DISSERTATION

SPATIAL MODELS WITH APPLICATIONS IN COMPUTER EXPERIMENTS

Submitted by

Ke Wang

Department of Statistics

In partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Summer 2008

COLORADO STATE UNIVERSITY

June 2, 2008

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY KE WANG ENTITLED SPATIAL MODELS WITH APPLICATIONS IN COMPUTER EXPERIMENTS BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work

Richard A. Davis (Adviser)

F. Jay Breidt (Co-Adviser)

Hari K. Iyer (Committee Member)

Donald Estep (Outside Committee Member)

F. Jay Breidt (Department Head)

ABSTRACT OF DISSERTATION

SPATIAL MODELS WITH APPLICATIONS IN COMPUTER EXPERIMENTS

Often, a deterministic computer response is modeled as a realization from a stochastic process such as a Gaussian random field. Due to the limitation of stationary Gaussian process (GP) in inhomogeneous smoothness, we consider modeling a deterministic computer response as a realization from a stochastic heteroskedastic process (SHP), a stationary non-Gaussian process. Conditional on a latent process, the SHP has non-stationary covariance function and is a non-stationary GP. As such, the sample paths of this process exhibit greater variability and hence offer more modeling flexibility than those produced by a traditional GP model. We use maximum likelihood for inference in the SHP model, which is complicated by the high dimensionality of the latent process. Accordingly, we develop an importance sampling method for likelihood computation and use a low-rank kriging approximation to reconstruct the latent process. Responses at unobserved locations can be predicted using empirical best predictors or by empirical best linear unbiased predictors. In addition, prediction error variances are obtained. The SHP model can be used in an active learning context, adaptively selecting new locations that provide improved estimates of the response surface. Estimation, prediction, and adaptive sampling with the SHP model are illustrated with several examples.

Our spatial model can be adapted to model the first partial derivative process. The derivative process provides additional information about the shape and smoothness of the underlying deterministic function and can assist in the prediction of responses at unobserved sites. The unconditional correlation function for the

derivative process presents some interesting properties, and can be used as a new class of spatial correlation functions. For parameter estimation, we propose to use a similar strategy to develop an importance sampling technique to compute the joint likelihood of responses and derivatives. The major difficulties of bringing in derivative information are the increase in the dimensionality of the latent process and the numerical problems of inverting the enlarged covariance matrix. Some possible ways to utilize this information more efficiently are proposed.

Ke Wang
Department of Statistics
Colorado State University
Fort Collins, Colorado 80523
Summer 2008

ACKNOWLEDGEMENTS

There are several people, without whose efforts and support, I would not be here. I would first like to thank my advisers, Professor Richard Davis and Professor Jay Breidt. I could not have asked for better advisers. The ideas and guidance they gave to me made this dissertation possible. Their kindness, gentle demeanor and flexibility were very meaningful to me personally. Thank you.

Next in line is Professor Don Estep. I want to thank him for providing the motivating example for this work and for his lectures in adjoint methods. Thanks also for the support from NSF grant MSPA-CSE-0434354, "Novel A Posteriori Analysis of Ecological Models: The Carbon Cycle".

I would also like to thank the faculty members and my colleagues of this department at CSU. In particular, I want to thank Professor Hari Iyer for his encouragement in my education, excellent teaching and for serving as a member of my dissertation committee. I also want to thank my research partner, Wenying, for her friendship and good company during the past three years.

Thanks to Mom and Dad for their sacrifices and allowing me to postpone growing up for a few more years. Thanks to my sister for being my best friend throughout my life. Today is not possible without their endless support and love.

Finally, I want to express my deepest gratitude to my husband Bo for his complete and unconditional love. There are no words to describe my appreciation for all the support he has given to me and always being there for me. Thanks, Bo, for all the happiness and excitement that you have brought to my life, for all the hopes and dreams in the years to come.

DEDICATION

Dedicated to my parents Qinling and Yuzhi, sister Yu and husband Bo

my foundation of happiness and strength

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

This dissertation proposes a new approach for modeling the code output of computer experiments. The use of computer experiment has been an emerging alternative for studying many complicated physical phenomena, which are usually described by a mathematical model. There often is no analytical solution to the quantity of interest in the mathematical model. Fortunately, numerical solutions can be obtained by implementing the mathematical model through computer simulations. For any given input value $\boldsymbol{x} \in \mathbb{R}^d$, running the computer simulation gives one or more outputs, defining a mapping from the input space to output responses. This is called a *computer experiment.* Computer experiments have been widely applied in many scientific fields and engineering. For example, many human diseases are epidemics and have the potential to affect large segments of a population. An epidemic is a complicated matter, but the danger posed by new and uncontrollable diseases compels us to learn as much as we can about the nature of epidemics. Mathematics offers a way to help the understanding of the spread of disease. Susceptible-Infected-Resistant (SIR) model is a class of epidemiological models that describes the dynamics of disease spread through a system of ordinary differential equations relating at time t, the number of susceptible people S(t), the number of infected people I(t), and the number of resistant people R(t). The results from such an experiment would help prediction of an *outbreak*, i.e., when the infected population hits its peak.

In a typical computer experiment, the input is usually high-dimensional and the output y is deterministic, i.e., running the code with the same input $\boldsymbol{x}$ would give the same output. Most computer codes are expensive to execute though it is easy to control and cheaper compared with physical experiments. The limited expense of running complex computer code leads to some interesting problems in the study of computer experiments. The SIR model in Estep and Neckels (2006) can be used as an illustrative example. The model is seven-dimensional and over the time interval $[0, T]$. There are three responses of interest: the average number of susceptible individuals $q(\boldsymbol{x})_1 = \frac{1}{T}\int_0^T S(s, \boldsymbol{x})ds$, the average number of infected individuals $q(\boldsymbol{x})_2 = \frac{1}{T}\int_0^T I(s, \boldsymbol{x})ds$, and the average number of resistant individuals $q(\boldsymbol{x})_3 = \frac{1}{T}\int_0^T R(s, \boldsymbol{x})ds$. These responses are linear functionals of the solutions from the model. The method that Estep and Neckels (2006) use provides not only the response but also the first partial derivatives of the response. Supposing this model is expensive to execute, the natural questions that may arise are:

1. Which data points in the high-dimensional input space should be selected to run the computer code? The data set cannot be large due to the expense of running the code, but the input space needs to be fully explored.

2. How to predict the responses at untried locations? That is, how to model the relationship between response and input based on observed computer runs?

3. Which input factors are more important to explain the variation in the response?

4. The derivatives of the response may provide additional useful information about the shape of the response surface. How can we bring the derivative information into modeling and prediction?

5. If the inputs have known distribution, how does the variation in the inputs impact the variation in the outputs?

The first question is the problem of design in computer experiments. A lot of study has been done in this area. The Latin hypercube design and maximin distance design are used in this study. A brief introduction of this is given in Section 1.1. The second and fourth questions relate to the modeling problem in computer experiments. There are several classes of models that can be used to model the computer code outputs. The objective of this work is to propose a new model, called stochastic heteroskedastic process (SHP) model, that has more flexibility in capturing the salient features of computer code outputs and does a better job in quantifying uncertainty in the selection of new data points.

1.1 Design of Computer Experiments

Traditional statistical design of experiments (DOE) are based on observations from physical experiments in which the random error exists due to nuisance factors. Since most computer experiment outputs are deterministic, the statistical approaches for design in physical experiments such as replicates, blocking, and randomization cannot be directly applied to the output from the computer experiment (Santer et al. (2003)). Among the classical DOE methods in physical experiments, the central composite design and full- and fractional-factorial design are the most widely used in practice. In traditional DOE, people assume some knowledge of the trend of a true response surface and the objective is to minimize the random error. This goal is followed in traditional DOE methods by placing several sample points on the boundary of the input space and a few sample points in the interior of the input space (Myers and Montgomery (1995)). In the design of computer experiments, the true response trend is unknown. This requires one to place the sample points on the interior of the input space. Despite the difference of observations from physical and computer experiments, the experimental designs share the common goal of extracting as much as information as possible from the experiments. Some approaches in design of computer experiments are briefly summarized in the following subsections.

1.1.1 Latin hypercube design

Since Mckay et al. (1979) originally developed the Latin hypercube sampling (LHS) method, LHS has been one of the popular methods in the design of computer experiments. Under certain assumptions, the LHS provides a more accurate estimate for the mean than simple random sampling. Another attractive aspect of LHS is that there is no restriction on the sample size with increasing dimensionality in the input space. The algorithm that generates Latin hypercube samples is described in detail by Santer et al. (2003). Suppose the input space has been standardized into $[0,1]^d$. To obtain a LHS of size n, divide each axis [0,1] into n equally-spaced intervals. Fill the cells with integers $1, ..., n$ in such a way that each integer appears exactly once in each row and column. The algorithm for generating LHS with n points is:

$$x_{ik} = \frac{\Pi_{ik} - U_{ik}}{n}, \quad i = 1, ..., n; \quad k = 1, ...d,$$

where x_{ik} is the k^{th} component of $\boldsymbol{x}_i$, $\Pi = (\Pi_{ik})$ is an $n \times d$ matrix containing in each column an independent random permutations of the sequence of integers $1, ..., n$ and U is the uniform random value on $[0,1]$ (Santer et al. (2003)).

One property of LHS is that the sampled points are spread evenly over the domain of each input variable. But the sample from LHS is not guaranteed to spread evenly over the whole input space. To overcome this limitation, some variations of LHS design have been studied, such as orthogonal array (OA) sampling. An orthogonal array produces a sample that yields uniform sampling in any $t-$dimensional projection of a $d-$dimensional input space where $t < n$. But one restriction on OA sampling is that there are only certain values of n, t for which the orthogonal arrays exists (Wu and Hamada (2000)).

