow simulation in order to assess the uncertainty in reservoir flow performance. We use the multidimensional scaling and clustering method described in this chapter to select a representative subset of these realizations and perform only a small number of flow simulations whose response has the same characteristics as the entire set of realizations.

The probability distribution of the flow response of interest, in this case the oil recovery factor, is then determined.

The Brugge field is a synthetic reservoir generated by TNO for Scientific Research groups to study history matching and optimization methods for a water flooding project (Peters et al., 2009). It is a large scale benchmark case. Figure 4.57 shows the top structure of this field which consists of an E-W elongated half-dome with a large boundary fault at its northern edge and one internal fault with transmissibility of one.

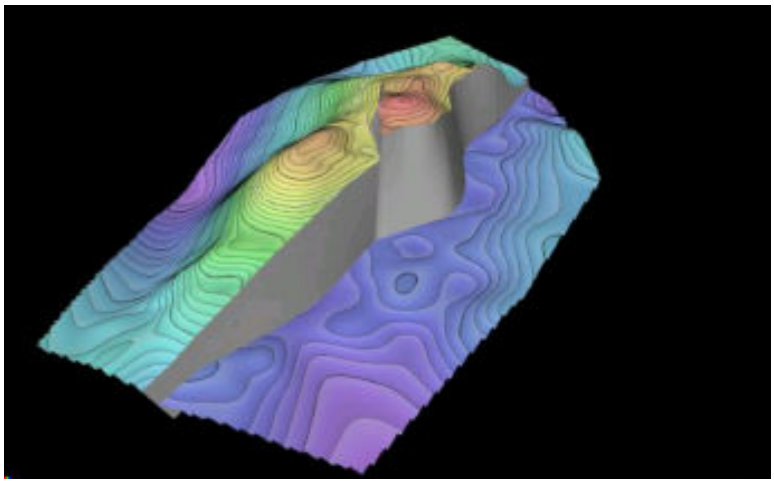


Figure 4.57: 3D view of the top structure map and faults of the Brugge field

The Brugge field consists of four formations from top to bottom, namely Schelde, Maas, Waal and Schie. There is no continuous shale barrier within each formation. Table 4.2 shows the stratigraphy of the Brugge field with main characteristics. The formations vary from a highly permeable layer to a less permeable layer. The underlying and edge aquifers are inactive.

Table 4.2: Stratigraphy sequence of Brugge field with main characteristics

Formations	Avg Thickness, m	Avg Porosity, %	Avg Horizontal Permeability, mD	Avg N/G, %	Depositional Environment	Remarks
Schelde	10	20.7	1105	60	Fluvial	Discrete sand bodies in shale
Maas	20	19	90	88	Lower Shoreface	Contain loggers: carbonate concretions
Waal	26	24.1	814	97	Upper Shoreface	-
Schie	5	19.4	36	77	Sandy shelf	Irregular carbonate patches

For this model, first, facies modeling is created by a sequential indicator simulation method based on facies logs for each formation. Next, porosity is modeled by the sequential Gaussian simulation method following facies types and porosity logs, and then permeability is generated by a cloud transform method collocated with the calculated porosity as shown in Figure 4.58.

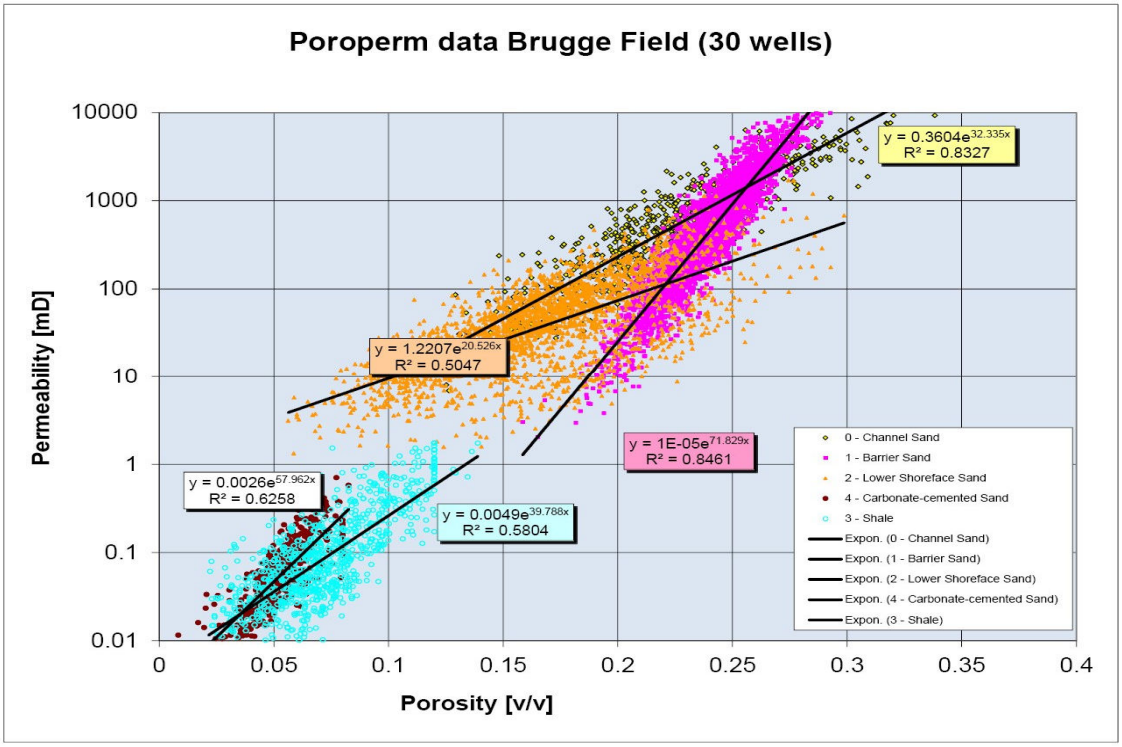


Figure 4.58: Correlation between porosity and permeability based on well log data

Thus the properties of the simulation model satisfy geological trends. Finally, net to gross and water saturation are also generated by the sequential Gaussian simulation method for the whole reservoir.

In the history matching process (Figure 4.59), uncertainty is added to the variograms parameters and an optimizer invokes the geomodeling software with updated variogram parameters to generate multiple realizations (new reservoir properties). The optimizer software then matches the model history by minimizing specified target differences with respect to the field data (Nguyen et al., 2015).

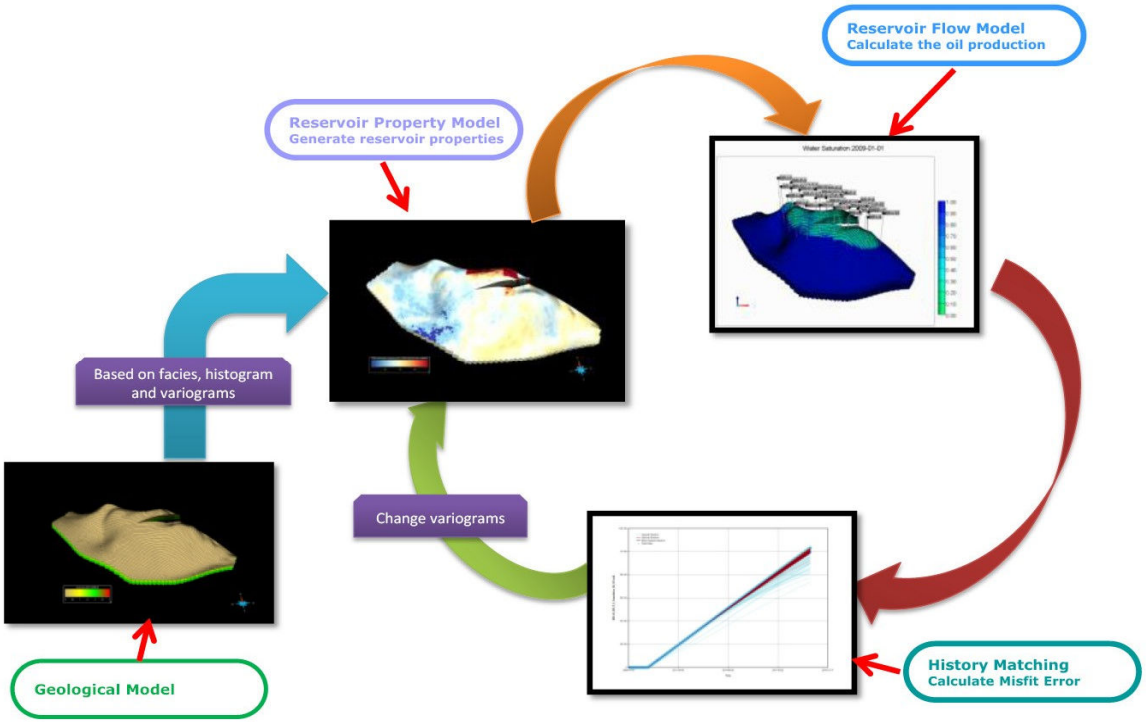


Figure 4.59: The history matching workflow (Nguyen et al., 2015)

After obtaining a good matched model, 100 best-matched models which have a history matching error less than 7%, are selected. Their cumulative oil production curves are shown in Figure 4.60. Then six representative models out of these 100 models with probability of 0, 20, 40, 60, 80, and 100% are chosen to optimize the production rate and the injection rate for the next twenty years with the objective being maximizing an average net present value of these six models.

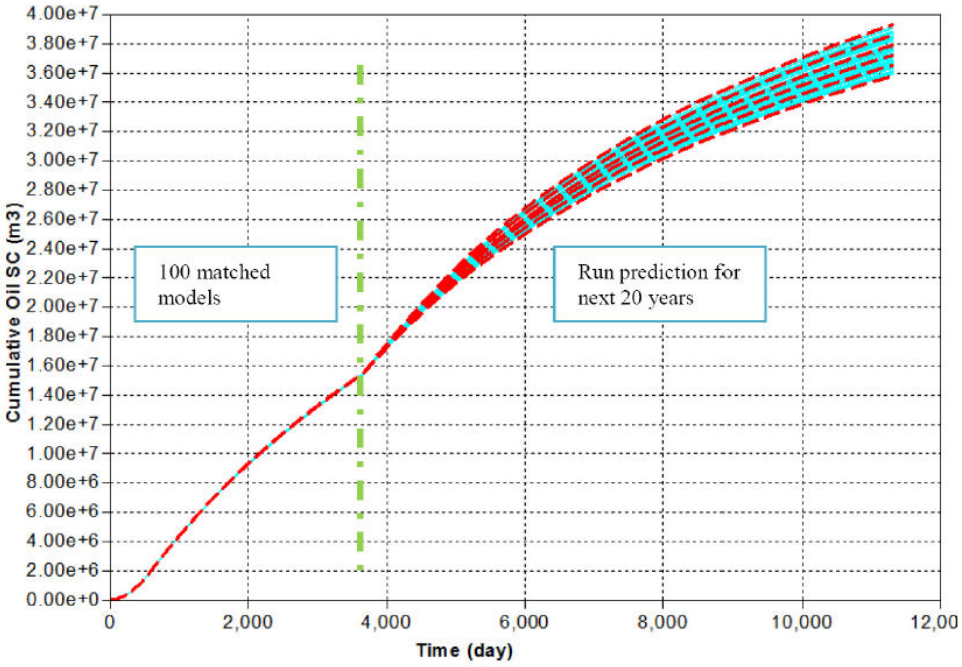


Figure 4.60: Cumulative Oil Production curves of 100 models with history matching error less than 7%, and the 6 representative models shown in red

We will incorporate these 6 representative model to represent the reservoir's geological uncertainty in the next part of our study. We define the input parameters of model in Table 4.3.

Table 4.3: Brugge field model input parameters and values

Parameters	Unit	Uncertainty Range
DWOC	m	1670 - 1690
no	-	1 – 1.5
nw	-	4 – 4.5
krwiro	-	0.6 – 1
krocw	-	0.45 – 0.6
swcon	-	0.25 – 0.4
swcrit	-	swcon
Pb	kPa	800 – 3000
Ratio	-	0.1 – 0.3
sorw	-	0 – 0.2
Min Perm	mD	2500 - 3500
Min_Por	-	800 – 1400
Min_facies	-	1000 - 1200

Now, in addition to these continuous parameters, the reservoir geological property distribution (Distribution) is added as an additional parameter. Six distributions obtained from the previous realization selection step are specified (Figure 4.60), each representing a different geological interpretation. The probabilities of these realizations are equal. therefore, runs have different geostatistical realizations assigned. Distances between the oil recovery factor for the entire field over the simulation period of the models are calculated

from a total of 618 models. The clustering algorithm thus employs hundreds of response values to distinguish the models and group them into 5 classes.

4.6.5.1 Clustering

The runs for the entire field recovery factor from which the distances are calculated are plotted in Figure 4.61.

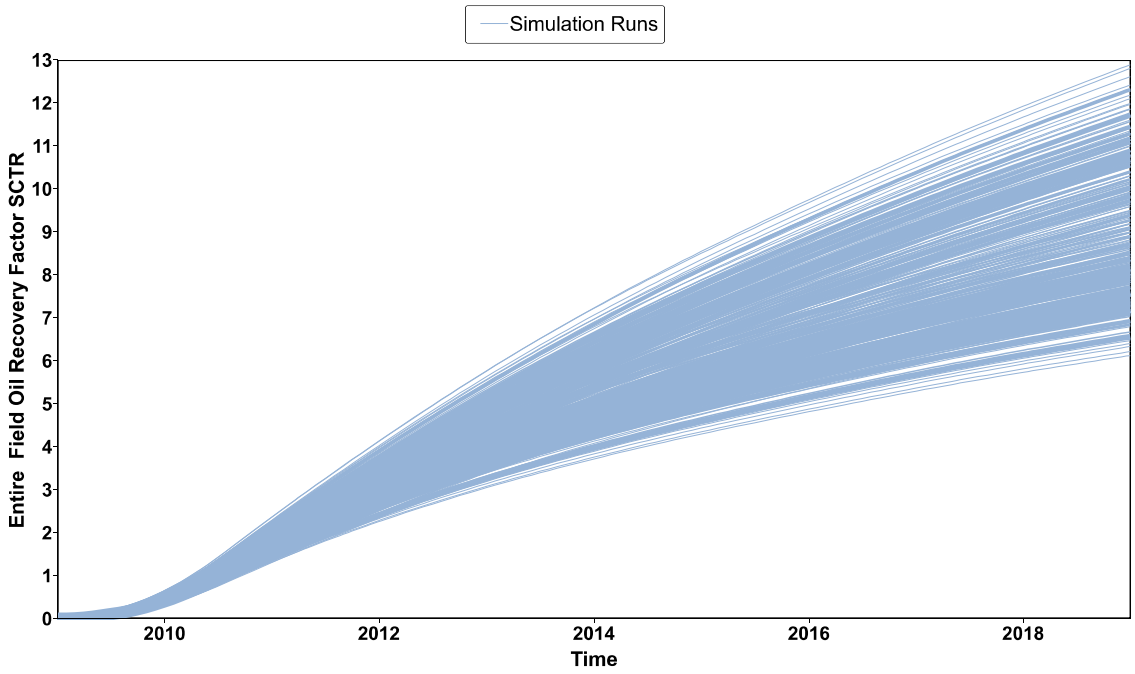


Figure 4.61: Time series model ensemble for the entire field Oil Recovery Factor

We define the number of classifications to 5 to represent P10, P25, P50, P75 and P90 of the runs though the clustering medoids. The MDS space for 5 clusters are visualized below in Figure 4.62 and the corresponding medoids are highlighted in the model ensemble in Figure 4.63.

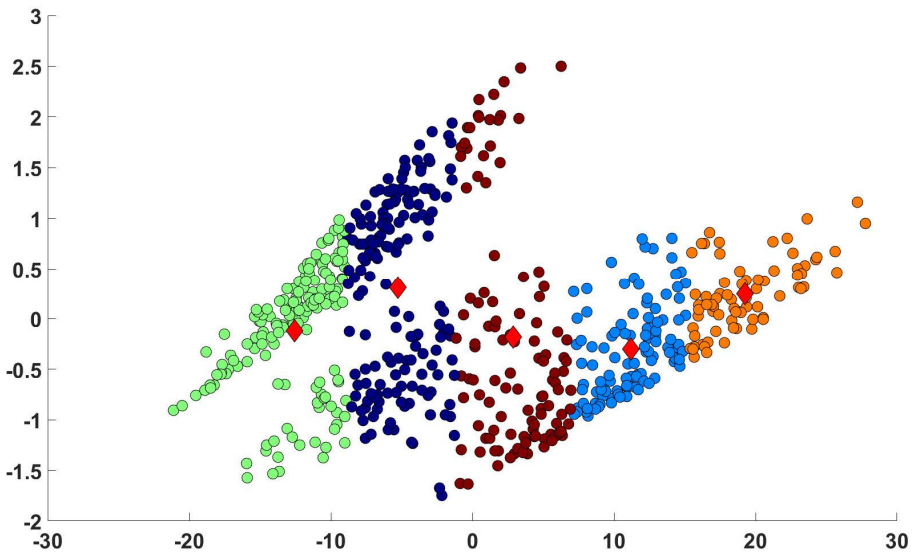


Figure 4.62: 2D clustering result for CSS pilot Field Oil Recovery Factor model ensemble

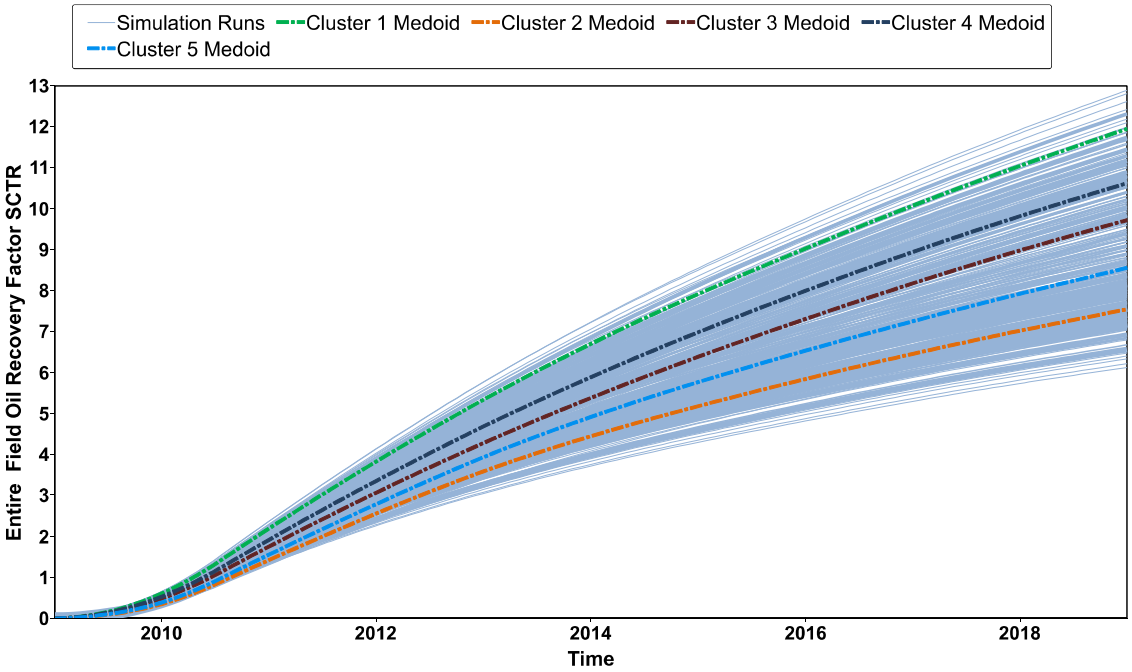


Figure 4.63: 5 cluster medoids for Oil Recovery Factor model ensemble

4.6.5.2 Parameter Distributions

Figures 4.64 to 4.76 show the CDFs for the input parameters with the prior CDF in black, and the class-conditional CDFs for each cluster (blue, green, red, orange, and light blue).

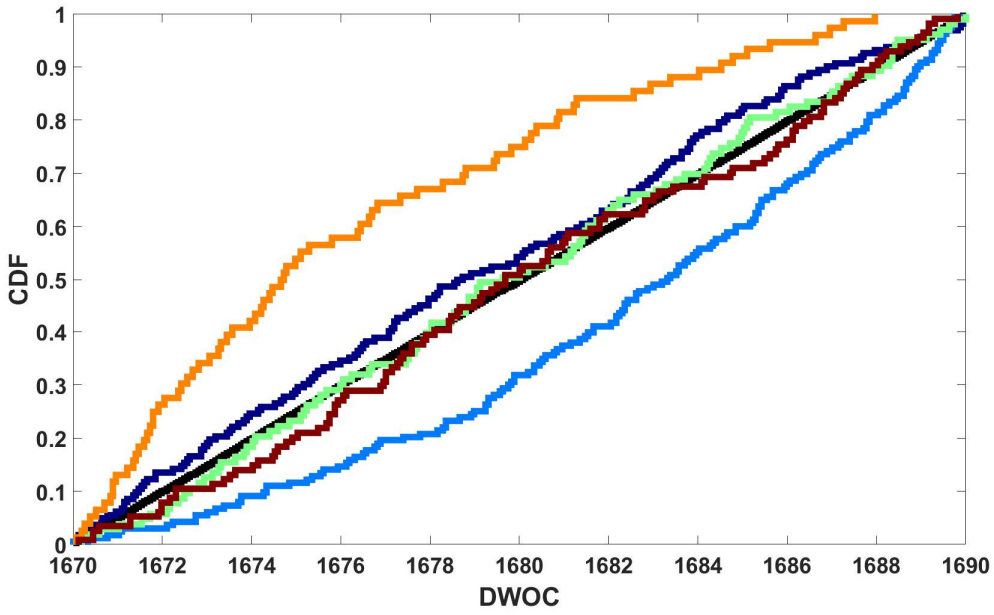


Figure 4.64: Empirical CDF curves for parameter 'DWOC'

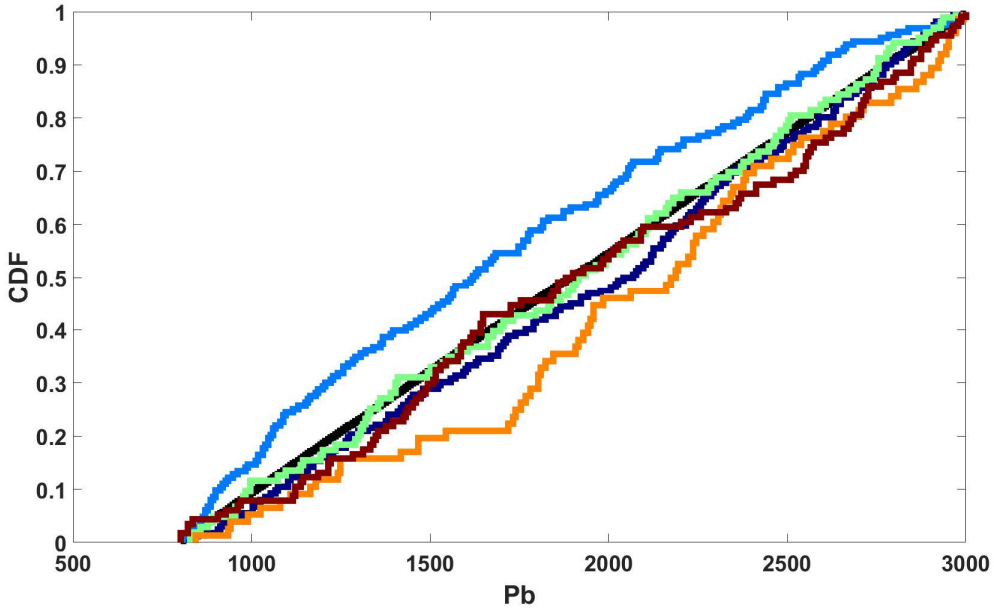


Figure 4.65: Empirical CDF curves for parameter 'Pb'

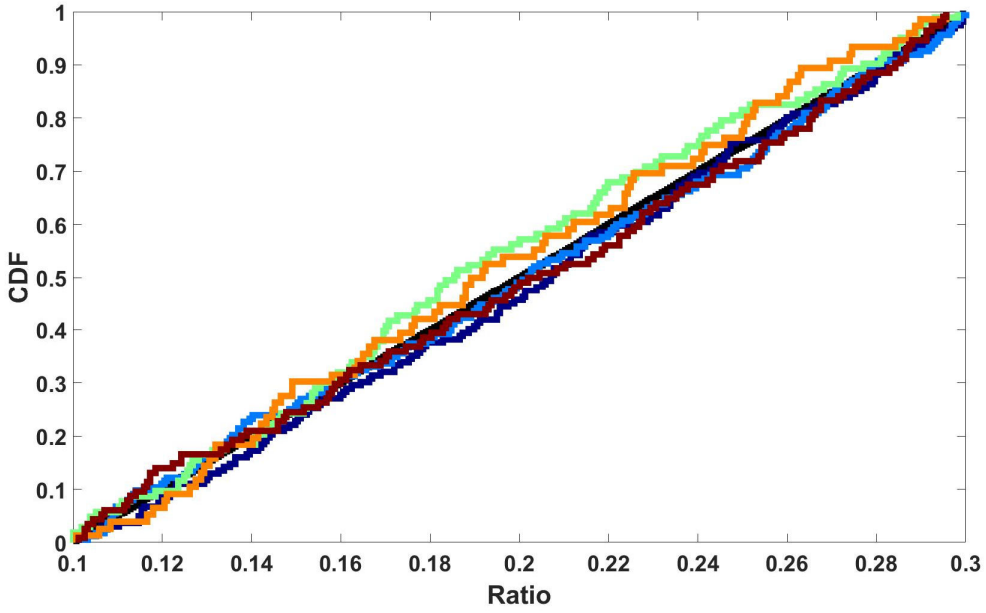


Figure 4.66: Empirical CDF curves for parameter 'Ratio'

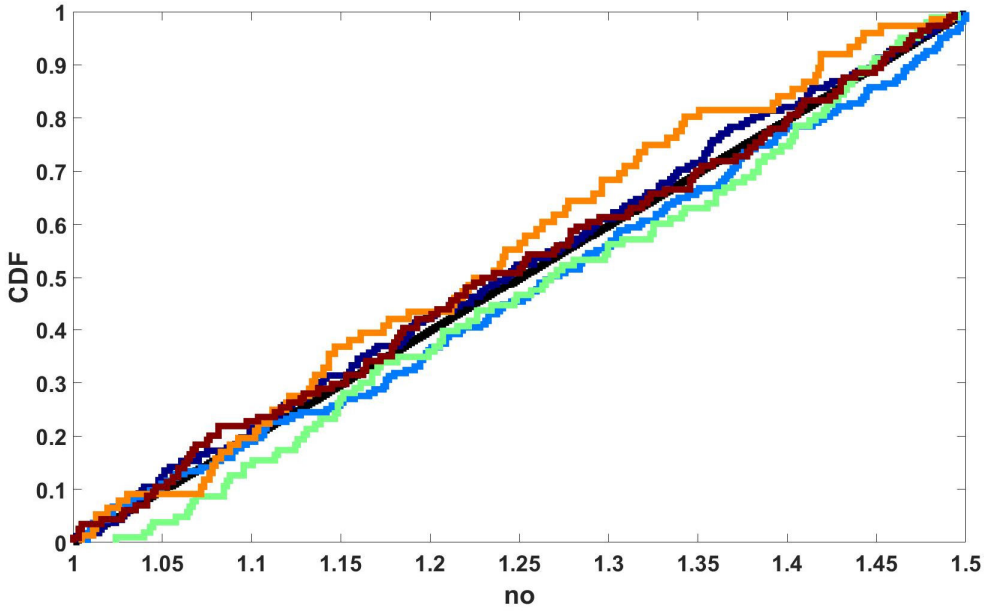


Figure 4.67: Empirical CDF curves for parameter 'no'

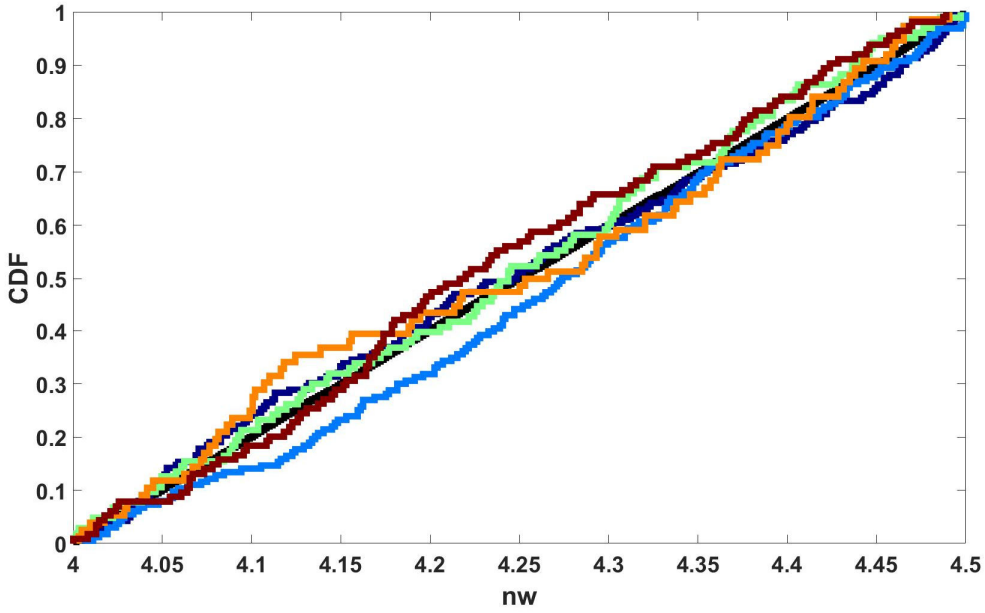


Figure 4.68: Empirical CDF curves for parameter 'nw'

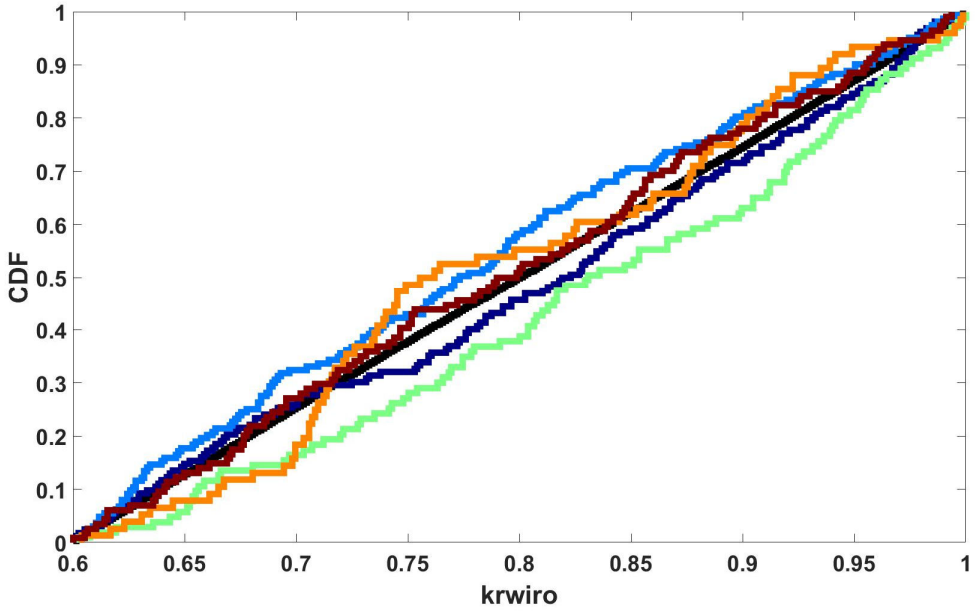


Figure 4.69: Empirical CDF curves for parameter 'krwiro'