1.1.2 Maximin and minimax designs

A design can be selected based on a distance measure that quantifies how spread out a set of points are. Let $\mathcal{D} \subset \chi \subset \mathbb{R}^d$ be a design of size n and ρ be a metric on

χ. A design that maximizes the smallest distance between any two points in $\mathcal{D}$ is a *maximin distance design* and denoted by $\mathcal{D}_{Mm}$ (Santer et al. (2003)):

$$\min_{\boldsymbol{x}_1,\boldsymbol{x}_2\in\mathcal{D}_{Mm}} \rho(\boldsymbol{x}_1,\boldsymbol{x}_2) = \max_{\mathcal{D}\subset\chi} \min_{\boldsymbol{x}_1,\boldsymbol{x}_2\in\mathcal{D}} \rho(\boldsymbol{x}_1,\boldsymbol{x}_2).$$

A maximin strategy for sampling would ensure that no two points are too close to each other so that the sampled points spread out over the space.

A design that minimizes the maximum distance between arbitrary points $\boldsymbol{x} \in \chi$ and the candidate design $\mathcal{D}_{mM}$ over all designs $\mathcal{D} \subset \chi$ is called a *minimax distance design* (Santer et al. (2003)):

$$\max_{\boldsymbol{x}\in\chi} \rho(\boldsymbol{x},\mathcal{D}_{mM}) = \operatorname*{minmax}_{\mathcal{D}\subset\chi\ \boldsymbol{x}\in\chi} \rho(\boldsymbol{x},\mathcal{D}).$$

A minimax strategy would ensure that every point $\boldsymbol{x} \in \chi$ is not too far from some point in $\mathcal{D}$. The minimax design will generally lie in the interior of the design space and the maximin design will be more likely to place the sample points on the boundary to make points further away from each other. Even though the minimax distance design provides better coverage on the input space, the maximin distance design is more popular in design of computer experiments because it is easier to implement than minimax design. For more details about the maximin and minimax design, refer to Santer et al. (2003).

Each of the design methods produces certain attractive properties, but none of them is completely satisfactory. Combining two or more methods together can generate an approach that has several attractive features simultaneously. One popular way is to restrict the set of candidate designs to LHS and then using a maximin distance-based criterion to select a design from this restricted class (Gramacy (2005)). This combined strategy is used in the several applications of this study.

1.1.3 Adaptive sampling designs

Adaptive sampling, also called sequential design of experiments or active data selection in the world of machine learning, aims at finding a set of sample points that yield a desired accuracy with minimal computational cost. Active learning studies how to query data points based on previously obtained training data so as to incorporate as much new information into the model as possible. Active learning is important in the situation when data are expensive or difficult to obtain. In computer experiments, each training point may take days to compute and cost thousands of dollars. Optimally sampled points save time and cost.

There are different goals in computer experiments using active learning. One primary interest of computer experiments is optimization. Sequential experimental designs have been proposed for locating those interesting $\boldsymbol{x}$ that optimize the output or a function of the output of a computer code (Santer et al. (2003)). The idea of these methods is to obtain some information about the whole surface through an initial set of sampled points, while additional points may be sampled in some subregions until some criterion is met. In the traditional DOE, many researchers use the following criteria (Fang et al. (2006)) in adaptive sampling:

- *D-optimality*: maximize the determinant of $\mathbf{M}$
- *A-optimality*: minimize the trace of $\mathbf{M}^{-1}$
- *E-optimality*: minimize the largest eigenvalue of $\mathbf{M}^{-1}$

where $\mathbf{M}$ is the information matrix for the chosen model. When the model is true, these optimal designs are the best under the criterion, but they lack robustness to model misspecification.

In computer experiments, entropy and mean square prediction error are two most-often studied criteria. Mean square error designs aim at minimizing the expected mean square error; more details are discussed in Sacks et al. (1992). The

integrated mean squared prediction error (IMSPE) and maximum mean square prediction error (MMSPE) can be used as optimality criteria. In a Gaussian process model, a *maximum entropy design* is equivalent to maximizing the determinant of the covariance matrix of $\boldsymbol{y}$. These criterion-based designs require the knowledge of the correlation function of $\boldsymbol{y}$ (for more technical details, see Santer et al. (2003) and Koehler and Owen (1996)). Due to the lack of easily accessible of software to generate these designs, LHS design is still a more popular choice.

In contrast to the goal of optimization, the goal of this study on adaptive sampling is to efficiently and accurately predict y for a given $\boldsymbol{x}_0$. Instead of using the above sequential design criteria, we consider two useful algorithms for active data selection in machine learning. The first one is developed by Mackay (1992), called ALM for Active Learning Mackay, and has been aimed to maximize the expected information gain about the model parameter values ψ when we receive the new data $\tilde{\boldsymbol{x}}$. Shannon's entropy is used as the information measure. Mackay proved that we will learn most about the model by selecting the data $\tilde{\boldsymbol{x}}$ with largest predictive variance in the input space. One non-ideal property of this criterion is that the error bars are largest at the most extreme points where data have been gathered for most models. This leads us to repeatedly sample data at the edges of the input space (Mackay (1992)). But this disadvantage can be reduced by querying data from a set of candidates $\tilde{\boldsymbol{X}}$ in a defined region of interest or spread out over the space.

The second approach suggested by Cohn (1996), called ALC for Active Learning Cohn, is to select the data $\tilde{\boldsymbol{x}}$ to minimize the expectation of the mean square error over input space χ. More technical details are described in Chapter 4, on adaptive sampling.

1.2 Stationary Gaussian Process Model

In a typical computer experiment, a high-dimensional vector $\boldsymbol{x} \in \mathbb{R}^d$ is used as input to a computer code, yielding an output $y(\boldsymbol{x})$. Because the code is expensive

to execute, one of the major goals of computer experiments is to seek an approximation model (metamodel) which is close to the true code but faster to run. A statistical approach to the problem is to model the response $y(\boldsymbol{x})$ as a realization from a stochastic process and to construct a predictor appropriate for that process. For example, a stationary Gaussian process (GP) leads to kriging, or empirical best linear unbiased prediction (BLUP), which is a popular technique in computer experiments. A closely-related approach is Bayesian prediction of the deterministic function under a GP model, which has also been studied extensively during the past twenty years.

A GP, $Z(\boldsymbol{x})$, is a collection of random variables indexed by $\boldsymbol{x}$, on some probability space $(\Omega, \mathcal{F}, \mathcal{P})$, $\boldsymbol{x} \in \chi \subset \mathbb{R}^d$. The finite dimensional distributions of a GP are multivariate normal for every n and every collection $\{Z(\boldsymbol{x}_1), Z(\boldsymbol{x}_2), ..., Z(\boldsymbol{x}_n)\}$. A GP is completely specified by its mean and covariance functions. Let $\boldsymbol{x}$ and $\boldsymbol{x}'$ be two inputs in the space and denote the mean function as $\mu(\boldsymbol{x})$ and the covariance function as $C(\boldsymbol{x}, \boldsymbol{x}')$. Then

$$\begin{aligned} \mu(\boldsymbol{x}) &= E[Z(\boldsymbol{x})], \\ C(\boldsymbol{x}, \boldsymbol{x}') &= E[(Z(\boldsymbol{x}) - \mu(\boldsymbol{x}))(Z(\boldsymbol{x}') - \mu(\boldsymbol{x}'))], \end{aligned}$$

and the corresponding correlation function is:

$$R(\boldsymbol{x}, \boldsymbol{x}') = \frac{C(\boldsymbol{x}, \boldsymbol{x}')}{\sigma(\boldsymbol{x})\sigma(\boldsymbol{x}')},$$

where $\sigma(\boldsymbol{x}) = C(\boldsymbol{x}, \boldsymbol{x})$. Thus, a GP can be written as

$$Z(\boldsymbol{x}) \sim GP(\mu(\boldsymbol{x}), C(\boldsymbol{x}, \boldsymbol{x}'))$$

A stationary GP in the wide sense (weak stationarity) is defined as (Banerjee et al. (2003)):

- $\mu(\boldsymbol{x}) = \mu$

- $E|Z(\boldsymbol{x})|^2 < \infty$
- $E[(Z(\boldsymbol{x}) - \mu(\boldsymbol{x}))(Z(\boldsymbol{x}') - \mu(\boldsymbol{x}'))] = C(\boldsymbol{x} - \boldsymbol{x}') = C(\boldsymbol{h})$, where $\boldsymbol{h} = \boldsymbol{x} - \boldsymbol{x}'$.

If, in addition, $E[(Z(\boldsymbol{x}) - \mu(\boldsymbol{x}))(Z(\boldsymbol{x}') - \mu(\boldsymbol{x}'))] = C(||\boldsymbol{x} - \boldsymbol{x}'||) = C(h)$, then the covariance function is solely a function of the distance between two locations and the GP is *isotropic*. The isotropic GP has been widely used in spatial statistics (Banerjee et al. (2003)).

Sacks et al. (1989) brought GP in computer experiments. The classic GP model has the form

$$y(\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{x})^T\boldsymbol{\beta} + \sigma Z(\boldsymbol{x}), \qquad \sigma > 0, \tag{1.1}$$

where $\boldsymbol{g}(\boldsymbol{x}) = [g_1(\boldsymbol{x}), ..., g_p(\boldsymbol{x})]^T$ are known regression functions, $\boldsymbol{\beta} = (\beta_1, ...\beta_p)^T$ is a vector of unknown regression coefficients, and $\sigma Z(\boldsymbol{x})$ is a stationary GP with mean 0, variance σ^2 and correlation function R. The term $\boldsymbol{g}(\boldsymbol{x})^T\boldsymbol{\beta}$ captures the large-scale variation in the process, while $Z(\boldsymbol{x})$ captures the small-scale variation.

1.2.1 Covariance functions

A process can possess different levels of smoothness. In a GP, σ^2 and the correlation function R characterize the distributional properties of the spatially correlated error process $Z(\boldsymbol{x})$, so the choice of a GP can be reduced to that of a covariance or correlation function that has desired differentiability and smoothness characteristics. In this section, we give two examples of correlation functions: the power family and Matérn family.

Power family

A popular family of correlation functions is the *power family*. The isotropic correlation function in the power family can be written as

$$R(\boldsymbol{x}, \boldsymbol{x}') = \exp\{-\phi||\boldsymbol{x} - \boldsymbol{x}'||^p\}, \tag{1.2}$$

where ϕ is referred to as the range parameter, and the correlation increases as ϕ decreases. The power $0 < p \leq 2$ determines the smoothness of the underlying process, which can either be chosen in advance or estimated. Every process for $0 < p < 2$ is continuous at the origin but not differentiable at the origin. When $p = 1$, the power correlation function is actually an exponential correlation function. When $p = 2$, the correlation function is called the Gaussian correlation function, which is infinitely times differentiable at the origin. In computer experiments, $p = 2$ is a popular choice since the response function is assumed to be smooth.

An extension of the isotropic power family is to allow different values of the range parameter ϕ_k in each dimension ($k = 1, ..., d$). The resulting d-dimensional separable version of the power correlation has the form

$$R(\boldsymbol{x}, \boldsymbol{x}') = \exp\left\{-\sum_{k=1}^{d} \phi_k |x_k - x_k'|^p\right\}, \tag{1.3}$$

which is a legitimate correlation function since it is a production of correlation functions. With the separable power family, one can allow for different strengths of correlation in each dimension. But extra parameters have been introduced into the model, which will reduce the efficiency of the implementation if the true underlying correlation structure is isotropic.

Figures 1.1 and 1.2 show the effects of changing the power parameter p and range parameter ϕ on the sample path from a GP with power correlation function. As we can see from the top two panels in Figure 1.1, the sample paths are not differentiable for $p < 2$, and the sample paths in the bottom panel are infinitely differentiable for $p = 2.0$. As the range parameter ϕ decreases, the sample path becomes more smoother since the correlation between two fixed locations increases. As ϕ decreases to 0, the correlation becomes 1 and the sample paths becomes more constant.

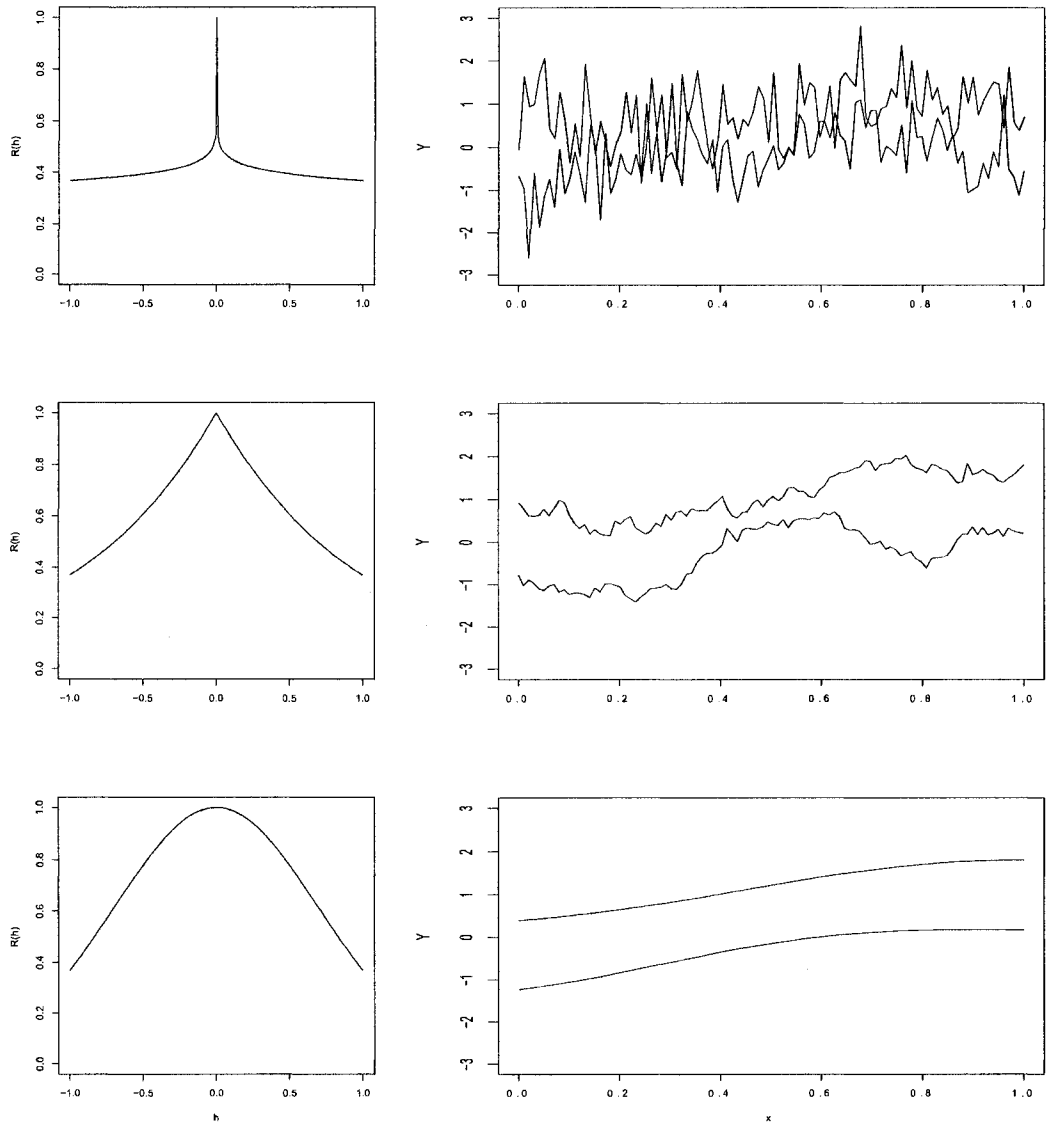


Figure 1.1: Effect of varying the power parameter p on a GP with power correlation function. Correlation function (left panels) and two realized sample paths (right panels) for a zero mean, unit variance GP with power correlation having fixed $\phi = 1$ with $p = 0.1$ (top plots), $p = 1$ (middle plots), and $p = 2$ (bottom plots).

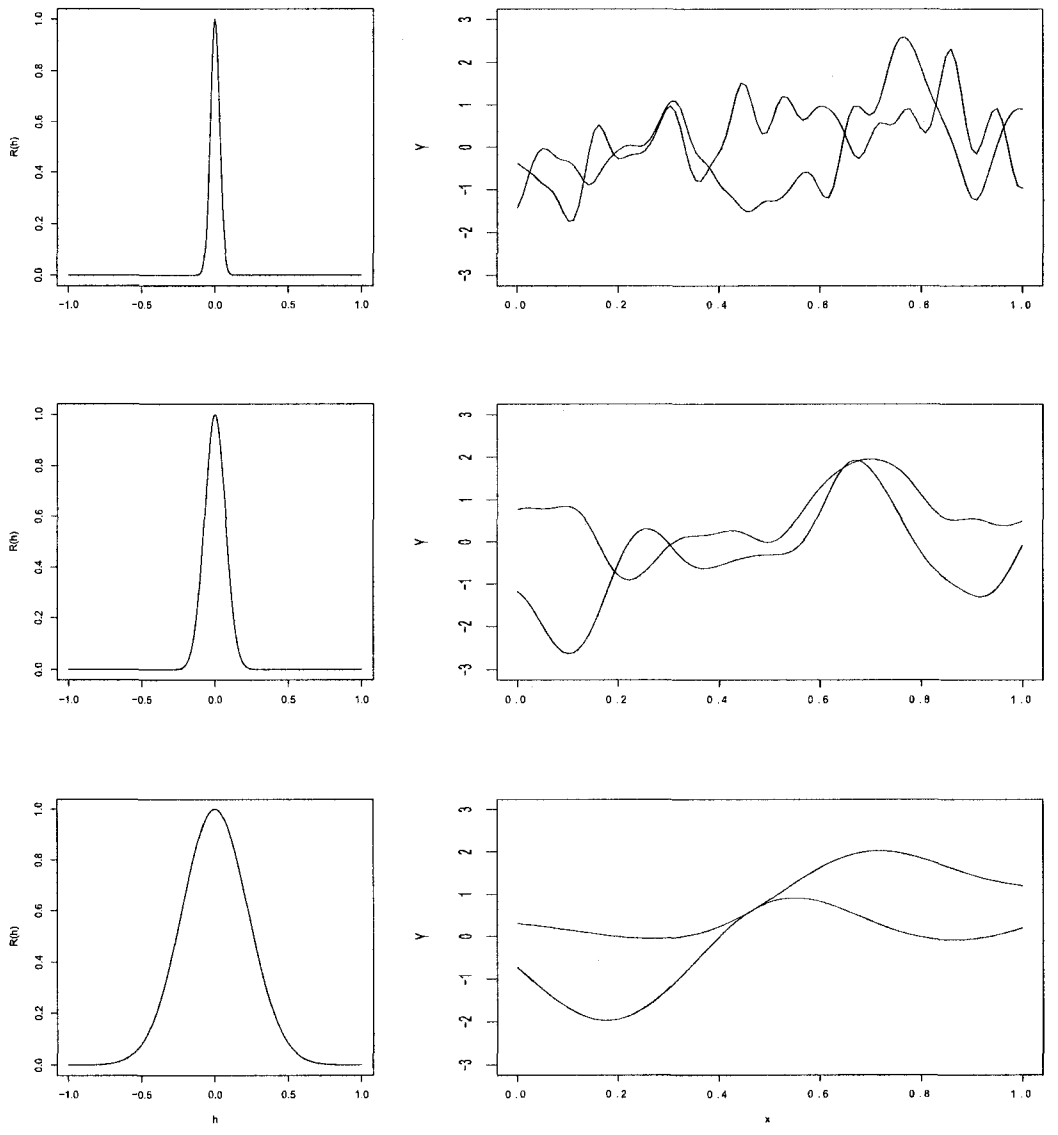


Figure 1.2: Effect of varying the range parameter ϕ on a GP with a power correlation function. Correlation function (left panels) and two realized sample paths (right panels) for a zero mean, unit variance GP with power correlation having fixed $p = 2$ with $\phi = 500$ (top plots), $\phi = 100$ (middle plots), and $\phi = 10$ (bottom plots).