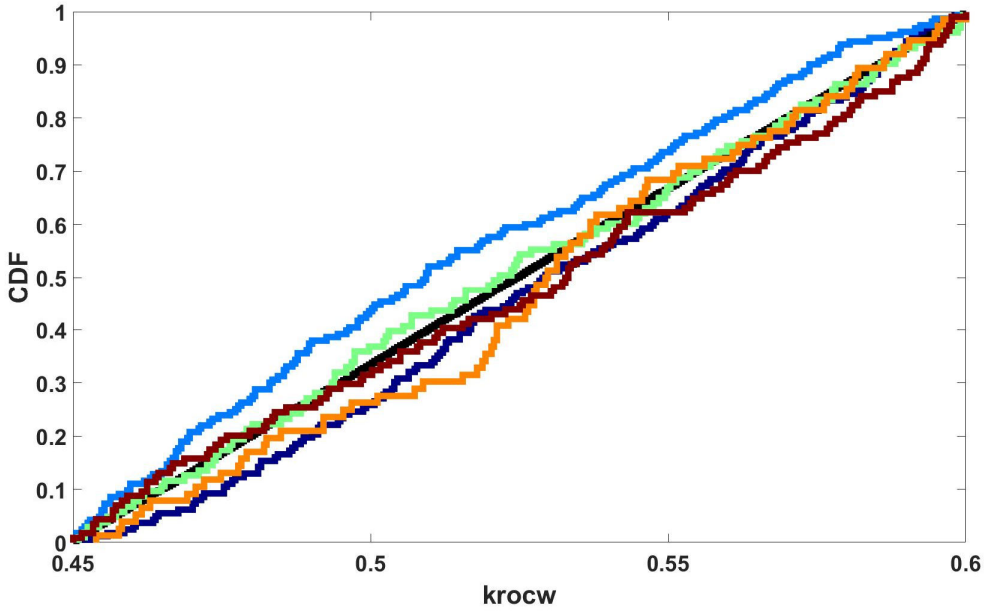


Figure 4.70: Empirical CDF curves for parameter 'krocw'

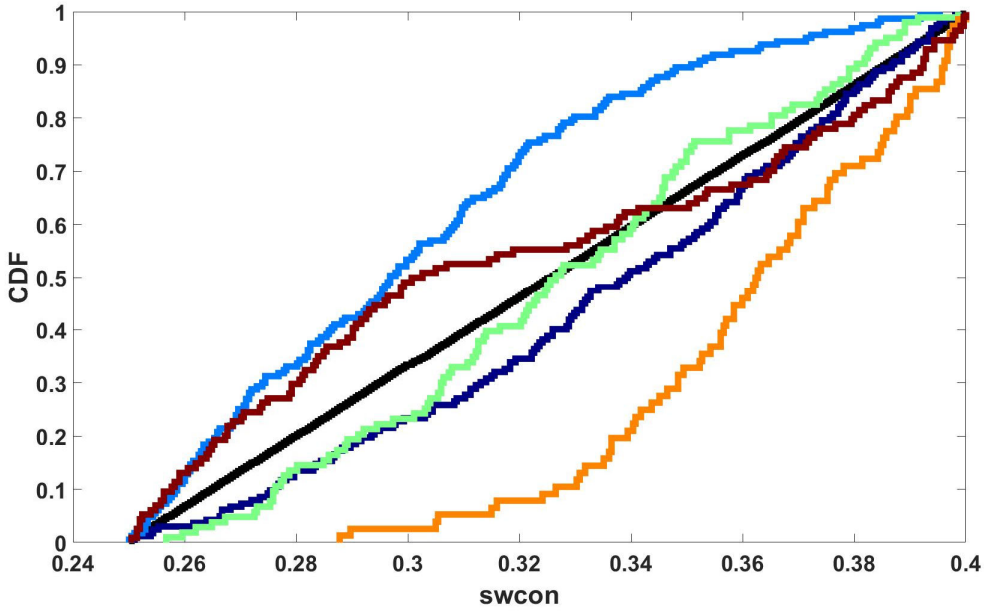


Figure 4.71: Empirical CDF curves for parameter 'swcon'

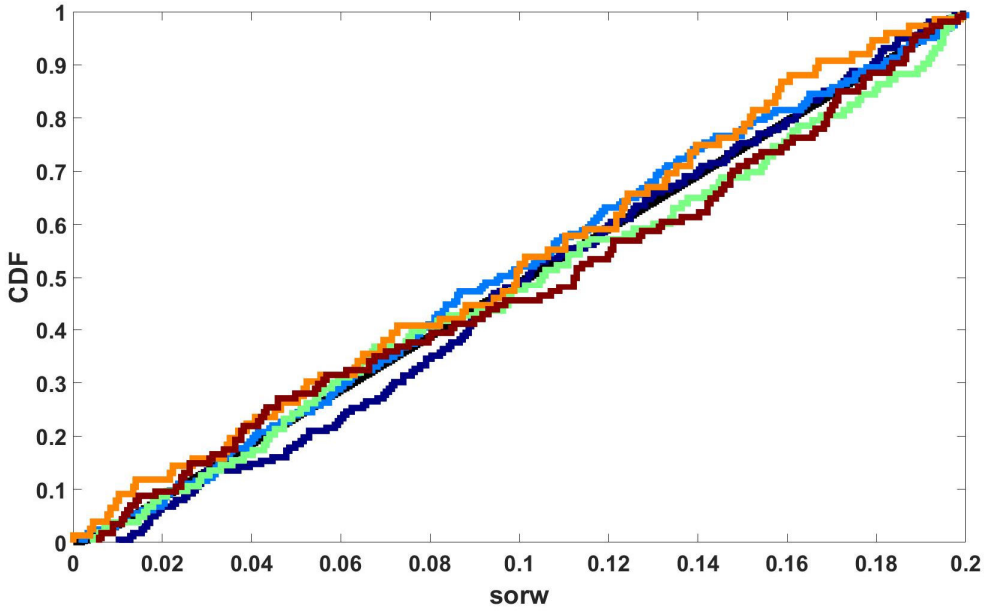


Figure 4.72: Empirical CDF curves for parameter 'sorw'

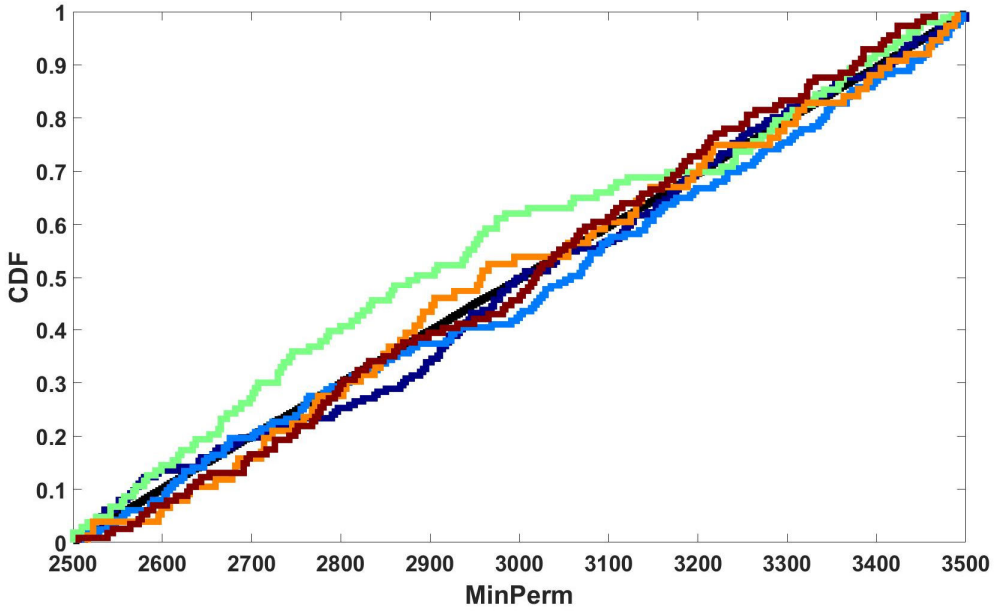


Figure 4.73: Empirical CDF curves for parameter 'MinPerm'

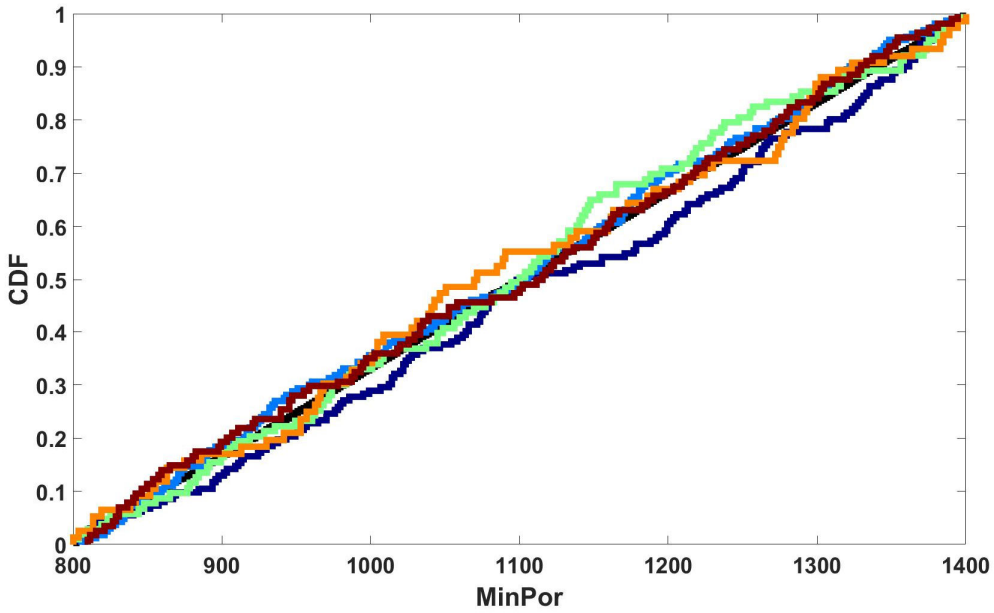


Figure 4.74: Empirical CDF curves for parameter 'MinPor'

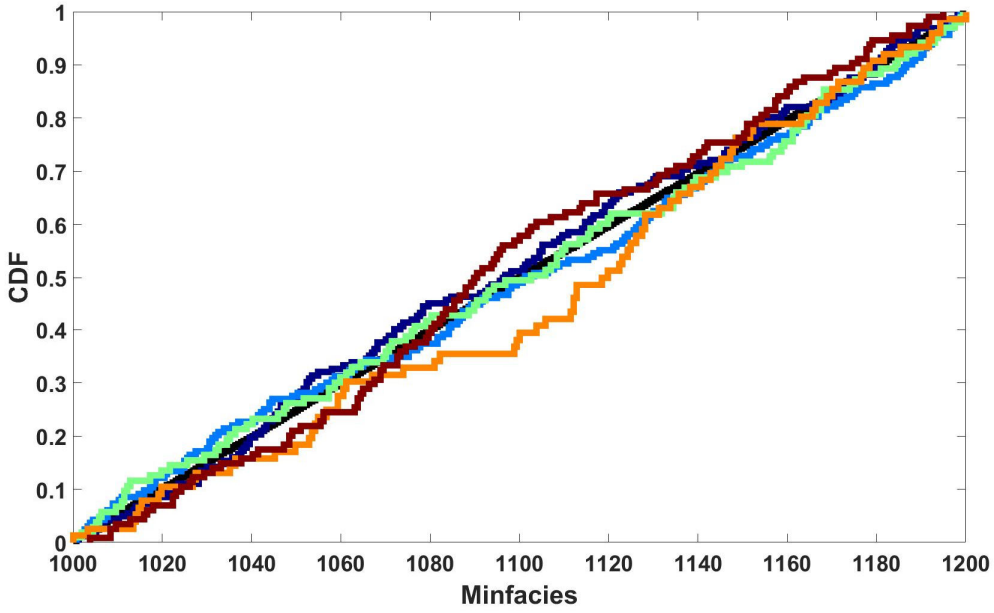


Figure 4.75: Empirical CDF curves for parameter 'MinFacies'

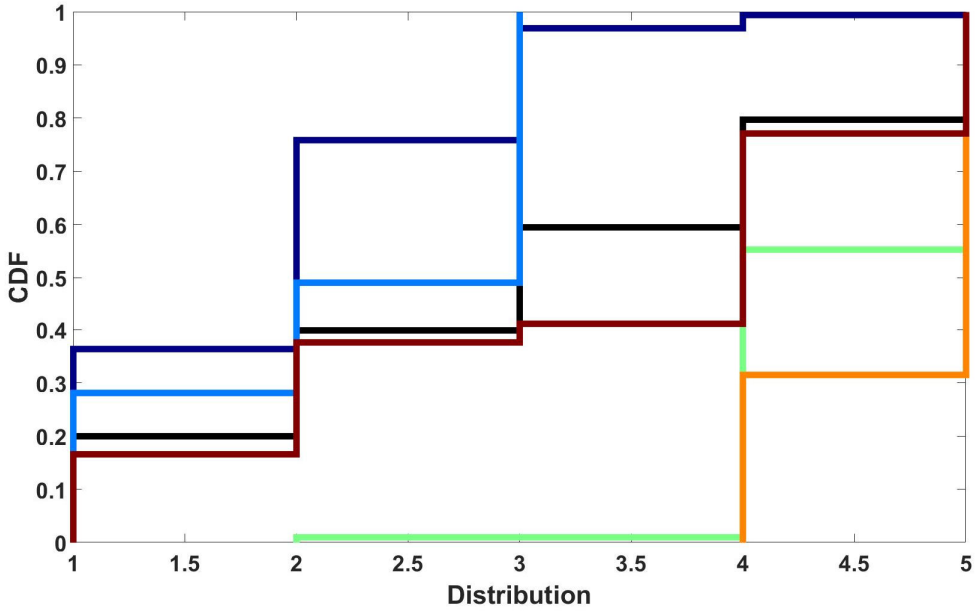


Figure 4.76: Empirical CDF curves for parameter 'Distribution'

4.6.5.3 Main effect sensitivities

The standardized CDF distance for each class (Figure 4.77), and average distance over all classes (Figure 4.78) show the dominat parameters on Oil Recovery Factor for the Brugge field.

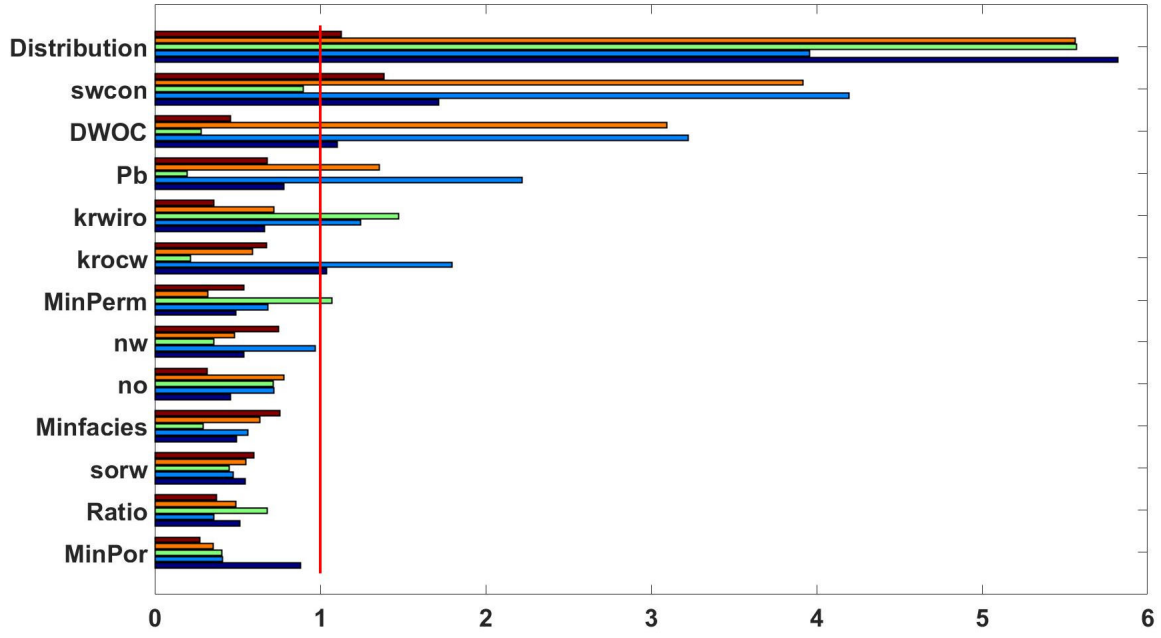


Figure 4.77: Standardized CDF distances for each cluster for Oil Recovery Factor objective function

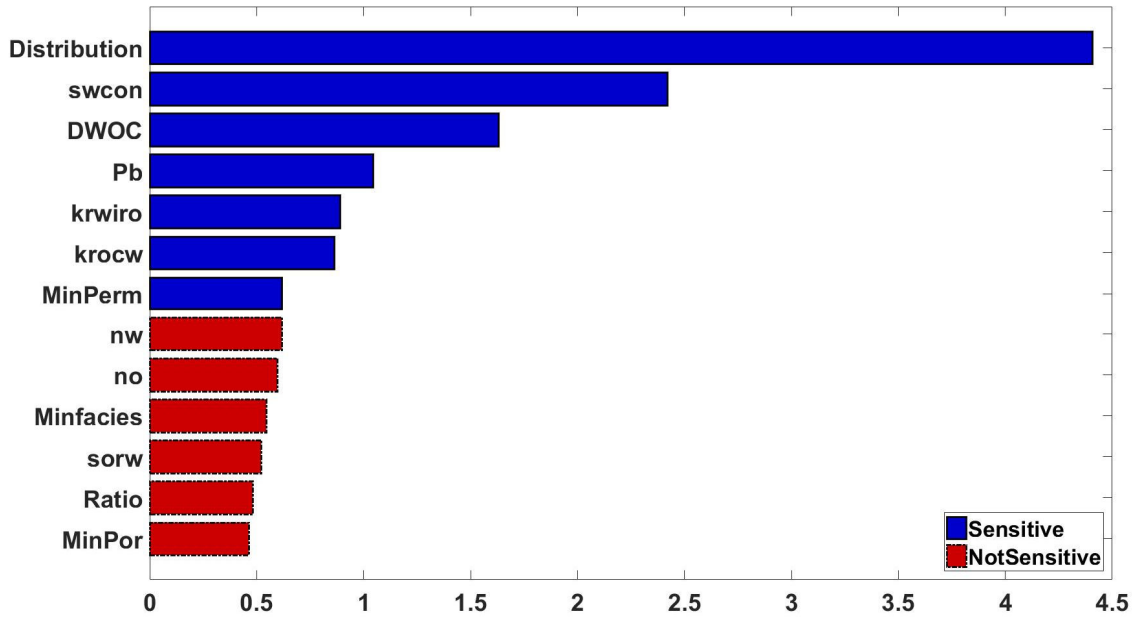


Figure 4.78: Average distance (sensitivity measure) for each parameter of Brugge field

The ranking of influential parameters demonsterates the contribution of reservoir property distrubution that we incorporated in the model through the geological realizations, and proves the importance of including the geological uncertainty in the workflow.