Matérn family

The isotropic Matérn correlation function first introduced by Matérn (1960) has the form

$$\rho(h) = \frac{1}{\Gamma(\nu)2^{\nu-1}}(2\sqrt{\nu}\phi\|h\|)^{\nu}\mathcal{K}_{\nu}(2\sqrt{\nu}\phi(h), \tag{1.4}$$

where $\mathcal{K}_{\nu}(\cdot)$ is the modified Bessel function of order ν. There are two parameters in this correlation function. The range parameter ϕ controls the correlation decay rate and the smoothness parameter ν controls the smoothness of the sample path. As ν increases, the smoothness of the sample paths from a GP increases. As $\nu \to \infty$, the Matérn approaches the Gaussian correlation function with form (Santer et al. (2003))

$$R(h) = \exp(-\phi^2 h^2).$$

When $\nu = 0.5$, the Matérn correlation function is equivalent to an exponential correlation function with parameterization

$$R(h) = \exp(-\sqrt{2}\phi h).$$

The top panel in Figure 1.3 shows the marginal effect of ν on the smoothness of the Matérn correlation function. The order of differentiability of the Matérn correlation at the origin increases as ν increases. It is actually hard to tell the difference between the Matérn correlation and a Gaussian correlation when $\nu > 10$. When $\nu = 0.5$, we see the overlap of the exponential correlation with the Matérn correlation function. The sample paths become wiggly as ν decreases, as shown in the bottom panel in Figure 1.3.

1.2.2 Estimation and prediction

Given observed data $(\boldsymbol{x}_1, y(\boldsymbol{x}_1)), \ldots, (\boldsymbol{x}_n, y(\boldsymbol{x}_n))$, we want to predict $y(\cdot)$ at an untried input $\boldsymbol{x}_0$ based on the stationary GP model equation in (1.1). One approach

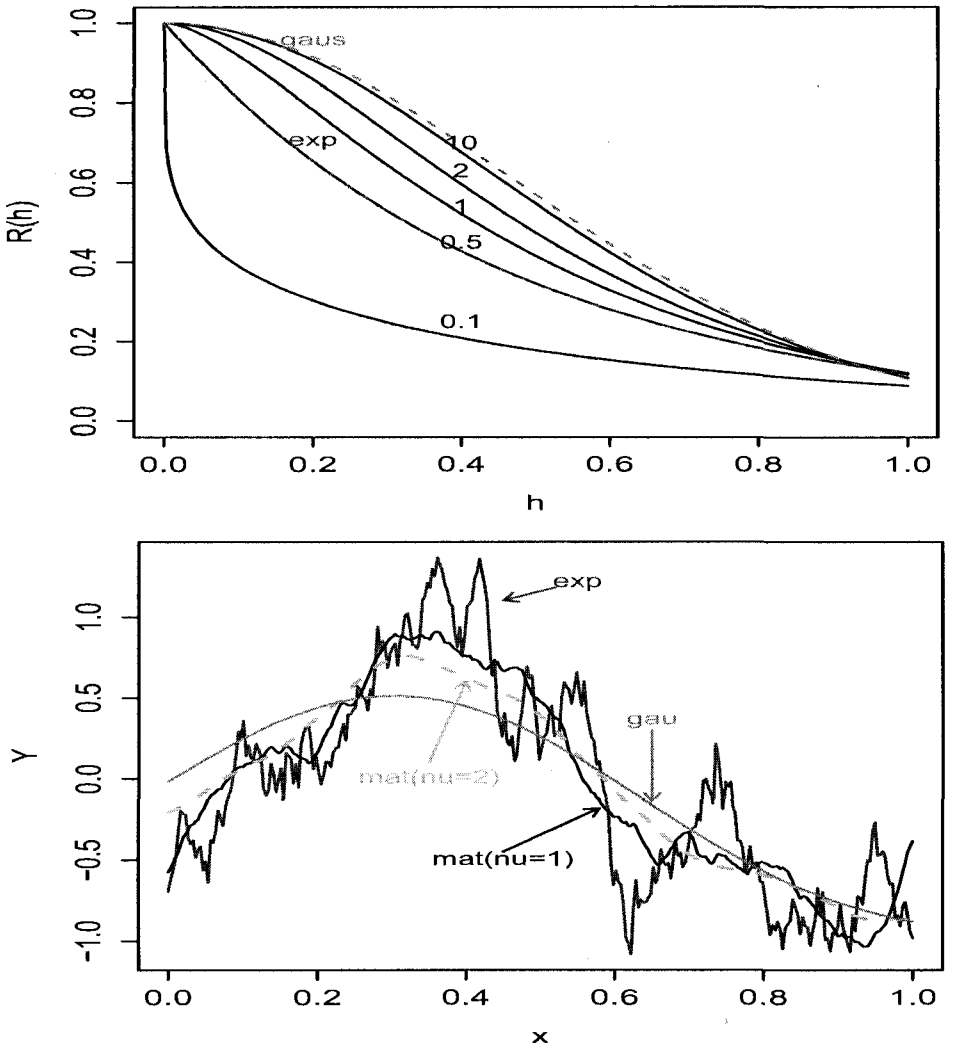


Figure 1.3: Effect of varying the smoothness parameter ν on the correlation functions and sample paths of a GP with Matérn correlation function. Top panel: Matérn correlation function with fixed $\phi = 1.5$ and $\nu = 0.1, 0.5, 1, 2, 10$. The dashed line is a Gaussian correlation function with $\phi = 1.5^2$. The dotted line is an exponential correlation with $\phi = \sqrt{2} \times 1.5$. Bottom panel: Realizations from a zero mean, unit variance GP with Matérn correlation having fixed $\phi = 1.5$ and $\nu = 0.5$ (exp), 1, 2, ∞(gau).

is similar to kriging (Matheron (1963)), which essentially is the empirical best linear unbiased estimator (BLUP) for $y(\boldsymbol{x}_0)$ given by (Sacks et al. (1989))

$$\hat{y}(\boldsymbol{x}_0) = \boldsymbol{g}(\boldsymbol{x}_0)^T \boldsymbol{\beta} + \boldsymbol{r}(\boldsymbol{x}_0, \boldsymbol{x}) R^{-1} (\boldsymbol{y} - G^T \boldsymbol{\beta}), \tag{1.5}$$

and mean square prediction error for the BLUP is

$$\mathrm{MSE}(\hat{y}(\boldsymbol{x}_0)) = \sigma^2 \left\{1 - r(\boldsymbol{x}_0, \boldsymbol{x})R^{-1}r(\boldsymbol{x}, \boldsymbol{x}_0) + \boldsymbol{h}^T(G^T R^{-1} G)\boldsymbol{h}\right\}, \qquad (1.6)$$

where $r = [R(\boldsymbol{x}_0, \boldsymbol{x}_1), ..., R(\boldsymbol{x}_0, \boldsymbol{x}_n)]^T$ is a $n \times 1$ correlation vector between $z(\boldsymbol{x}_0)$ and $z(\boldsymbol{x}_1), ..., z(\boldsymbol{x}_n)$, $G = [\boldsymbol{g}(\boldsymbol{x}_1), ..., \boldsymbol{g}(\boldsymbol{x}_n)]^T$ is the $n \times p$ matrix of regressors for $\boldsymbol{y}$, R is the $n \times n$ correlation matrix between z values at observed locations, and $\boldsymbol{h} = \boldsymbol{g}(\boldsymbol{x}_0) - G^T R^{-1}\boldsymbol{r}$.

Alternatively, from a Bayesian point of view, the mean and variance of the posterior distribution $p(y(\boldsymbol{x}_0)|y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n))$ would be the best predictor and predictive variance for $y(\boldsymbol{x}_0)$. Assuming that all parameters are known, the posterior mean and variance are given by

$$E[y(\boldsymbol{x}_0)|y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n)] = \boldsymbol{g}(\boldsymbol{x}_0)^T\boldsymbol{\beta} + \boldsymbol{r}_z(\boldsymbol{x}_0, \boldsymbol{x})R_z^{-1}(\boldsymbol{y} - G^T\boldsymbol{\beta}), \qquad (1.7)$$

$$\mathrm{Var}[(y(\boldsymbol{x}_0)|y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n)] = \sigma^2(1 - r_z(\boldsymbol{x}_0, \boldsymbol{x})R_z^{-1}r_z(\boldsymbol{x}, \boldsymbol{x}_0)). \qquad (1.8)$$

Further, if all the model parameters are known, the predictors (1.5) and (1.7) are identical for the GP model, but with different predictive variances. In practice, the model parameters $\boldsymbol{\beta}, \sigma^2, \boldsymbol{\phi}$ are unknown and need to be estimated. Since $\boldsymbol{y}$ is normally distributed, the maximum likelihood estimator (MLE) of $\boldsymbol{\beta}$ given $\boldsymbol{\phi}$ is the generalized least squares estimator

$$\hat{\boldsymbol{\beta}} = (G^T R^{-1} G)^{-1} G^T R^{-1}\boldsymbol{y},$$

and the MLE for σ^2 is

$$\hat{\sigma^2} = \frac{(\boldsymbol{y} - G\hat{\boldsymbol{\beta}})^T R^{-1}(\boldsymbol{y} - G\hat{\boldsymbol{\beta}})}{n}.$$

The MLE $\hat{\boldsymbol{\phi}}$ of $\boldsymbol{\phi}$ maximizes

$$-\frac{1}{2}(n\log(\hat{\sigma^2}) + \log|R|.$$

The values of $\boldsymbol{r}$ and R with $\boldsymbol{\phi}$ replaced by $\hat{\boldsymbol{\phi}}$ will be denoted by $\hat{\boldsymbol{r}}$ and $\hat{R}$.