4.7 Conclusions and Summary

In this chapter, a distance-based sensitivity analysis approach has been presented which could be applicable to models with stochastic responses and diverse types of input parameters (continuous, discrete, scenario based, etc.). There are several advantages to this approach. First, the experiment design procedure can account for any type of parameter

with different CDFs, discrete, continuous, or even categorical or scenario based. Second, the classification is not limited to a single response, but can be an amalgam of several responses, based upon the requirements of the simulation engineer. Third, the responses can be stochastic in nature, without any modification of the SA approach, and can thus deal with spatial uncertainty in Earth models. Finally, since the model responses are used only for classification, proxy models (as introduced in the previous chapter) can also be employed in this approach. In other words, what is important is that the responses correctly classify the models—the absolute accuracy of the proxy response itself is less consequential. This is a significant advantage for computationally intensive models when a fast proxy model could be constructed. Overall, the CDF distance allows quantifying parameter influence conveniently and is a flexible measure of sensitivity that works for all distributions which is robust for small sample sizes, hence providing a computational advantage even in cases of using CPU exhausting simulations. In addition, standardization of the CDF distance allows for a simple interpretation of sensitivity. Through the case studies provided in this chapter, comparison of the distance based sensitivity analysis with the other SA approaches was shown and demonstrated good agreement of the results for all the analyzed cases.

Chapter Five: Conclusions and Recommendations

5.1 Summary and Conclusions

For many reservoir engineering scenarios, current methodologies do not adequately quantify and manage the parameter uncertainty as it arises during the design process, which can lead to unacceptable risks, wrong conclusions, increase in cost, and field development schedule overruns. Therefore, the main objective of this thesis was to provide innovative workflows for petroleum reservoir engineers during sensitivity analysis studies. In this work, we presented and developed several methods for parameter uncertainty quantification and sensitivity analysis, which can be used to better understand and mitigate the effects of uncertainty in the reservoir modeling workflow.

In chapter 2, we discussed situations where a large number of parameters must be used to describe reservoir dynamics in all possible operating conditions. Hence, the Morris screening approach has been applied to help with categorizing the parameters in groups of relative importance with a low computational cost. An iterative procedure based on creating trajectories was introduced and executed in order to calculate the partial derivatives of the objective functions and find the main effect contribution of the parameters to the output uncertainty as well as indication of interactions.

Morris method can be used as a qualitative approach to get overall sensitivity estimations with a limited number of simulations. It is a reliable screening step to simplify a model, as

it can identify the input parameters with relative low influence which could be fixed to a nominal value.

Regarding parameter significance ranking, Morris approach reduces the risk to underestimate and fix non-negligible factors, and compared to the local sensitivity analysis, we can conclude that it overcomes the inherent drawbacks and challenges of local SA including being dependent on a given point in the input uncertainty space, and assumptions of linearity and normality. Also, the Morris approach provides a good approximation of global sensitivity measure, where the computational cost of running more sophisticated sensitivity analysis methods could be excessively high.

In chapter 3, the variance based Sobol method of Global sensitivity analysis was introduced. By decomposing the variance of the output of the model into fractions and attributing the fractions to inputs, Sobol method provides sensitivity measure quantitatively. This includes the main and interaction effects for the parameters.

It is a very good approach for cases of high complexity where Sobol method provides accurate relative contributions of parameters. However, we observed that Sobol method is not computationally very efficient for complex models with many inputs. If sensitivity indices for other effects such as two-way and three-way interactions between inputs are to be estimated, the computational cost will increase substantially as well. Due to this drawback, the application of surrogate modelling was incorporated in this research to create an approximation of the response so that the computation could be done on the surrogate

model. Through this approach, the surrogate model is built and its accuracy of prediction is verified first. If the accuracy was deemed reasonable, then the Sobol analysis would be followed. Overall, the Sobol GSA results supplemented with reservoir engineering knowledge provide valuable insights into the engineering problems at hand, thereby helping engineers in doing their modeling work with much less uncertainty, fine-tuning parameters, and checking the robustness of reservoir model, among others.

In chapter 4, we presented a classification-based method for sensitivity analysis. The method relied on a correlation between a chosen distance measure and the response variables of interest. Using a chosen model-specific distance is an important additional tool which makes the task of response uncertainty quantification for reservoirs more effective. Given the distance measure, we employed kernel k-means clustering to select sub-categories of models which include the P10, P50, P90, etc. of the entire set of realizations depending on the choice of the user. With these clusters available, the CDF distance allows quantifying parameter influence. This is a flexible measure of sensitivity that is robust and works for all distributions. In addition, interpretation of sensitivity results is simple through standardization of the CDF distance. This approach could also be applied to models with stochastic responses and different types of input parameters including continuous and discrete, categorical, scenario based, etc. This includes parameters that cannot be ranked or given a value. It could also be used for an amalgamation of several responses, for example the reservoir's history can be considered a special model added as a response. For

this approach, although there are automatic clustering algorithms that can select the optimal number of classes, ultimately the user should decide on the optimal number with which to perform the analysis by incorporating the engineering knowledge about the model. This degree of flexibility in interpretation is an advantage of this SA methodology. Studying the joint variation of parameters within a cluster as well as the joint variation of parameters between clusters also provide an insight into which combination of parameters is important to the reservoir model response under study.

Whether using screening methods such as Morris, variance based Sobol, or the distance based SA, sensitivity analysis is necessarily subjective. The methods presented in this work could all be either used separately to carry out sensitivity studies with reliable results, or complimentary to each other as the complexity of the modeling increases and in cases where more\different requirements are needed.

In this work, many comparisons of the methods with each other and against known test functions were conducted and have shown good agreement of the results for the case studies analyzed. Although we should not expect that all SA methods provide identical results in general, different methods introduced in this work and their applications demonstrated the usability of each approach under many different scenarios.

A robust and reliable software infrastructure is also the backbone of this study, as the number of runs, parameters, and data mining needs put significant usability requirements

when executing the workflows. Although this thesis did not describe any of the software architecture and design requirements associated with the implementation of the work, there is an important software element involved that should not be dismissed.

5.2 Major Contributions

The major contributions of this study are briefly summarized as follows:

- Application of an efficient sensitivity screening method, Morris Elementary Effects, as a qualitative SA technique to tackle models with large number of uncertain parameters with a relative low computational cost as the main approach to first analyzing the inputs.
- Extending the sampling algorithm of Morris method for better exploration of input space and increasing the accuracy of the result.
- Application of the robust variance based Sobol global sensitivity analysis method for reservoir models of high complexity, non-linearity and non-monotonicity and providing exact quantitative measure for the effect of parameters and interactions involved in the model.
- Implementing the LHD sampling in comparison with the original approach of using the random Monte Carlo sampling design due to its drawback in providing not enough information regarding the input space.

- Development of a very efficient Gray code implementation of low-discrepancy sampling sequence for when much higher uniformity is required in the sampling space.
- Investigating the effect of the orthogonality of the experiment deign for sensitivity analysis results and incorporating the low-discrepancy design over LHD or random Monte Carlo sampling for cases when high orthogonality is a requirement for accuracy and reliability of the SA approach.
- Integrating a surrogate modeling based approach to successfully handle the high computational cost of sensitivity analysis methods through application of Single-Layer RBF, and Multi-Layer Neural Network proxy models, with providing necessary verification of the proxy approximation.
- Incorporation of a distance based classification of models using full-physics finite-difference simulation and providing desired number of response classes.
- Application of classification based sensitivity analysis method to conduct SA studies with any type of parameter and conveniently including geological uncertainty in the SA workflow.

5.3 Recommendations

The Morris method screening method can be extended so that it works with groups of parameters. Due to the limited applicability of parameter grouping in cases of large number

of uncertain parameters in the reservoir, this was not included in this study. However, given a realistic case, Morris method could be extended to include parameter grouping.

The methods in this study could be used in some situations where the simulation is not a deterministic but a stochastic. This means that two model calls with the same set of input variables leads to different output values. A few studies have been done to propose solutions for dealing with Sobol indexes in for these scenarios, but more work is recommended specially in the context of stochastic geological modeling.

Using the distance based approach, more relevant case studies could be investigative for addressing the spatial uncertainty in reservoir modeling workflow. The recommendation could be extended for other case studies involving various types of uncertainty which commonly arise in real field applications in the geosciences.

References

Archer G., Saltelli A., and Sobol I.M., 1997. Sensitivity measures, ANOVA like techniques and the use of bootstrap. J Stat Comput. 58(1): 99–120.

Arpat, G.B., 2005. Sequential simulation with patterns. Stanford University.

Bastidas, L.A., Gupta, H.V., Sorooshian, S., Shuttleworth, W.J., Zhang, Z.L., 1999. Sensitivity analysis of a land surface scheme using multicriteria methods. J. Geophys. Res. 104 (D16), 19481–19490. 383–397.

Batycky, R.P., Blunt, M.J., Thiele, M.R., 1997. A 3D field-scale streamline-based reservoir simulator. SPE Reserv. Eng. 12 (4), 246–254.

Beven, K.J., Binley, A., 1992. The future of distributed models: model calibration and uncertainty prediction. Hydrol. Process. 6 (3), 279–298.

Bishop, C.M., 2009. Pattern Recognition and Machine Learning, Springer Science + Business Media, LLC, NY.

Breiman, L., Friedman, J., Olshen, R., Stone, C., 1984. Classification and Regression Trees. Wadsworth International Group, Pacific Grove, California.

Broomhead, D. and Lowe D., 1988. Radial Basis Functions, Multi-Variable Functional Interpolation and Adaptive Networks: Technical Report, DTIC Document.

Caers JK, 2005. Petroleum geostatistics. Society of Petroleum Engineers.

Caers, J., 2008. Distance-based stochastic modeling: theory and applications. In 21st SCRF Annual Meeting Report.

Caers J., 2011. Modeling Uncertainty in the Earth Sciences. John Wiley and Sons, Chicheseter, UK.

Campolongo F., Cariboni J., Saltelli A., 2007, An effective screening design for sensitivity analysis of large models. Environmental Modelling and Software 22:1509–1518

Campolongo, F. and Braddock, R., 1999. The use of graph theory in the sensitivity analysis of the model output: a second order screening method. Reliability Engineering & System Safety, 64(1), pp.1-12.

Campolongo F., Cariboni J., Saltelli A., 2003. Sensitivity analysis: the Morris method versus the variance based measures.

Cannavó, F., 2012. Sensitivity analysis for volcanic source modeling quality assessment and model selection. Comput. Geosci. 44, 52–59.

Ciriello, V., Di Federico, V., Riva, M., Cadini, F., De Sanctis, J., Zio, E., Guadagnini, A., 2013a. Polynomial chaos expansion for global sensitivity analysis applied to a model of radionuclide migration in a randomly heterogeneous aquifer. Stoch. Environ. Res. Risk Assess. 27 (4), 945–954.

Ciriello, V., Guadagnini, A., Di Federico, V., Edery, Y., Berkowitz, B., 2013b. Comparative analysis of formulations for conservative transport in porous media through sensitivity-based parameter calibration. Water Resour. Res. 49 (9), 5206–5220.

Computer Modelling Group Ltd. “CMOST User Guide; Version 2015.10”, Calgary, Alberta, Canada

Corder GW, Foreman DI, 2009. Nonparametric statistics for non-statisticians: a step by step approach. Wiley, New Jersey Cox DR (1984) Interaction. Intern Stat Rev 52(1):1–24

Crosetto, M., Tarantola, S., 2001. Uncertainty and sensitivity analysis: tools for GIS-based model implementation. Int. J. Geogr. Inf. Sci. 15, 415–437.

D.J.Kentwell, L.M.Bloom, G.A.Comber, 1999. Geostatistical Conditional Fractal Simulation with Irregularly Spaced Data, Mathematics and Computers in Simulation, Vol. 48, p. 447 – 456.

Davies DL, Bouldin DW, 1979. A cluster separation measure. IEEE Trans Pattern Anal Mach Intell PAMI- 1(2):224–227

Dejean, J.P., Blanc, G., 1999. Managing uncertainties on production predictions using integrated statistical methods. In: Proceedings of the SPE Reservoir Simulation Symposium. Houston, Texas.

Dubuisson, M.P. and Jain, A.K., 1994. A modified Hausdorff distance for object matching. In Pattern Recognition, Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on (Vol. 1, pp. 566-568). IEEE.

Efron B, Tibshirani R, 1993. An introduction to the bootstrap. Chapman Hall, New York

Epifanio, I., 2014. Mapping the asymmetrical citation relationships between journals by h-plots. J. Assoc. Inf. Sci. Technol. 65 (6), 1293–1298.

Fajraoui, N., Ramasomanana, F., Younes, A., Mara, T.A., Ackerer, P., Guadagnini, A., 2011. Use of global sensitivity analysis and polynomial chaos expansion for interpretation of non-reactive transport experiments in laboratory-scale porous media. Water Resour. Res. 47 (2), 1–14.

Fedutenko, E., Yang, C., Card C., Nghiem, L., 2013, Optimization of SAGD process accounting for geological uncertainties using proxy models. Proceedings of CSPG/CSEG/CWLS GeoConvention, Calgary, AB, Canada.

Fenwick, D., Batycky, R., 2011. Using metric space methods to analyse reservoir uncertainty, In: Proceedings of the Gussow Geoscience Conference. Banff, Alberta.

Fenwick, D., Scheidt, C., Caers, J., 2014. Quantifying asymmetric parameter interactions in sensitivity analysis: application to reservoir modeling. Math. Geosci. 46 (4), 493–511.

Foo, J., Karniadakis, G., 2010. Multi-element probabilistic collocation method in high dimensions. J. Comput. Phys. 229 (5), 1536–1557.

Formaggia, L., Guadagnini, A., Imperiali, I., Lever, V., Porta, G., Riva, M., Scotti, A., Tamellini, L., 2013. Global sensitivity analysis through polynomial chaos expansion of a basin-scale geochemical compaction model. Comput. Geosci. 17 (1), 25–42.

Fuerbringer, J. M., 1996. Sensitivity Analysis for Modellers, Air Infiltration. Review 17(4), 8-10.