1.3 Applications of GP Model in Computer Experiments

1.3.1 Multi-level/Multi-resolution responses

The GP model (1.1) is not only a popular metamodeling approach for deterministic computer code outputs, it can also model the bias between the multi-level or multi-resolution responses. The computer code can be run at different levels of complexity. A simple version of the code is a fast approximation to reality, which includes the most important features of the response but might be biased. A complex version of the code is a better approximation to reality, but is limited by expense of computation. There is a trade-off between the complexity of the expensive code and the availability of simpler approximations. Kennedy and O'Hagan (2000) introduced a Baysian analysis of multilevel code using an autoregressive model to integrate the outputs from different levels. Each level t of the code can be modeled using a GP:

$$\begin{aligned} Z_t(\boldsymbol{x}) &= \rho_{t-1} Z_{t-1}(\boldsymbol{x}) + \delta_t(\boldsymbol{x}), \quad t = 2, \ldots, s \\ \delta_t(\cdot) &\sim GP(h(\cdot)^T \beta_t, \sigma_t^2 \rho_\delta(\cdot, \cdot)), \\ Z_1(\cdot) &\sim GP(\beta_1, \sigma_1^2 \rho_z(\cdot, \cdot)). \end{aligned}$$

The problem of model validation in computer experiments can be considered as a two-level response: cheaper computer code outputs and expensive, real physically observed data. The computer model needs adjustment to fit the observed data. Kennedy and O'Hagan (2001) presents a Bayesian approach to the validation of computer models, starting from the model

$$\begin{aligned} y^o(\boldsymbol{x}) &= y^T(\boldsymbol{x}) + e(\boldsymbol{x}), \\ y^T(\boldsymbol{x}) &= y^c(\boldsymbol{x}) + b(\boldsymbol{x}), \end{aligned}$$

where $y^T(\boldsymbol{x})$ denotes the true process, $y^c(\boldsymbol{x})$ denotes the computer code output, $b(\boldsymbol{x})$ denotes the model inadequacy and $e(\boldsymbol{x})$ is measurement error. Gaussian processes are used to model $y^c(\boldsymbol{x})$ and $b(\boldsymbol{x})$.

1.3.2 Derivative of GP

Some computer experiments provide both $y(\cdot)$ and its first partial derivatives at observed inputs $\boldsymbol{x}$. The derivative of a GP is still a GP and can be combined into the modeling paradigm. Morris et al. (1993) proposed a Bayesian modeling procedure that can simultaneously model the response and its derivatives. Let $y^{(j)}(\boldsymbol{x}) = \partial y(\boldsymbol{x})/\partial x_j$ denote the first partial derivative of $y(\boldsymbol{x})$ with respect to the j^{th} component of $\boldsymbol{x}$ for $1 \leq j \leq d$. If $C(\boldsymbol{x}_1, \boldsymbol{x}_2)$ denotes the covariance function for $y(\cdot)$ at two points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, the covariance function for the partial derivative process is given by

$$\mathrm{Cov}\{y^{(i)}(\boldsymbol{x}_1), y^{(j)}(\boldsymbol{x}_2)\} = \frac{\partial^2 C(\boldsymbol{x}_1, \boldsymbol{x}_2)}{\partial x_{1i} \partial x_{2j}},$$

and the covariance function between response and derivative is

$$\mathrm{Cov}\{y(\boldsymbol{x}_1), y^{(j)}(\boldsymbol{x}_2)\} = \frac{\partial C(\boldsymbol{x}_1, \boldsymbol{x}_2)}{\partial x_{2j}}.$$

The derivative information can be combined into modeling and prediction since the joint distribution of $(y_0, \boldsymbol{y}, \boldsymbol{y}')$ is

$$\begin{bmatrix} y_0 \\ \boldsymbol{y} \\ \boldsymbol{y}' \end{bmatrix} \sim \mathrm{N}\left(\begin{bmatrix} \boldsymbol{g}(\boldsymbol{x}_0)^T \\ \boldsymbol{G}(\boldsymbol{x})^T \\ \boldsymbol{G}'(\boldsymbol{x})^T \end{bmatrix} \boldsymbol{\beta}, \sigma^2 \begin{bmatrix} 1 & \boldsymbol{r}_y(\boldsymbol{x}_0, \boldsymbol{x}) & \boldsymbol{r}_{yy'}(\boldsymbol{x}_0, \boldsymbol{x}) \\ \boldsymbol{r}_y(\boldsymbol{x}, \boldsymbol{x}_0) & R_y & \boldsymbol{r}_{yy'}(\boldsymbol{x}, \boldsymbol{x}) \\ \boldsymbol{r}_{y'y}(\boldsymbol{x}, \boldsymbol{x}_0) & \boldsymbol{r}_{y'y}(\boldsymbol{x}, \boldsymbol{x}) & R_{y'} \end{bmatrix} \right). \quad (1.9)$$

The empirical BLUP and BP for $y(\boldsymbol{x}_0)$ based on $\boldsymbol{y}$ and $\boldsymbol{y}'$ easily follow.

1.4 Other Approaches for Metamodeling

The stationary GP model is a popular metamodeling approach in computer experiments because it is conceptually straightforward to implement and can produce prediction intervals. A stationary covariance function of a GP assumes uniform smoothness of the response surface over the input space. The stationarity assumption is a severe restriction for functions whose smoothness varies considerably over

the input space. In computer experiments, outputs often possess heterogeneous features. For example, subsonic flow is quite different from supersonic flow in the application of computational fluid dynamics (Gramacy et al. (2004)). Using the stationary GP model will oversmooth in some regions and will undersmooth in others.

To overcome this limitation of a stationary GP, both regression and nonstationary GP techniques have been developed. Multivariate adaptive regression splines (MARS) have been used in the metamodeling of computer experiments by Jin et al. (2000) and Simpson et al. (2001a). The number of knots and locations for the splines are adaptively determined from the data to account for inhomogeneity. Artificial neural networks (ANN) are another approach for flexible modeling of the output from computer experiments (Chen and Varadarajan (1997) and Simpson et al. (2001b)). The MARS and ANN approaches have implicit covariance functions, and both have large numbers of coefficients with no clear interpretation. Chen et al. (2003) gives a good review about metamodeling in computer experiments.

Gaussian processes with nonstationary covariance functions have been widely studied in the fields of statistics and geostatistics (Higdon et al. (1999), Fuentes and Smith (2001), Gelfand et al. (2003) and Yan (2007)). Most of these nonstationary models work well in low-dimensional (e.g. $\mathbb{R}^2$ or $\mathbb{R}^3$) physical experiments. Xiong et al. (2007) incorporate a nonstationary covariance function in modeling high-dimensional computer experiments. The nonstationary covariance function is formulated through a non-linear map with sparse parameterization, but the total number of model parameters may still be large in complicated cases. Gramacy et al. (2004) developed a tree-based Gaussian method to model nonstationarity in a response surface, by fitting individual GP models within subregions. The computational cost is reduced by this method, but discontinuities across subregions cannot be avoided.

There is currently no universally accepted approach to metamodeling in computer experiments. It is therefore desirable to develop more sophisticated models

that are able to model very complex functions with respect to a large number of variables and are attractively interpretable.

1.5 Motivation for Stochastic Heteroskedastic Process Model

In order to capture the inhomogeneous features in computer experiment data, we adapt the idea of heteroskedasticity modeling from time series. The two classes of heteroskedastic models in time series are autoregressive conditional heteroskedasticity (ARCH) and stochastic volatility (SV); see Shephard (1996) for a review. In an ARCH model, the conditional variance is modeled as a linear function of the squares of the past information. In an SV model, the conditional variance is modeled as a latent stochastic process. The most popular stochastic volatility model from Taylor (1986) assumes an AR(1) model for the latent process. This model is given by

$$\begin{aligned} y_t &= \epsilon_t \exp(h_t/2), \qquad \{\epsilon_t\} \text{ iid } \mathrm{N}(0,1), \\ h_t &= \gamma_0 + \gamma_1 h_{t-1} + \eta_t, \quad \{\eta_t\} \text{ iid } \mathrm{N}(0,\sigma_\eta^2), \end{aligned}$$

where iid denotes independent and identically distributed. For $|\gamma_1| < 1$, this is a strictly stationary model. The bottom plot in Figure 1.5 is a sample path from an SV model, which looks like a realization from a nonstationary model. The apparent inhomogeneity in the data comes from the conditional variance, shown in the middle plot. The top panel is the plot for the corresponding latent process h_t. The conditional variance of y_t given h_t is e^{h_t}, so that the correlated latent process h_t captures "volatility clustering effects"; that is, positively autocorrelated conditional variance. In his study of heteroskedasticity for spatial lattice data, Yan (2007) adapts the SV idea by introducing a spatial stochastic volatility (SSV) component into the widely-used conditional autoregressive (CAR) model.

We extend the SV idea to the spatial context, where time t is replaced by a continuous (non-lattice) multi-dimensional index $\boldsymbol{x}$, and where the uncorrelated noise

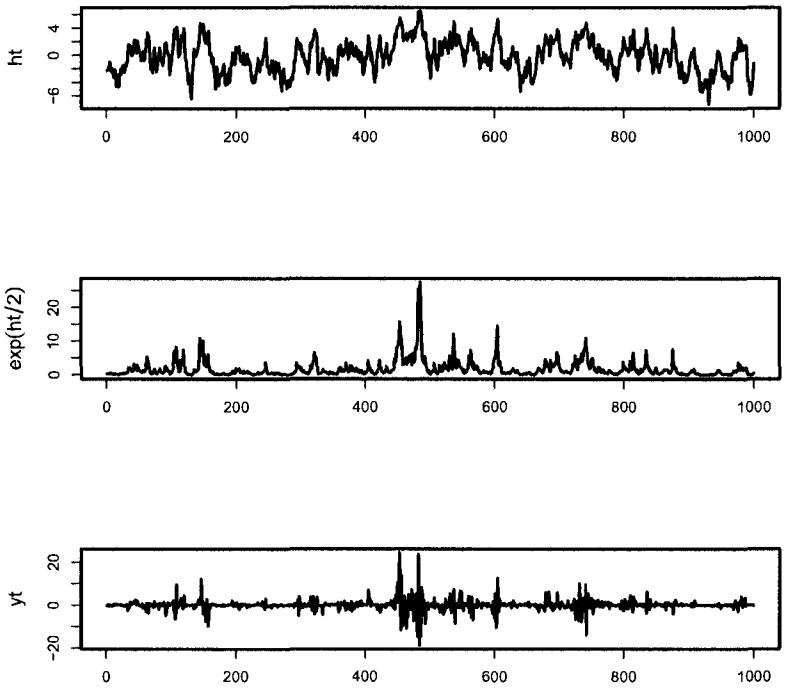


Figure 1.4: Laten process $\{h_t\}$, conditional variance e^{h_t} and SV sample path y_t.