Helton JC, Davis FJ, 2003. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. Reliab Eng Syst Safe 81(1):1175–1209

Hill, M.C., Tiedeman, C.R., 2006. Effective Groundwater Model Calibration: With Analysis of Data, Sensitivities, Predictions, and Uncertainty. John Wiley and Sons, Chicheseter, UK.

Homma T., Saltelli A., 1996, Importance measures in global sensitivity analysis of non-linear models. Reliability Engineering and System Safety 52:1–17

Hu, L.Y., 2000. Gradual deformation and iterative calibration of Gaussian-related stochastic models. Math. Geol. 32 (1), 87–108.

Iooss B., Popelin AL., Blatman G., Ciric C., Gamboa F., Lacaze S., Lamboni M., 2012, Some new insights in derivative-based global sensitivity measures. In: Proceedings of the PSAM11 ESREL 2012 Conference, Helsinki, Finland, pp 1094–1104

Iooss, B., Lemaître, P., 2015. A review on global sensitivity analysis methods. In Uncertainty Management in Simulation-Optimization of Complex Systems (pp. 101-122). Springer US.

Ishigami, T., & Homma, T., 1990, An importance quantification technique in uncertainty analysis for computer models. In Uncertainty Modeling and Analysis. Proceedings., First International Symposium on (pp. 398-403). IEEE.

Jansen K., Leovey H., Nube A., Griewank A., Mueller-Preussker M., 2014, A first look of quasi-Monte Carlo for lattice field theory problems. Computer Physics Communication 185:948–959

Jansen M., 1999. Analysis of variance designs for model output. Computer Physics Communication 117:25–43

Jin R, Chen W, Sudjianto A., 2005. An efficient algorithm for constructing optimal design of computer experiments. Journal of Statistical Planning and Inference. 134(1):268–287.

Kalla S, White CD, 2007. Efficient design of reservoir simulation studies for development and optimization. SPE Reserv Eval Eng 10(6):629–637

King, D.M. and Perera, B.J.C., 2013. Morris method of sensitivity analysis applied to assess the importance of input variables on urban water supply yield–A case study. Journal of hydrology, 477, pp.17-32.

Koehler JR, Owen AB, 1996. Computer experiments. Handbook of Statistics. 13:261–308.

Kohonen, T., 1982. Self-organized formation of topologically correct feature maps. Biol. Cybern. 43 (1), 59–69.

Levy, G., 2011, an introduction to quasi-random numbers, Numerical Algorithms Group Ltd. Retrieved from https://www.nag.com/IndustryArticles/introduction_to_quasi_random_numbers

Li, L., Boucher, A., Caers, J., 2014. SGEMS-UQ: an uncertainty quantification toolkit for SGEMS. Comput. Geosci. 62, 12–24.

Lilburne, L., Tarantola, S., 2009. Sensitivity analysis of spatial models. Int. J. Geogr. Inf. Sci. 23 (2), 151–168.

Martinius, A.W., Næss, A., 2005. Uncertainty analysis of fluvial outcrop data for stochastic reservoir modelling. Pet. Geosci. 11 (3), 203–214.

McKay, M.D., Beckman, R.J. and Conover, W.J., 2000. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics, 42(1), pp.55-61.

Mirzabozorg, A., 2015. Incorporation of Engineering Knowledge in History Matching, Optimization, and Uncertainty Assessment Frameworks with Application to the SAGD Process. PhD dissertation, University of Calgary.

Morris, M.D., 1991, Factorial Sampling Plans for Preliminary Computational Experiments. Technometrics. 33: 161–174.

Myers, D.E., and Journel, A., 1990. Variograms with zonal anisotropies and noninvertible kriging systems. Math. Geology, Vol. 22, N 7, p. 779-785.

Myers, R.H., Montgomery, D.C., 2002. Response Surface Methodology. Process and Product Optimization Using Designed Experiments. John Wiley & Sons Inc., NY.

Nguyen, N.T., Chen, Z., Dang, C.T., Nghiem, L.X., Yang, C., Bourgoult, G. and Li, H., 2015. Integrated Modeling for Assisted History Matching and Robust

Optimisation in Mature Reservoirs. In SPE/IATMI Asia Pacific Oil & Gas Conference and Exhibition. Society of Petroleum Engineers.

Oladyshkin, S., de Barros, F.P.J., Nowak, W., 2012. Global sensitivity analysis: a flexible and efficient framework with an example from stochastic hydro- geology. Adv. Water Resour. 37, 10–22.

Oreskes, N., Shrader-Frechette, K., Belitz, K., 1994. Verification, validation, and confirmation of numerical models in the earth sciences. Science 263 (5147), 641–646.

Pan, F., Zhu, J., Ye, M., Pachepsky, Y.A., Wu, Y.S., 2011. Sensitivity analysis of unsaturated flow and contaminant transport with correlated parameters. J. Hydrol. 397 (3), 238–249.

Pappenberger, F., Beven, K.J., Ratto, M., Matgen, P., 2008. Multi-method global sensitivity analysis of flood inundation models. Adv. Water Resour. 31 (1), 1–14.

Park, H., Scheidt, C., Fenwick, D., Boucher, A., Caers, J., 2013. History matching and uncertainty quantification of facies models with multiple geological interpretations. Comput. Geosci. 17 (4), 609–621.

Park, K., 2011. Modeling uncertainty in metric space. Stanford University.

Peters, L., Arts, R., Brouwer, G., Geel, C., Cullick, S., Lorentzen, R. J., … Reynolds, A., 2009, Results of the Brugge Benchmark Study for Flooding Optimization and History Matching. Society of Petroleum Engineers.

Ringrose, P., Bentley, M., 2015. Reservoir Model Design. Springer Netherlands, Dordrecht, The Netherlands.

Rohmer, J., 2014. Combining meta-modeling and categorical indicators for global sensitivity analysis of long-running flow simulators with spatially dependent inputs. Comput. Geosci. 18 (2), 171–183.

Rousseeuw PJ, 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 20:53–65

Ryan, T.P., 2009. Modern Regression Methods, 2nd ed. John Wiley and Sons, Chicheseter, UK.

Saltelli A, Chan K, Scott EM (eds), 2000. Sensitivity analysis. Wiley, New York

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., Tarantola, S., 2008. Global Sensitivity Analysis: The Primer. John Wiley and Sons, Chicheseter, UK.

Saltelli, A., Tarantola, S., Campolongo, F., Ratto, M., 2004. Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models. John Wiley and Sons, Chicheseter, UK.

Sarma, P., Durlofsky, L.J., Aziz, K., 2008. Kernel principal component analysis for efficient, differentiable parameterization of multipoint geostatistics. Math. Geosci. 40 (1), 3–32.

Sarma, P., Xie, J., 2011. Efficient and robust uncertainty quantification in reservoir simulation with polynomial chaos expansions and non-intrusive spectral

projection. In: Proceedings of the SPE Reservoir Simulation Symposium. Wood lands, Texas.

Sarma, P., Yang, C., Xie, J., Chen, W., & Wen, X.-H., 2015. Identification of "Big Hitters" with Global Sensitivity Analysis for Improved Decision Making Under Uncertainty. Society of Petroleum Engineers.

Scheidt, C., Caers, J., 2009. Representing spatial uncertainty using distances and kernels. Math. Geosci. 41 (4), 397–419.

Scheidt, C., Fernandes, A., Paola, C., Caers, J. 2015a. Can geostatistical models represent nature's variability? An analysis using flume experiments. In: Proceedings of the Petroleum Geostatistics. Biarritz, France.

Scheidt, C., Jeong, C., Mukerji, T., Caers, J., 2015b. Probabilistic falsification of prior geologic uncertainty with seismic amplitude data: application to a turbidite reservoir case. Geophysics 80 (5), M89-M12.

Sobol', I.M., 2001. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. Math. Comput. Simul. 55 (1), 271–280.

Spear, R.C., Grieb, T.M., Shang, N., 1994. Parameter uncertainty and interaction in complex environmental models. Water Resour. Res. 30 (11), 3159–3169.

Spear, R.C., Hornberger, G.M., 1980. Eutrophication in peel inlet—II. Identification of critical uncertainties via generalized sensitivity analysis. Water Resour. Res. 14 (1), 43–49.

Sudret, B., 2008. Global sensitivity analysis using polynomial chaos expansions. Reliab. Eng. Syst. Saf. 93 (7), 964–979.

Suzuki, S., Caumon, G., Caers, J., 2008. Dynamic data integration for structural modeling: model screening approach using a distance-based model parameterization. Comput. Geosci. 12 (1), 105–119.

Tang, Y., Reed, P., Wagener, T., Van Werkhoven, K., 2007. Comparing sensitivity analysis methods to advance lumped watershed model identification and evaluation. Hydrol. Earth Syst. Sci. Discuss. 11 (2), 793–817.

Thiele, M.R. and Batycky, R.P., 2016, Evolve: A Linear Workflow for Quantifying Reservoir Uncertainty. In SPE Annual Technical Conference and Exhibition. Society of Petroleum Engineers.

Van Griensven, A., Meixner, T., Grunwald, S., Bishop, T., Diluzio, M., Srinivasan, R., 2006. A global sensitivity analysis tool for the parameters of multi-variable catchment models. J. Hydrol. 324 (1), 10–23.

Volkova, E., Iooss, B., Van Dorpe, F., 2008. Global sensitivity analysis for a numerical model of radionuclide migration from the RRC "Kurchatov Institute" radwaste disposal site. Stoch. Environ. Res. Risk Assess. 22 (1), 17–31.

Wainwright, H.M., Finsterle, S., Jung, Y., Zhou, Q., Birkholzer, J.T., 2014. Making sense of global sensitivity analyses. Comput. Geosci. 65, 84–94.

Wei, P., Lu, Z., Song, J., 2015. Variable importance analysis: a comprehensive review. Reliab. Eng. Syst. Saf. 142, 399–432.

Westfall PH, Young SS, 1993. Resampling-based multiple testing: examples and methods for p-value adjustment. Wiley, New York

White CD, Willis BJ, Narayanan K, Dutton SP, 2001. Identifying and estimating significant geologic parameters with experimental design. SPE J 6(3):311–324

Wyss, D. G., and Jorgensen, H. K., 1998. A User's Guide to LHS: Sandia's Latin Hypercube Sampling Software, SAND98-0210

Ye KQ, Li W., Sudjianto A., 2000. Algorithmic construction of optimal symmetric Latin hypercube designs. Journal of Statistical Planning and Inference. 90(1):149–159.

Zabalza-Mezghani, I., Manceau, E., Feraille, M., Jourdan, A., 2004. Uncertainty management: from geological scenarios to production scheme optimization. J. Pet. Sci. Eng. 44 (1), 11–25.

Zhang, D., Lu, Z., 2004. An efficient, high-order perturbation approach for flow in random media via Karhunen–Loeve and polynomial expansions. J. Comput. Phys. 194 (2), 773–794.

APPENDIX A: Proof of Equations. (3.26) and (3.27)

By applying the differential operator $\partial/\partial {w_{\alpha\beta}}^{(M-K-1)}$ to Eq. (3.21) and employing the chain differentiation rule, it is straightforward to demonstrate that $\forall M, K$, where $K \in (0, M-2), L_0 = D$

$$\frac{\partial \tilde{Y}}{\partial {w_{\alpha\beta}}^{(M-K-1)}} = \theta(L_{M-K-2} + 1 - \alpha)\,\theta(L_{M-K-1} - \beta)\,{H_\alpha}^{(M-K-2)} F({H_\beta}^{(M-K-1)})\,\frac{\partial \tilde{Y}}{\partial {H_\beta}^{(M-K-1)}} \quad (A.1)$$

where $\theta(x)$ is Heaviside's step-function. At the same time, Eq. (3.27) can be re-written in the following form:

$${G_\beta}^{(M-K)} = \theta(L_{M-K} - \beta)\,F({H_\beta}^{(M-K)})\,\frac{\partial \tilde{Y}}{\partial {H_\beta}^{(M-K)}} \quad (A.2)$$

On the other hand, the derivative $\partial \tilde{Y}/\partial {H_\beta}^{(M-K-1)}$ can be represented as

$$\frac{\partial \tilde{Y}}{\partial {H_\beta}^{(M-K-1)}} = \sum_{\rho=1}^{L_{M-K}} \frac{\partial {H_\rho}^{(M-K)}}{\partial {H_\beta}^{(M-K-1)}}\,\frac{\partial \tilde{Y}}{\partial {H_\rho}^{(M-K)}} = \sum_{\rho=1}^{L_{M-K}} {w_{\beta\rho}}^{(M-K)}\,{G_\rho}^{(M-K)} \quad (A.3)$$

Substituting Eq. (A.3) into Eq. (A.1) we arrive at the final expression

$$\frac{\partial \tilde{Y}}{\partial w_{\alpha\beta}{}^{(M-K-1)}}$$

$$= \theta(L_{M-K-2} + 1$$

$$- \alpha)\, \theta(L_{M-K-1}$$

$$- \beta)\, H_{\alpha}{}^{(M-K-2)} F\left(H_{\beta}{}^{(M-K-1)}\right) \sum_{\rho=1}^{L_{M-K}} w_{\beta\rho}{}^{(M-K)}\, G_{\rho}{}^{(M-K)} \qquad \text{(A.4)}$$

or, in terms of Eqs. (3.26) and (3.27):

$$\frac{\partial \tilde{Y}}{\partial w_{\alpha\beta}{}^{(M-K-1)}} = H_{\alpha}{}^{(M-K-2)} G_{\beta}{}^{(M-K-1)} \qquad \text{(A.5)}$$

Where

$$G_{\beta}{}^{(M-K-1)} = F\left(H_{\alpha}{}^{(M-K-1)}\right) \sum_{\mu=1}^{N_{M-K}} w_{\beta\mu}{}^{(M-K)}\, G_{\beta}{}^{(M-K)} \qquad \text{(A.6)}$$

In other words, an assumption of validity of Eqs. (3.26) and (3.27) for any given hidden layer $M - K$ leads to the conclusion that it is also valid for the previous hidden layer $M - K - 1$. According to the principle of Mathematical Induction, it proves its validity for all hidden layers.