$\{\epsilon_t\}$ is replaced by a random field Z. In a standard GP model for CE data, the random field Z is used to capture the small-scale variation in the deterministic computer experiment outputs. By introducing a spatial stochastic volatility component into the GP model, we propose a new spatial model that allows for more flexibility in capturing the salient features of computer outputs. We model the deterministic computer response as a realization from a stochastic heteroskedastic process (SHP). The SHP is a stationary non-GP. Conditional on a latent process, the SHP is a GP with non-stationary covariance function. As such, the sample paths of this process are more varied and flexible than those produced by a traditional GP model. The SHP model can also recover Gaussian-like sample paths for certain model parameter values. We use maximum likelihood for inference, which is complicated by the

high dimensionality of the latent process. Accordingly, we develop an importance sampling method for likelihood computation and use a low-rank kriging approximation to reconstruct the latent process. Responses at unobserved locations can be predicted using empirical best predictors (EBP) or by empirical best linear unbiased predictors (EBLUP). Palacios and Steele (2006) proposed a similar model in geostatistical modeling. Bayesian inference is performed in their study. But the prior distribution needs to be chosen carefully to improve the convergence and avoid identification problems in parameter estimation.

This work is organized as follows. The detailed properties of the SHP model and estimation methods are introduced in Chapter 2. In Chapter 3, we compare the prediction performance of SHP with the traditional GP model via simulated and real computer experiment data. Similar to the GP model, the predictive variance with the SHP model can also be used in adaptive sampling, and the sampling efficiency is compared for GP and SHP models in Chapter 4. Chapter 5 introduces a low-rank SHP modeling of derivatives together with responses in computer experiments.

Chapter 2

STOCHASTIC HETEROSKEDASTIC PROCESS (SHP) MODEL

2.1 Definition of SHP

One of the approaches to model nonstationarity is through scaling (Banerjee et al. (2003)). Suppose $Z(\boldsymbol{x})$ is a mean 0, variance 1 stationary process with correlation function ρ, and $\sigma(\boldsymbol{x})$ is a pre-specified deterministic function. Then $W(\boldsymbol{x}) = \sigma(\boldsymbol{x})Z(\boldsymbol{x})$ is a nonstationary process. By allowing $\sigma(\boldsymbol{x})$ to be a random process, $W(\boldsymbol{x})$ retains the nonstationary flavor in terms of sample path behavior but has nice probabilistic structure. In particular, if $\sigma(\boldsymbol{x})$ is stationary, then $W(\boldsymbol{x}) = \sigma(\boldsymbol{x})Z(\boldsymbol{x})$ is also stationary, though its sample paths appear nonstationary (a time series version of this phenomenon is shown in Figure 1.5).

The SHP model is defined as:

$$\begin{aligned} y(\boldsymbol{x}) &= \boldsymbol{g}(\boldsymbol{x})^T\boldsymbol{\beta} + W(\boldsymbol{x}), \\ W(\boldsymbol{x}) &= \sigma \exp\left(\frac{\tau\alpha(\boldsymbol{x})}{2}\right) Z(\boldsymbol{x}), \qquad \sigma > 0, \qquad \tau > 0, \end{aligned} \tag{2.1}$$

where $\alpha(\boldsymbol{x})$ and $Z(\boldsymbol{x})$ are two independent stationary GP with mean 0, variance 1 and correlation functions ρ_α and ρ_z, respectively. For most examples in this paper, we take ρ_α and ρ_z to be isotropic correlation functions with range parameters ϕ_α and ϕ_z, respectively. The overall trend is $\boldsymbol{g}(\boldsymbol{x})^T\boldsymbol{\beta}$ and $W(\boldsymbol{x})$ represents the deviation from the trend. The latent process $\alpha(\boldsymbol{x})$ is used to model surface inhomogeneity, which is analogous to the log-volatility component in the stochastic volatility model. By allowing a separable covariance function for the Z process, the isotropic SHP model can be easily extended to a separable SHP model with different range parameters for each input dimension.

2.2 Properties of SHP

In this subsection, we describe some of the key properties of SHP that are useful for modeling.

2.2.1 Unconditional correlation function of SHP

The $y(\cdot)$ process has mean $\boldsymbol{g}(\boldsymbol{x})^T\boldsymbol{\beta}$, variance $\sigma^2\exp(\tau^2/2)$ and kurtosis $3\exp(\tau^2)$. Since the kurtosis is greater than 3, the y process has tail probabilities that are heavier than normal. Using the independence of the α and Z processes, W is isotropic with unconditional correlation given by

$$\rho_y(\boldsymbol{x},\boldsymbol{x}') = \exp\left(-\frac{1}{4}\tau^2 + \frac{1}{4}\tau^2\rho_\alpha(\|\boldsymbol{x}-\boldsymbol{x}'\|)\right)\rho_z(\|\boldsymbol{x}-\boldsymbol{x}'\|). \tag{2.2}$$

Equation (2.2) is derived as follows:

$$\begin{aligned}
\mathrm{Cov}[y(\boldsymbol{x}),y(\boldsymbol{x}')] &= \mathrm{Cov}[W(\boldsymbol{x}),W(\boldsymbol{x}')] \\
&= \sigma^2\mathrm{Cov}\left[\exp\left(\frac{\tau\alpha(\boldsymbol{x})}{2}\right)Z(\boldsymbol{x}),\exp\left(\frac{\tau\alpha(\boldsymbol{x}')}{2}\right)Z(\boldsymbol{x}')\right] \\
&= \sigma^2E\left[\exp\left(\frac{\tau(\alpha(\boldsymbol{x})+\alpha(\boldsymbol{x}'))}{2}\right)Z(\boldsymbol{x})Z(\boldsymbol{x}')\right] \\
&= \sigma^2E\left[\exp\left(\frac{\tau(\alpha(\boldsymbol{x})+\alpha(\boldsymbol{x}'))}{2}\right)\right]E[Z(\boldsymbol{x})Z(\boldsymbol{x}')] \\
&= \sigma^2E\left[\exp\left(\frac{\tau(\alpha(\boldsymbol{x})+\alpha(\boldsymbol{x}'))}{2}\right)\right]\rho_z(\boldsymbol{x},\boldsymbol{x}').
\end{aligned} \tag{2.3}$$

Since α is a GP with mean 0, variance 1 and correlation function ρ_α, $\alpha(\boldsymbol{x})+\alpha(\boldsymbol{x}')$ also follows the Gaussian distribution with mean 0 and variance $2+2\rho_\alpha(\boldsymbol{x},\boldsymbol{x}')$. From the moment generating function of the normal distribution, we get

$$E\left[\exp\left(\frac{\tau(\alpha(\boldsymbol{x})+\alpha(\boldsymbol{x}')}{2}\right)\right] = \exp\left\{\frac{\tau^2}{4}+\frac{\tau^2}{4}\rho_\alpha(\boldsymbol{x},\boldsymbol{x}')\right\}. \tag{2.4}$$

Substituting (2.4) into (2.3), the unconditional covariance is given by

$$\mathrm{Cov}[y(\boldsymbol{x}),y(\boldsymbol{x}')] = \sigma^2\exp\left\{\frac{\tau^2}{4}+\frac{\tau^2}{4}\rho_\alpha(\boldsymbol{x},\boldsymbol{x}')\right\}\rho_z(\boldsymbol{x},\boldsymbol{x}'). \tag{2.5}$$

Dividing (2.5) by $\mathrm{Var}(y(\boldsymbol{x})) = \sigma^2\exp(\tau^2/2)$, the unconditional correlation (2.2) follows easily.

Limiting behavior of SHP model

This unconditional correlation function in (2.2) motivates us to explore the limiting processes when $\phi_\alpha \to 0$ and $\phi_\alpha \to \infty$. When $\phi_\alpha = 0$, the α process degenerates to a single $N(0,1)$ random variable and the unconditional correlation function for y becomes ρ_z. While the process y remains non-Gaussian, a single realization of y is not distinguishable from a realization of a GP. In fact it is a realization of a GP, multiplied by a single scale factor. On the other hand, as $\phi_\alpha \to \infty$, the unconditional correlation function converges to

$$\rho(\boldsymbol{x}, \boldsymbol{x}') = \begin{cases} 1, & \text{if } \|\boldsymbol{x} - \boldsymbol{x}'\| = 0, \\ \exp(-\tau^2/4)\rho_z(\boldsymbol{x}, \boldsymbol{x}'), & \text{if } \|\boldsymbol{x} - \boldsymbol{x}'\| > 0. \end{cases} \tag{2.6}$$

In modeling spatial data, one often allows for possible measurement error and microscale variability. It is sometimes called "nugget" and the correlation function with relative nugget δ, $0 \le \delta \le 1$ is written as

$$\rho^\delta(\boldsymbol{x}, \boldsymbol{x}') = \begin{cases} 1, & \text{if } \|\boldsymbol{x} - \boldsymbol{x}'\| = 0, \\ (1-\delta)\rho_z(\boldsymbol{x}, \boldsymbol{x}'), & \text{if } \|\boldsymbol{x} - \boldsymbol{x}'\| > 0. \end{cases}$$

Thus, as $\phi_\alpha \to \infty$, the unconditional correlation function of SHP is simply ρ_z adding a relative nugget $\delta = 1 - \exp(-\tau^2/4)$.

Effect of ϕ_α

The effect of varying the correlation parameter ϕ_α on the correlation function can be seen from Figure 2.1. The correlation decreases with ϕ_α increasing for each value of h, the distance between sample points. As ϕ_α approaches infinity, one can see the emergence of "smoothed nugget", by which we mean nugget-like behavior, without an actual discontinuity at the origin. At $\phi_\alpha = \infty$, we have a full-fledged nugget of size $1 - \exp(\tau^2/4)$. In the other direction, as ϕ_α approaches zero, the correlation function decays smoothly. But the effect of ϕ_α on the unconditional correlation function also relates to the value of ϕ_z. As we increase ϕ_z from 5 to

100, comparing Figure 2.1 to Figure 2.2, the decay of the correlation function is dominated by the large value of ϕ_z. The correlation function with ϕ_α ranging from 0 too 100 are much less varied in Figure 2.2 ($\phi_z = 100$) than in Figure 2.1 ($\phi_z = 5$).