APPENDIX B: Empirical CDF curves for parameters of CSS pilot case study

5.4 B.1. Cumulative Oil Production

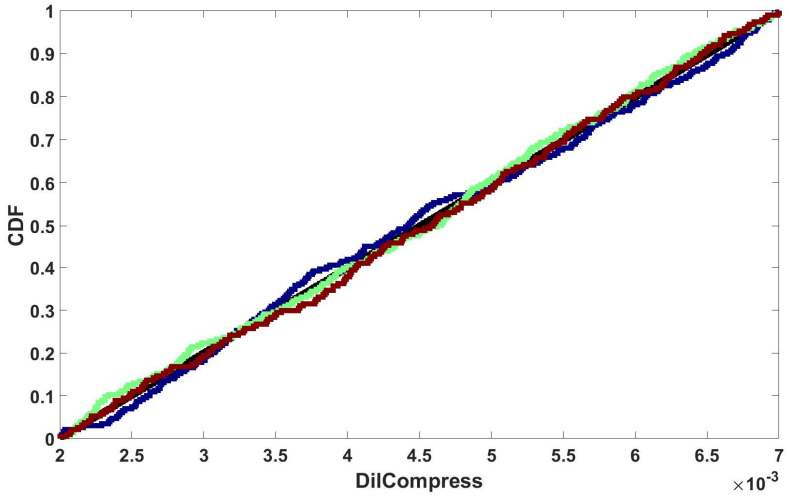


Figure B.1: Empirical CDF curves for parameter 'DilCompress'

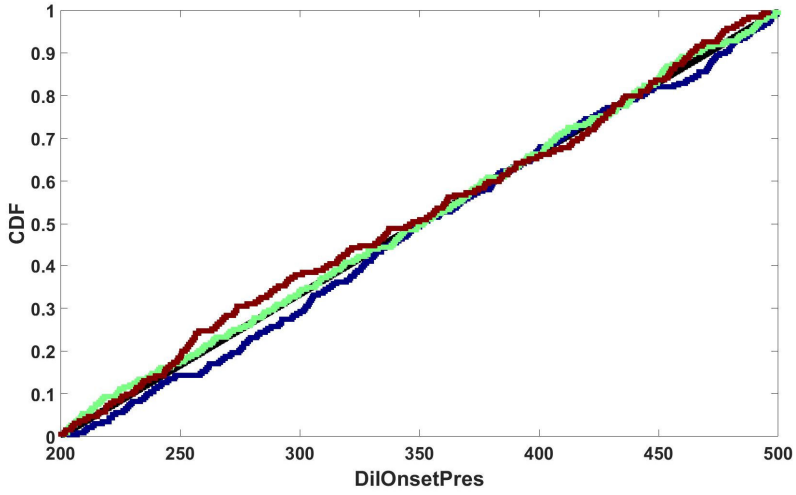


Figure B.2: Empirical CDF curves for parameter 'DilOnsetPres'

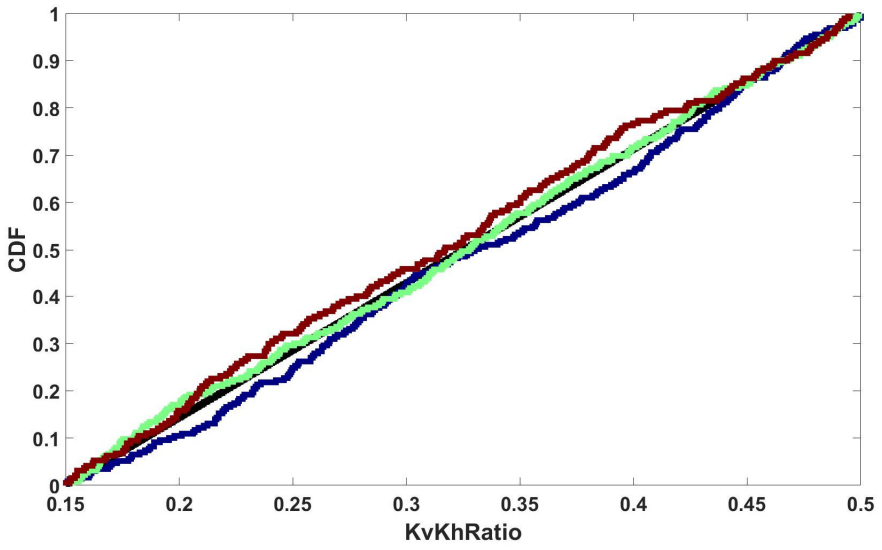


Figure B.3: Empirical CDF curves for parameter 'KvKhRatio'

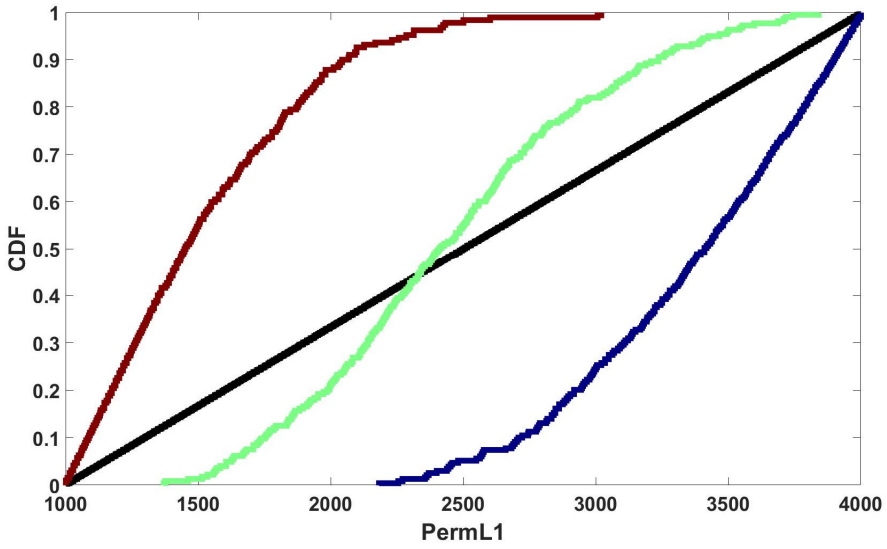


Figure B.4: Empirical CDF curves for parameter 'PermL1'

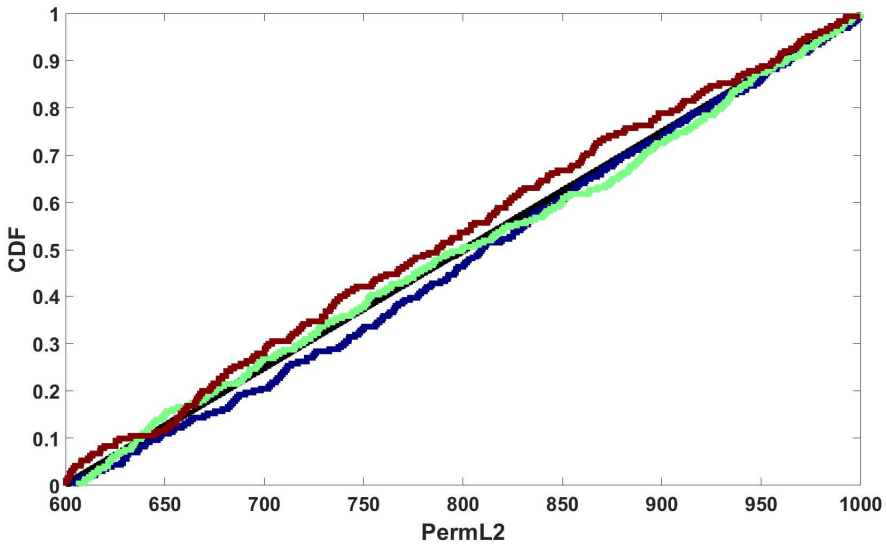


Figure B.5: Empirical CDF curves for parameter 'PermL2'

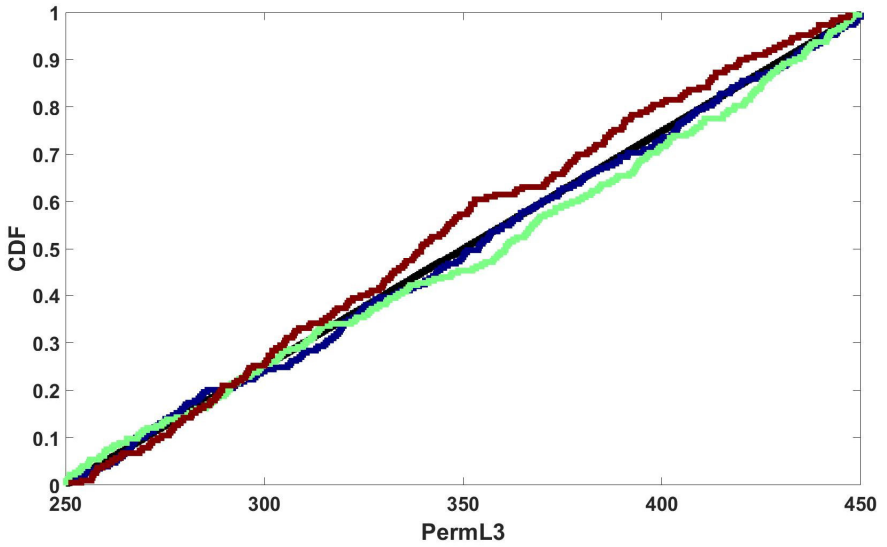


Figure B.6: Empirical CDF curves for parameter 'PermL3'

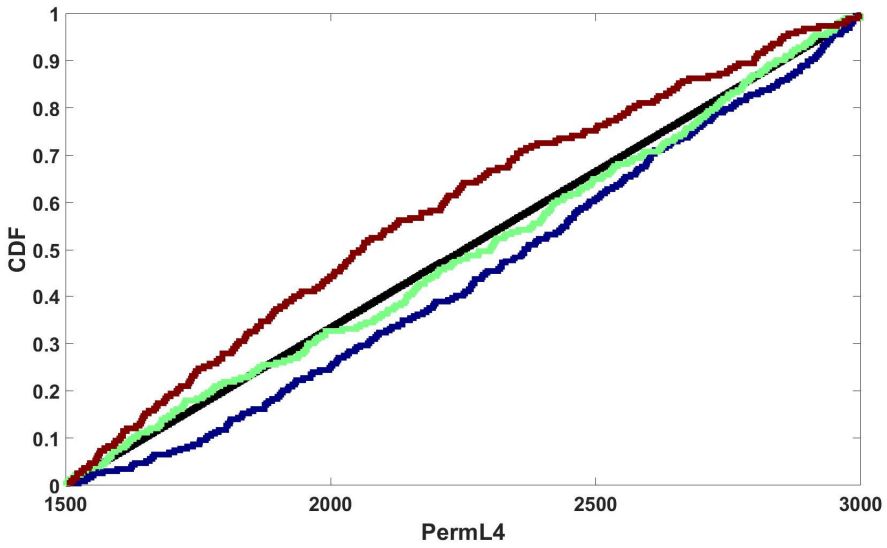


Figure B.7: Empirical CDF curves for parameter 'PermL4'

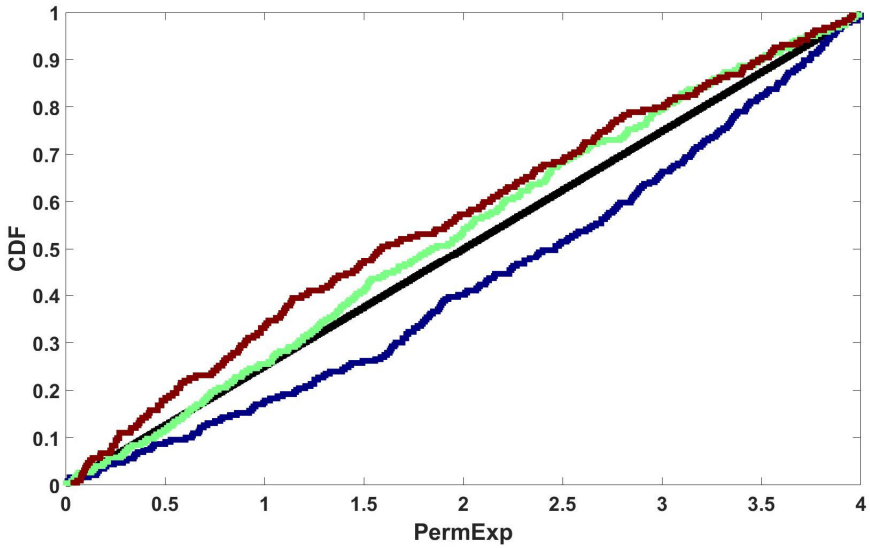


Figure B.8: Empirical CDF curves for parameter 'PermExp'

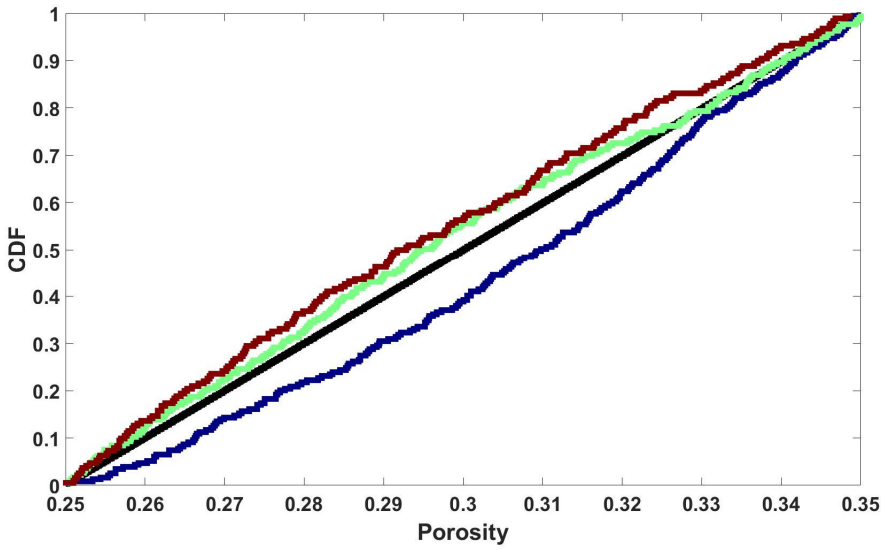


Figure B.9: Empirical CDF curves for parameter 'Porosity'

5.5 B.2. Cumulative Water Production

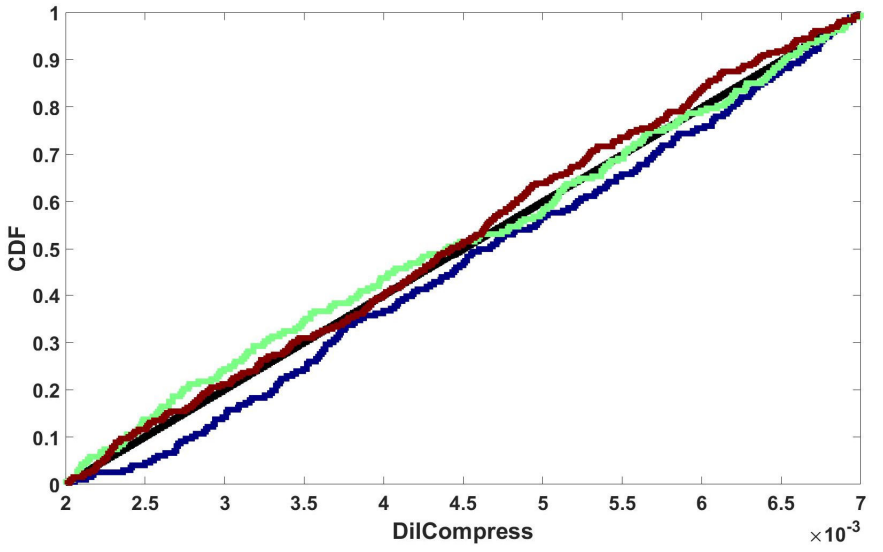


Figure B.10: Empirical CDF curves for parameter 'DilCompress'

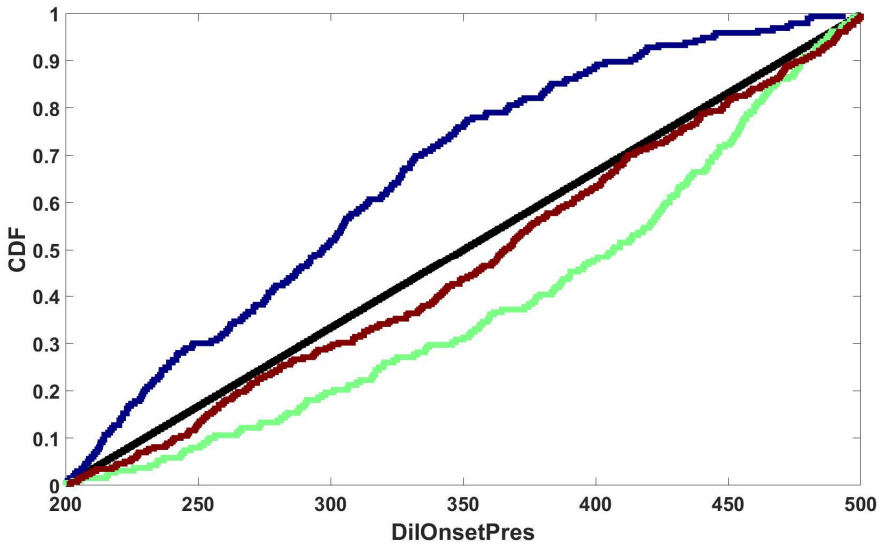


Figure B.11: Empirical CDF curves for parameter 'DilOnsetPres'

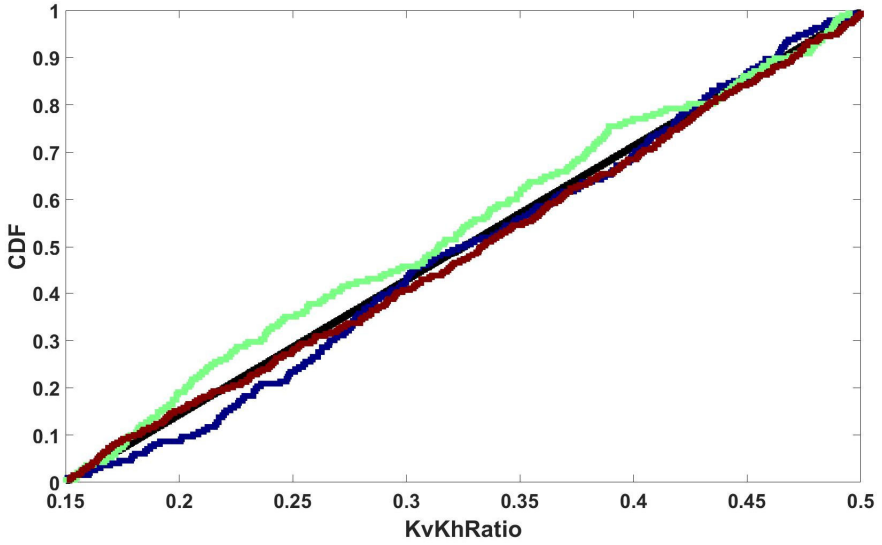


Figure B.12: Empirical CDF curves for parameter 'KvKhRatio'

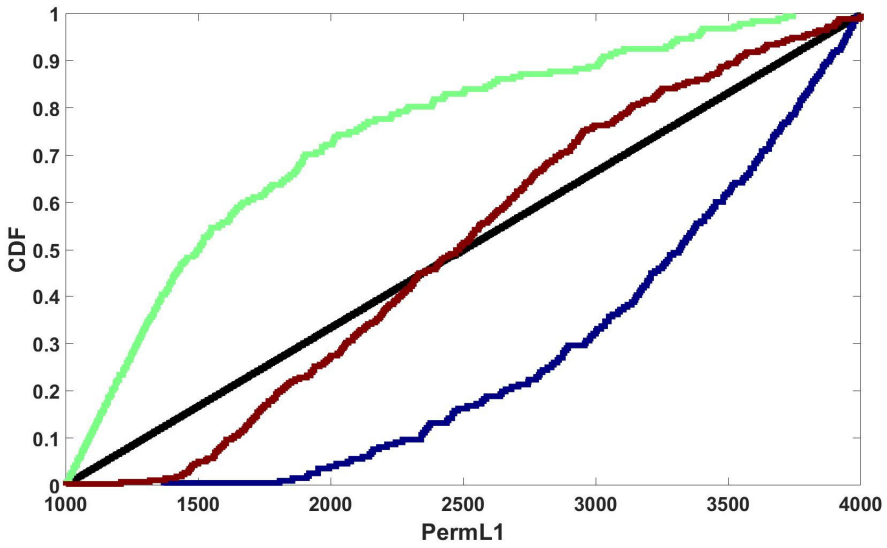


Figure B.13: Empirical CDF curves for parameter 'PermL1'

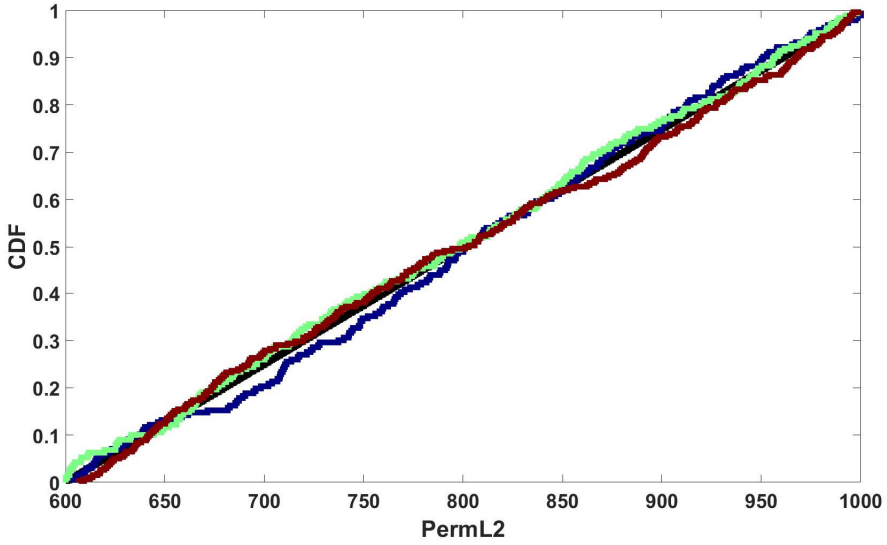


Figure B.14: Empirical CDF curves for parameter 'PermL2'

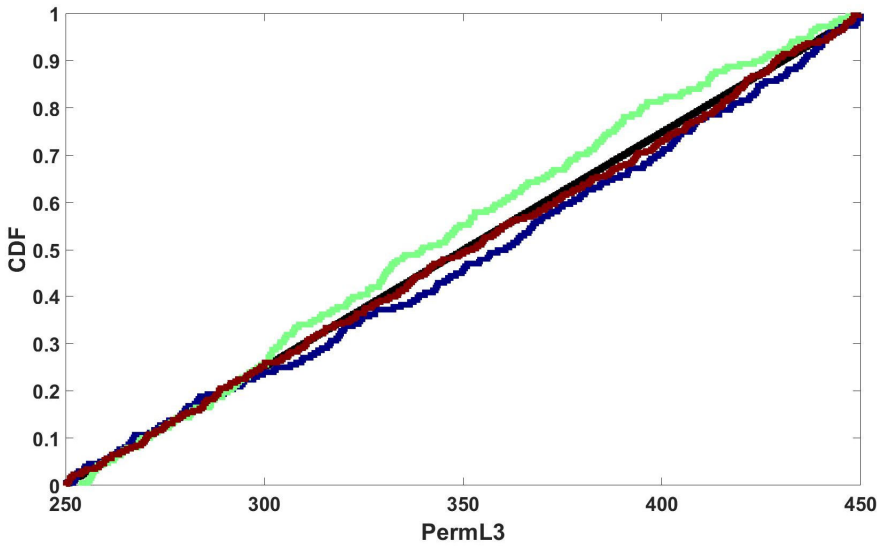


Figure B.15: Empirical CDF curves for parameter 'PermL3'

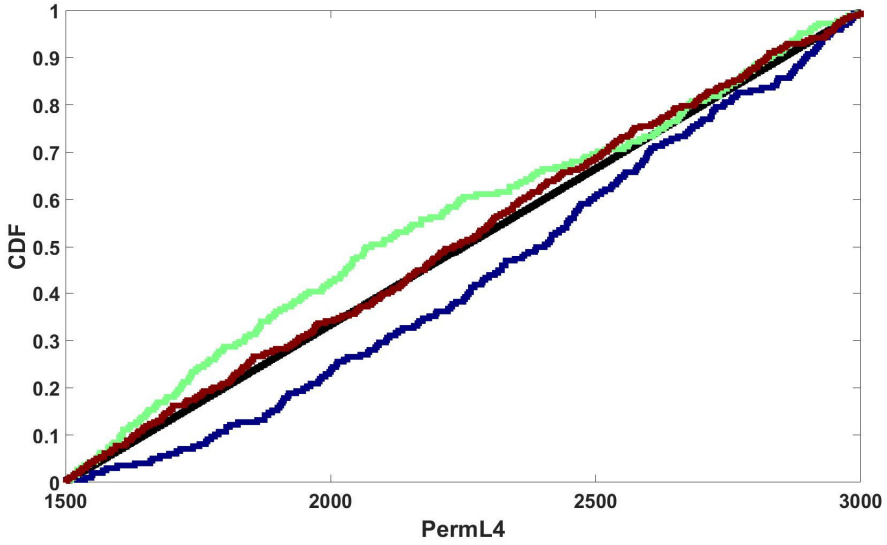


Figure B.16: Empirical CDF curves for parameter 'PermL4'

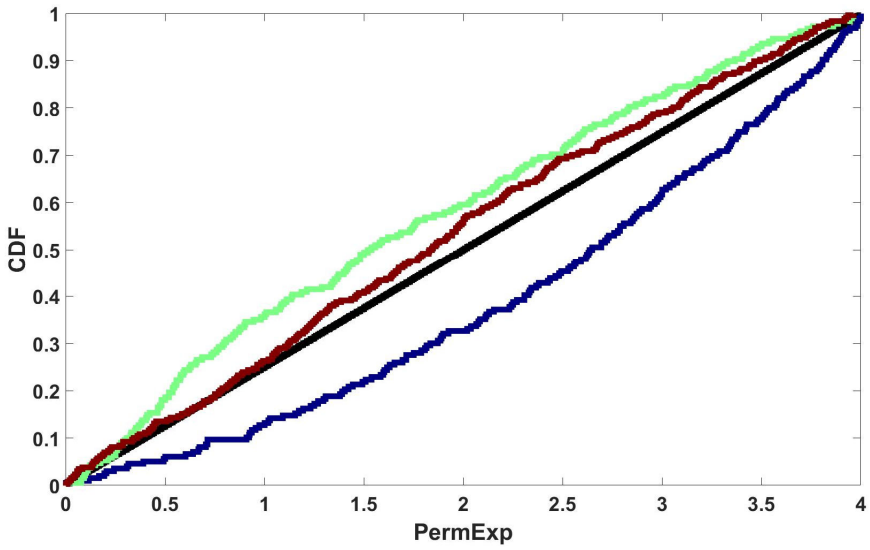


Figure B.17: Empirical CDF curves for parameter 'PermExp'

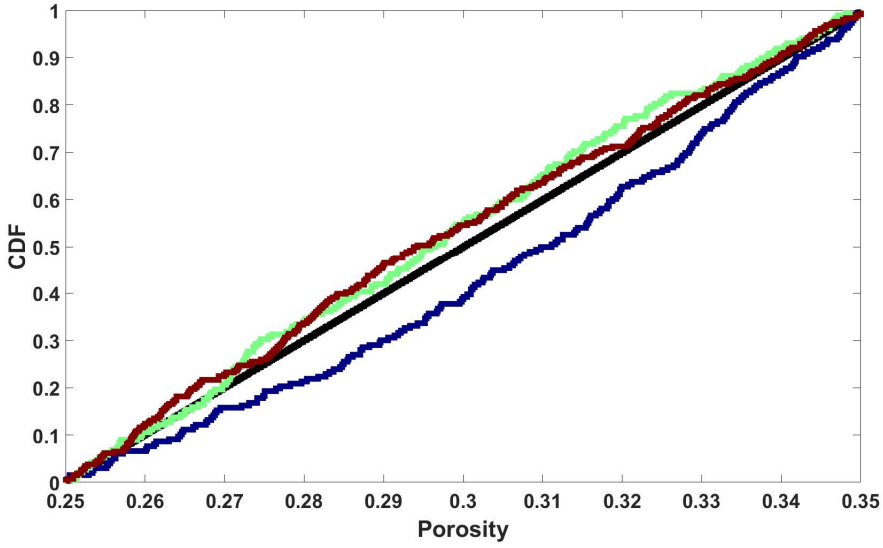


Figure B.18: Empirical CDF curves for parameter 'Porosity'

5.6 B.3. Cumulative Steam Injection

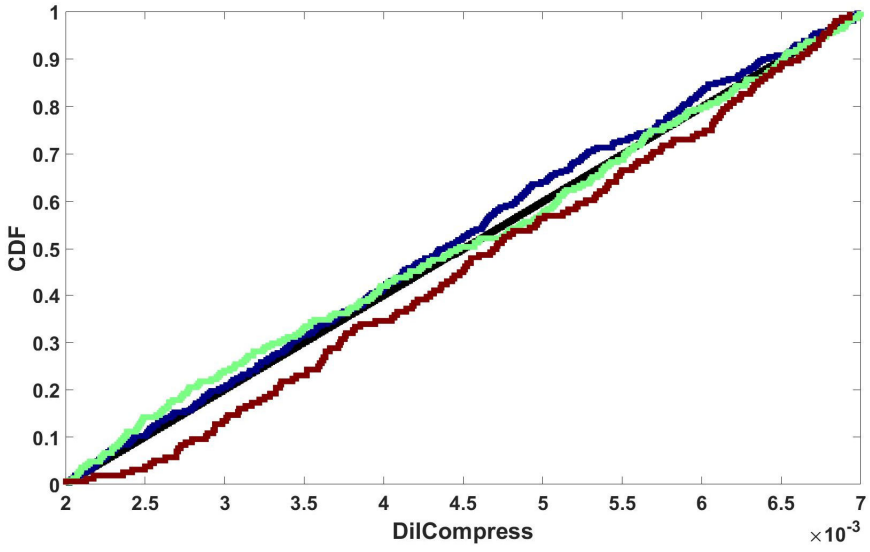


Figure B.19: Empirical CDF curves for parameter 'DilCompress'

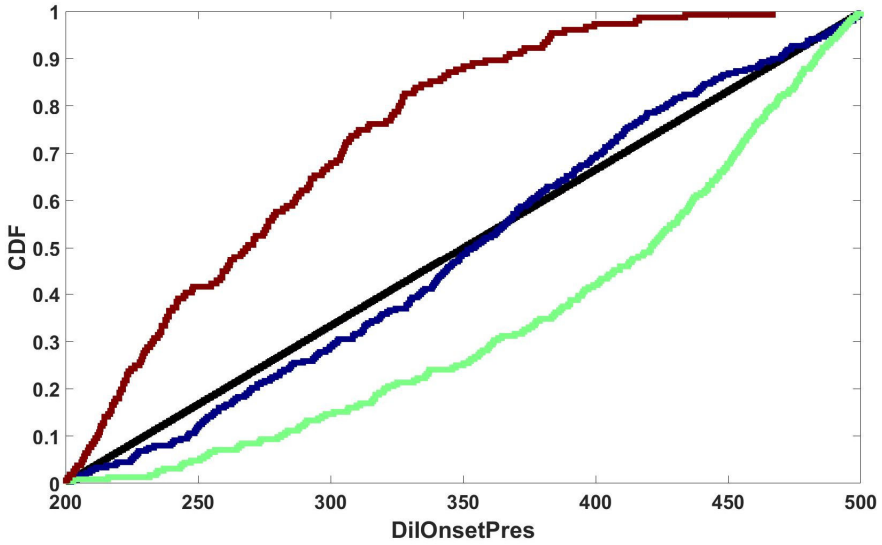


Figure B.20: Empirical CDF curves for parameter 'DilOnsetPres'

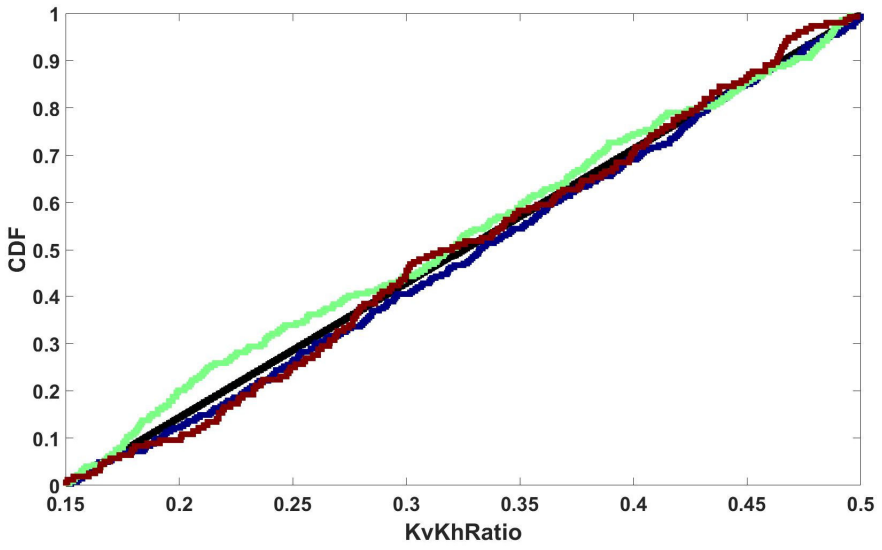


Figure B.21: Empirical CDF curves for parameter 'KvKhRatio'