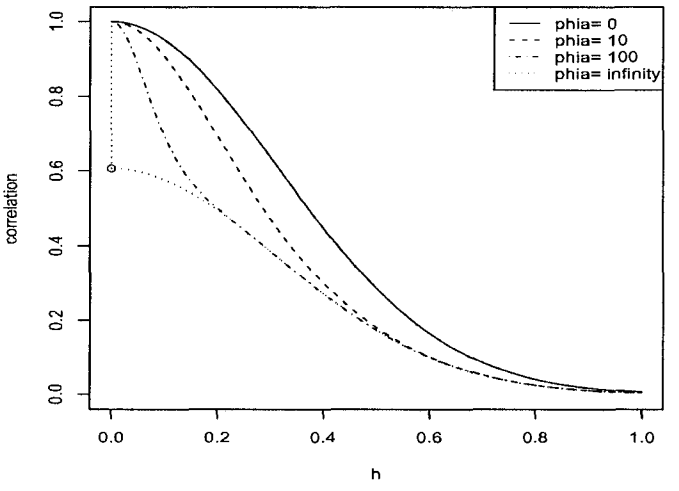


Figure 2.1: Effect of ϕ_α (with $\phi_z = 5$ and $\tau^2 = 2$) on the unconditional SHP correlation function. Both ρ_α and ρ_z are Gaussian correlation functions.

Effect of τ^2

When $\tau^2 = 0$, the SHP recovers a GP with correlation function ρ_z. Under fixed values of ϕ_z and ϕ_α, the correlation decreases with τ^2 increasing, as shown in Figure 2.3. For small values of ϕ_z and large values of ϕ_α, the emergence of a smooth nugget gets more obvious with increasing τ^2. As shown in (2.6), the value of τ^2 decides the scale of nugget in the limiting behavior. The larger the value of τ^2, the larger the value of the smoothed nugget. As ϕ_z increases, the decay of the correlation function shown in Figure 2.4 is again dominated by the large value of ϕ_z. Correlation functions with τ^2 ranging from 0 to 6 are much less varied in Figure 2.4 ($\phi_z = 100$) than in Figure 2.3 ($\phi_z = 5$).

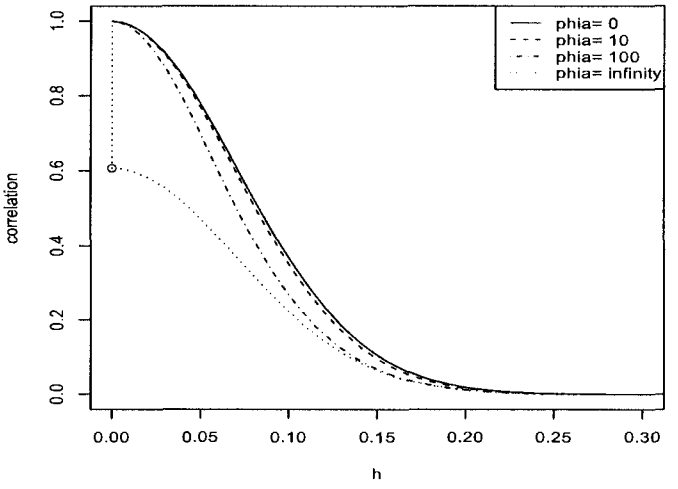


Figure 2.2: Effect of ϕ_α (with $\phi_z = 100$ and $\tau^2 = 2$) on the unconditional SHP correlation function. Both ρ_α and ρ_z are Gaussian correlation functions.

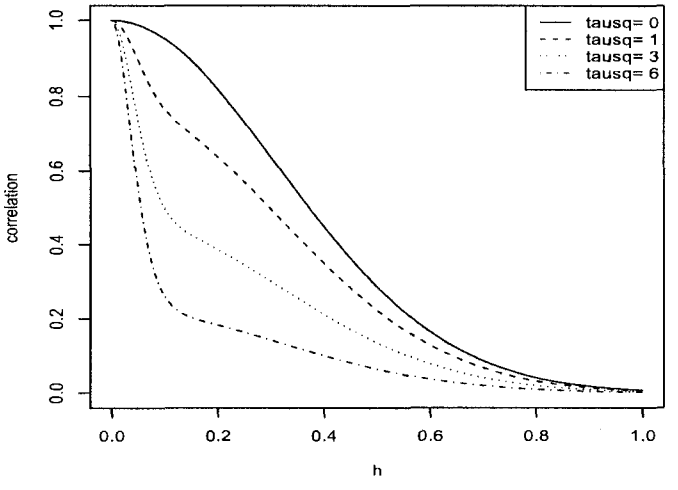


Figure 2.3: Effect of τ^2 (with $\phi_z = 5$ and $\phi_\alpha = 200$) on the unconditional SHP correlation function. Both ρ_α and ρ_z are Gaussian correlation functions.

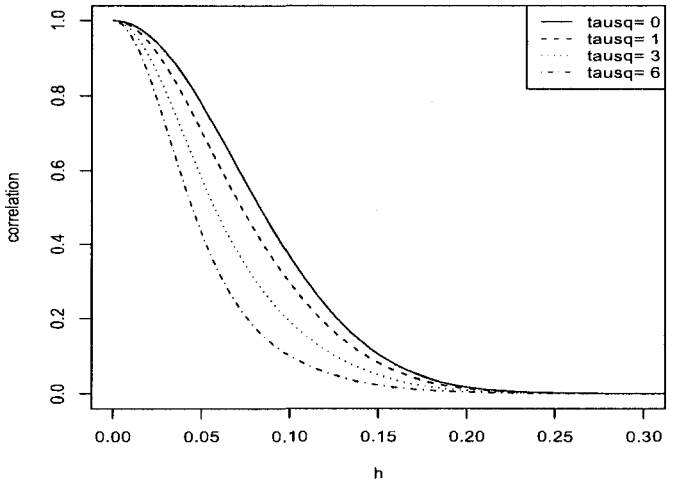


Figure 2.4: Effect of τ^2 (with $\phi_z = 100$ and $\phi_\alpha = 200$) on the unconditional SHP correlation function. Both ρ_α and ρ_z are Gaussian correlation functions.

Confounding effect of τ^2 and ϕ_α

From the previous correlation function plots, we see the potential for a confounding effect between ϕ_α and τ^2, i.e. the different values of ϕ_α and τ^2 can give the same correlation function. The overlap of two correlation functions in each panel of Figure 2.5 reveals the existence of this confounding effect between ϕ_α and τ^2 for different ϕ_z values. But the degree of confounding depends on the value of ϕ_z and ϕ_α. If $\phi_z \leq \phi_\alpha$, the confounding effect exists for a relatively small range of τ^2 values, such as those given in the correlation function plots in panels (b) and (c). The range of ϕ_α is related to the value of ϕ_z: the larger the value of ϕ_z, the larger the range of ϕ_α values for which the confounding effect exists. If $\phi_z > \phi_\alpha$, the confounding effect between ϕ_α and τ^2 holds for a larger range of τ^2 values, such as those given in the correlation plots in panel (a) and (d).

The confounding effect can be explained by the dominance of ϕ_z on the shape of the correlation function. As we observed in the previous section about the effect of model parameters on the correlation function, the decay rate of the correlation

function is most influenced by the value of ϕ_z. For small values of ϕ_z, the values of ϕ_α and τ^2 have more influence on the shape of the correlation function. But values of ϕ_α and τ^2 have much less influence on the shape of the correlation function for larger values of ϕ_z. This confounding effect will lead to identification problems in estimating SHP model parameters.

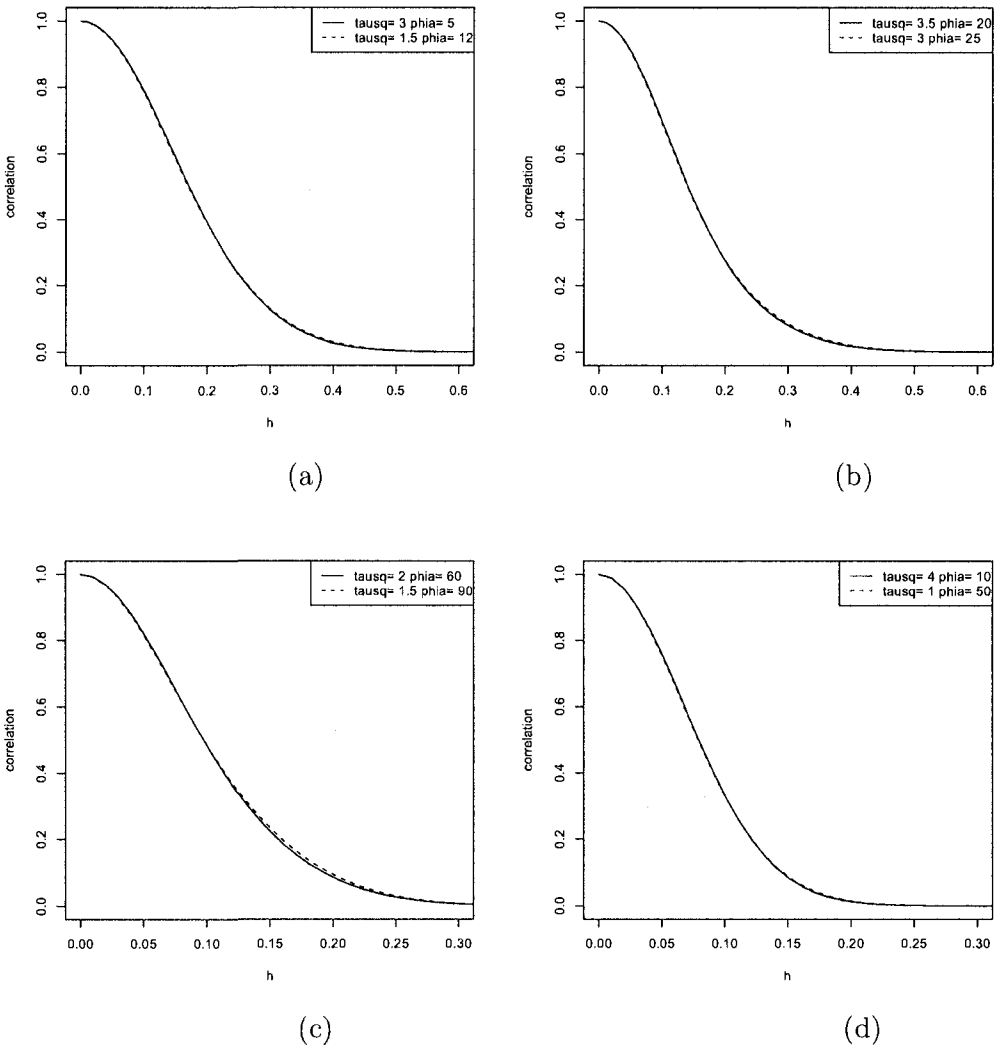


Figure 2.5: Confounding effect between ϕ_α and τ^2 for unconditional SHP correlation function. (a) $\phi_z = 10$. (b) $\phi_z = 20$. (c) $\phi_z = 50$. (d) $\phi_z = 100$.

2.2.2 Conditional covariance function of SHP

Conditioning on the latent process $\boldsymbol{\alpha}$, the covariance function of the process between two locations is

$$\mathrm{Cov}(y(\boldsymbol{x}), y(\boldsymbol{x}')|\boldsymbol{\alpha}) = \sigma^2 \exp\left(\frac{\tau\alpha(\boldsymbol{x})}{2}\right) \rho_z(\boldsymbol{x}, \boldsymbol{x}') \exp\left(\frac{\tau\alpha(\boldsymbol{x}')}{2}\right). \tag{2.7}$$

Equations (2.2) and (2.7) indicate that the W process is conditionally heteroscedastic and unconditionally stationary. The inhomogeneity of the conditional covariance function leads to versatile sample paths of SHP, which is shown in the next section.

2.2.3 Sample paths

In this section, we explore the versatile properties of sample paths from the SHP model. We generate 1-d and 2-d realizations from SHP with different model parameter values. These realizations are compared with realizations from GP models. For smooth GP realizations, as shown in Figure 2.6 (a), the SHP model can give very similar realizations with small values of ϕ_z, ϕ_α and τ^2, as shown in Figure 2.6 (b). The realizations in Figure 2.6 (c) and (d) show that for $\phi_z = 20$ the local inhomogeneity of SHP realizations increases as ϕ_α and/or τ^2 increases. For rough GP realizations, as shown in Figure 2.7 (a), the SHP model again can produce similar realizations, as shown in Figure 2.7 (b). The realizations in Figure 2.7 (c) are nearly identical, showing that ϕ_α has not much influence at this large value of $\phi_z = 100$. Figure 2.7 (d) again shows local inhomogeneity, as the value of τ^2 is increased.

For GP realizations similar to those from SHP in panel (a) and (b) of Figure 2.6 or Figure 2.7, their unconditional correlation functions are quite close to each other, as shown in Figure 2.8.

The above 1-d examples show that GP and SHP models can have similar unconditional correlation functions and can produce similar realizations. On the other hand, the realizations can differ from each other even with similar unconditional

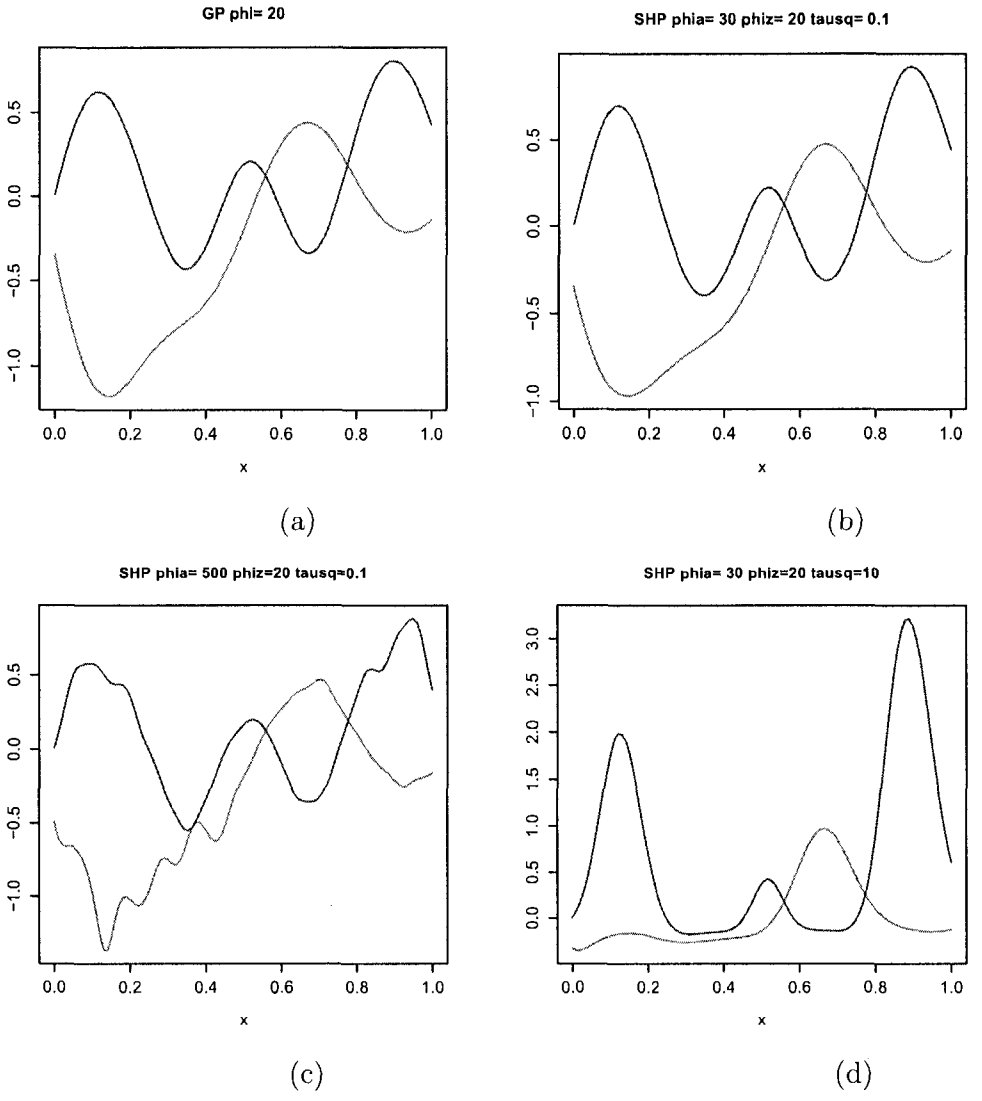


Figure 2.6: Two 1-d realizations of GP vs SHP for small value of ϕ_z. Gaussian correlation functions are used for the GP and for the latent process and random field in SHP. The same set of random noise sequences is used from panel to panel in generating the realizations.

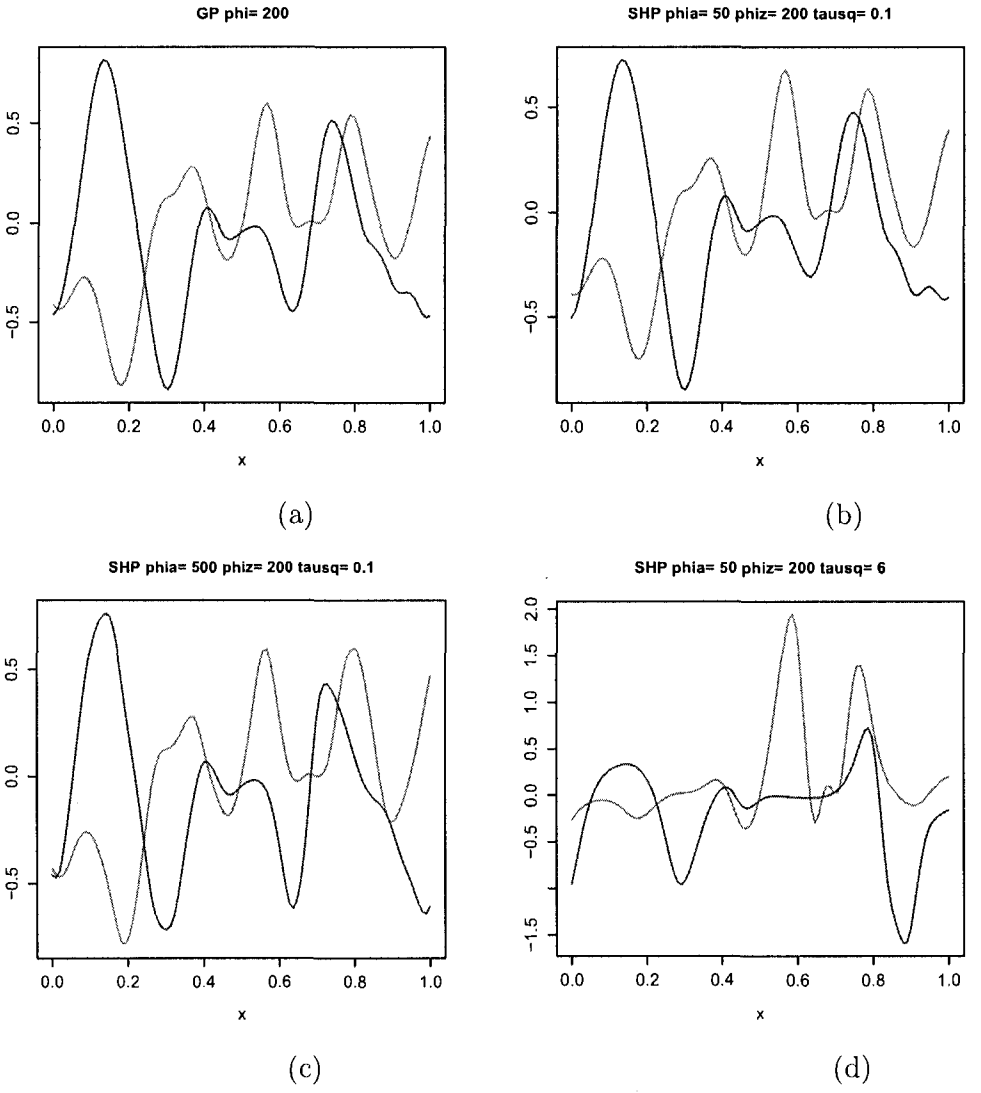