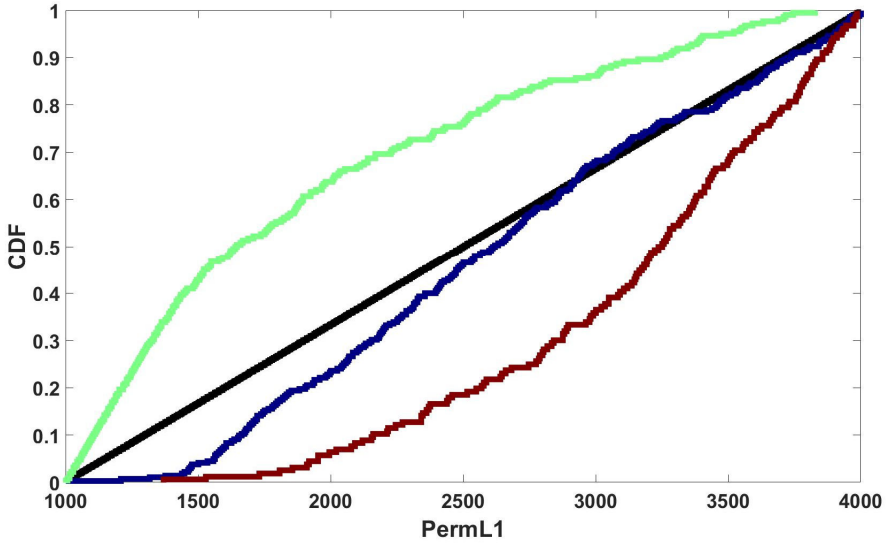


Figure B.22: Empirical CDF curves for parameter 'PermL1'

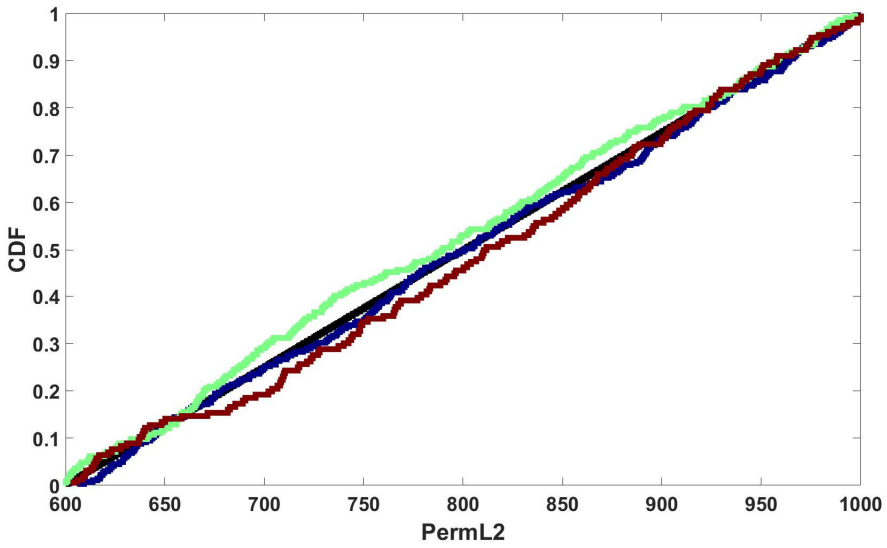


Figure B.23: Empirical CDF curves for parameter 'PermL2'

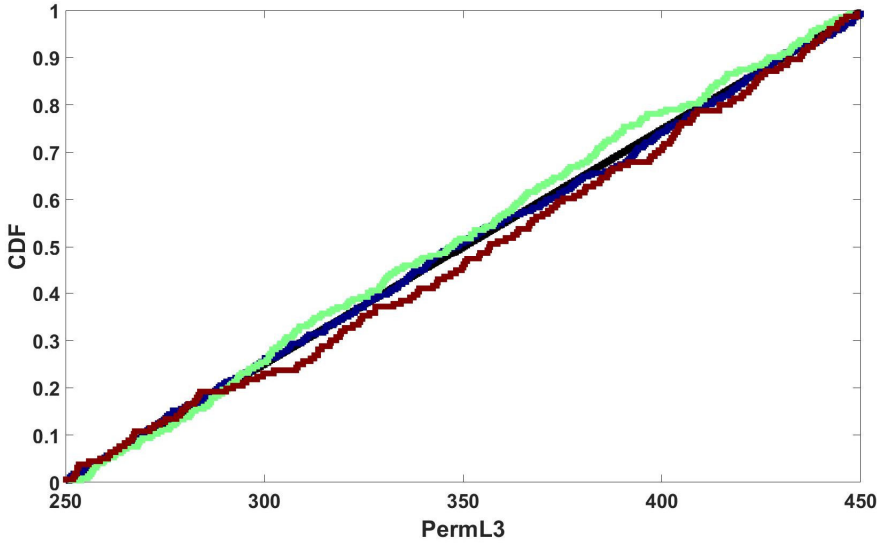


Figure B.24: Empirical CDF curves for parameter 'PermL3'

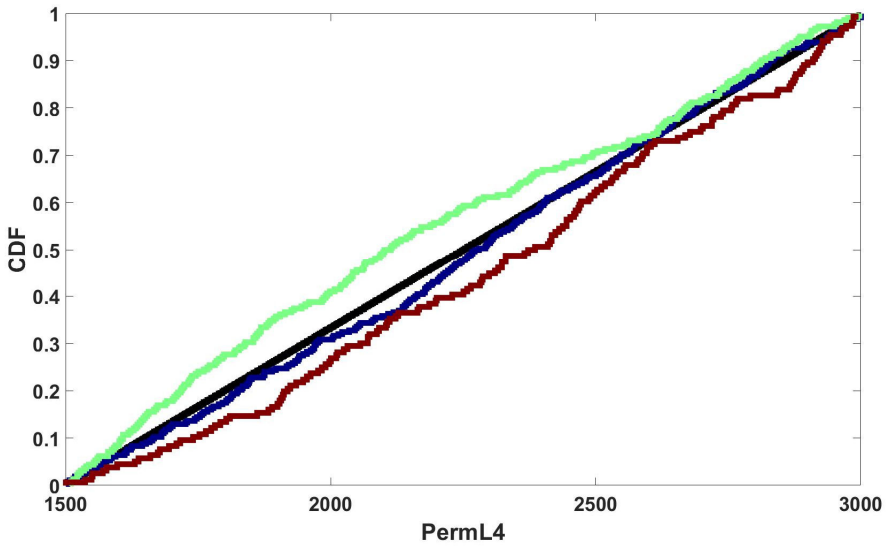


Figure B.25: Empirical CDF curves for parameter 'PermL4'

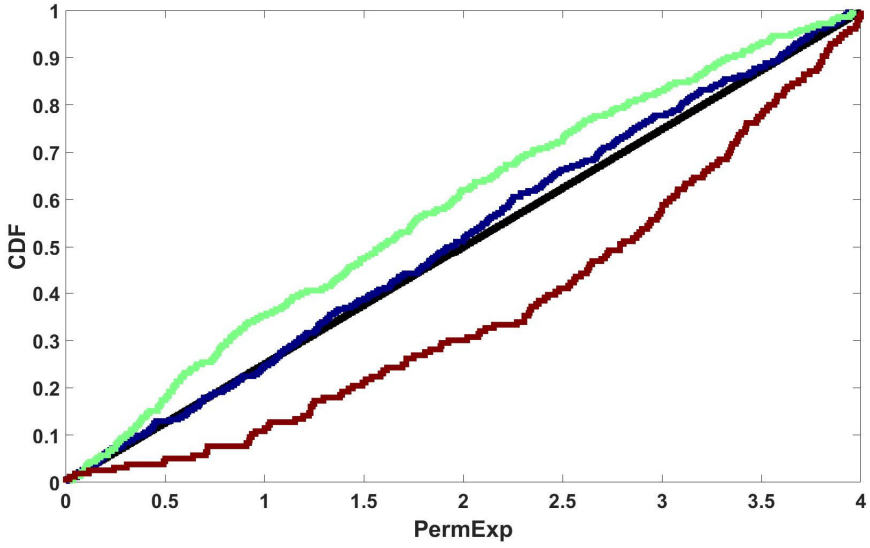


Figure B.26: Empirical CDF curves for parameter 'PermExp'

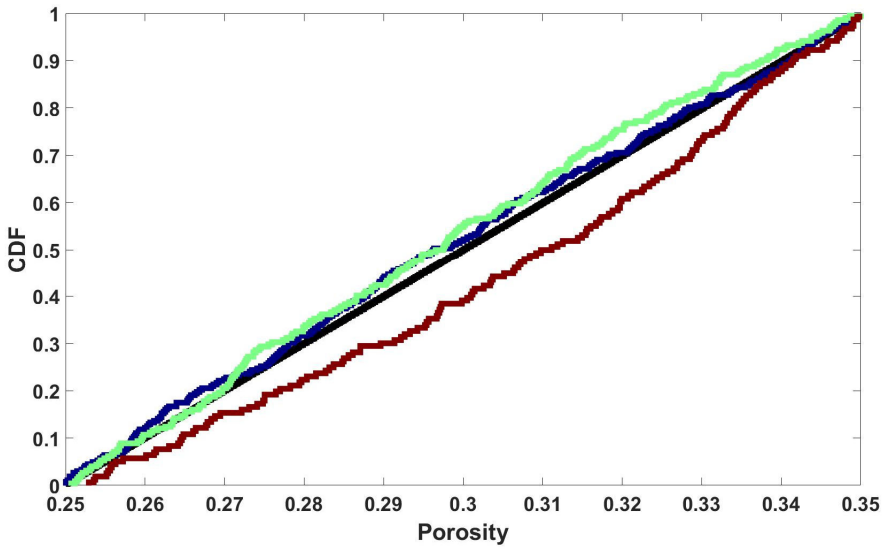


Figure B.27: Empirical CDF curves for parameter 'Porosity'

APPENDIX C: Copyright Permissions

JOHN WILEY AND SONS LICENSE
TERMS AND CONDITIONS

Jan 07, 2018

This Agreement between University of Calgary -- Ali Karami Moghadam ("You") and John Wiley and Sons ("John Wiley and Sons") consists of your license details and the terms and conditions provided by John Wiley and Sons and Copyright Clearance Center.

License Number	4253780163599
License date	Dec 21, 2017
Licensed Content Publisher	John Wiley and Sons
Licensed Content Publication	Wiley oBooks
Licensed Content Title	Index
Licensed Content Author	Jef Caers
Licensed Content Date	Jun 1, 2011
Licensed Content Pages	5
Type of use	Dissertation/Thesis
Requestor type	University/Academic
Format	Print and electronic
Portion	Figure/table
Number of figures/tables	3
Original Wiley figure/table number(s)	Figures 9.10, 9.11, 10.10
Will you be translating?	No
Title of your thesis / dissertation	Global Sensitivity Analysis for Covering Reservoir Geological and Flow Uncertainty
Expected completion date	Dec 2017
Expected size (number of pages)	290
Requestor Location	University of Calgary 2500 University Dr NW

JOHN WILEY AND SONS LICENSE
TERMS AND CONDITIONS

Jan 07, 2018

This Agreement between University of Calgary -- Ali Karami Moghadam ("You") and John Wiley and Sons ("John Wiley and Sons") consists of your license details and the terms and conditions provided by John Wiley and Sons and Copyright Clearance Center.

License Number	4253790283791
License date	Dec 21, 2017
Licensed Content Publisher	John Wiley and Sons
Licensed Content Publication	Wiley oBooks
Licensed Content Title	Index
Licensed Content Author	Jef Caers
Licensed Content Date	Jun 1, 2011
Licensed Content Pages	5
Type of use	Dissertation/Thesis
Requestor type	University/Academic
Format	Print and electronic
Portion	Text extract
Number of Pages	1
Will you be translating?	No
Title of your thesis / dissertation	Global Sensitivity Analysis for Covering Reservoir Geological and Flow Uncertainty
Expected completion date	Dec 2017
Expected size (number of pages)	290
Requestor Location	University of Calgary 2500 University Dr NW Calgary, AB T2N 1N4 Canada Attn: University of Calgary
Publisher Tax ID	EU826007151
Billing Type	Invoice
Billing Address	University of Calgary 2500 University Dr NW

SPRINGER NATURE LICENSE
TERMS AND CONDITIONS

Jan 07, 2018

This Agreement between University of Calgary -- Ali Karami Moghadam ("You") and Springer Nature ("Springer Nature") consists of your license details and the terms and conditions provided by Springer Nature and Copyright Clearance Center.

License Number	4263910428829
License date	Jan 07, 2018
Licensed Content Publisher	Springer Nature
Licensed Content Publication	Springer eBook
Licensed Content Title	A Review on Global Sensitivity Analysis Methods
Licensed Content Author	Bertrand Iooss, Paul Lemaître
Licensed Content Date	Jan 1, 2015
Type of Use	Thesis/Dissertation
Requestor type	academic/university or research institute
Format	print and electronic
Portion	figures/tables/illustrations
Number of figures/tables/illustrations	2
Will you be translating?	no
Circulation/distribution	>50,000
Author of this Springer Nature content	no
Title	Global Sensitivity Analysis for Covering Reservoir Geological and Flow Uncertainty
Instructor name	n/a
Institution name	n/a
Expected presentation date	Dec 2017
Portions	Figures 6, 7
Requestor Location	University of Calgary 2500 University Dr NW Calgary, AB T2N 1N4 Canada Attn: University of Calgary
Billing Type	Invoice
Billing Address	University of Calgary 2500 University Dr NW

JOHN WILEY AND SONS LICENSE
TERMS AND CONDITIONS

Jan 07, 2018

This Agreement between University of Calgary -- Ali Karami Moghadam ("You") and John Wiley and Sons ("John Wiley and Sons") consists of your license details and the terms and conditions provided by John Wiley and Sons and Copyright Clearance Center.

License Number	4263920601272
License date	Jan 07, 2018
Licensed Content Publisher	John Wiley and Sons
Licensed Content Publication	Wiley Books
Licensed Content Title	Global Sensitivity Analysis: The Primer
Licensed Content Author	Andrea Saltelli Debora Gatelli Francesca Campolongo Jessica Cariboni Marco Ratto Michaela Saisana Stefano Tarantola Terry Andres
Licensed Content Date	Feb 1, 2008
Licensed Content Pages	1
Type of use	Dissertation/Thesis
Requestor type	University/Academic
Format	Print and electronic
Portion	Figure/table
Number of figures/tables	2
Original Wiley figure/table number(s)	Figures 3.2, 3.4
Will you be translating?	No
Title of your thesis / dissertation	Global Sensitivity Analysis for Covering Reservoir Geological and Flow Uncertainty
Expected completion date	Dec 2017
Expected size (number of pages)	290
Requestor Location	University of Calgary 2500 University Dr NW Calgary, AB T2N 1N4 Canada Attn: University of Calgary

SOCIETY OF PETROLEUM ENGINEERS LICENSE
TERMS AND CONDITIONS

Jan 09, 2018

This Agreement between University of Calgary -- Ali Karami Moghadam ("You") and Society of Petroleum Engineers ("Society of Petroleum Engineers") consists of your license details and the terms and conditions provided by Society of Petroleum Engineers and Copyright Clearance Center.

License Number	4265021010518
License date	Jan 09, 2018
Licensed Content Publisher	Society of Petroleum Engineers
Licensed Content Publication	SPE Proceedings
Licensed Content Title	Integrated Modeling for Assisted History Matching and Robust Optimisation in Mature Reservoirs
Licensed Content Author	Ngoc T. B. Nguyen, University of Calgary;Zhangxin Chen, University of Calgary;Cuong T. Q. Dang, Computer Modelling Group Ltd. et al
Licensed Content Date	Jan 1, 2015
Type of Use	Thesis/Dissertation
Requestor type	employer of the author of the original work
SPE member	no
Format	print and electronic
Portion	figures/tables/images
Number of figures/tables/images	3
Will you be translating?	no
Distribution	100000
Order reference number	
Title of your thesis / dissertation	Global Sensitivity Analysis for Covering Reservoir Geological and Flow Uncertainty
Expected completion date	Dec 2017
Estimated size (number of pages)	290
Requestor Location	University of Calgary 2500 University Dr NW Calgary, AB T2N 1N4 Canada Attn: University of Calgary

20/10/2017

Dear Dr. Fedutenko:

I am in the process of submitting my PhD thesis to the Faculty of Graduate Studies at University of Calgary. I would like your permission to include the material we have worked on regarding the single-layer and multi-layer neural network proxy models for sensitivity analysis in this project.

The material will be used in Chapter three of my thesis, under the section "Neural Network Based Surrogate Modelling for Sensitivity Analysis". It will be available to the readers of my thesis after being archived at University of Calgary library.

Your permission confirms that you hold the right to grant the permission requested here. Permission includes non-exclusive world rights to use the material and will not limit any future publications-including future editions and revisions-by you or others authorized by you.

I would greatly appreciate your consent to my request. If you require any additional information, please do not hesitate to contact me. If you agree with the terms as described above, please sign the below.

Sincerely,

Ali Karami Moghadam

Permission granted for the use of the material as described above: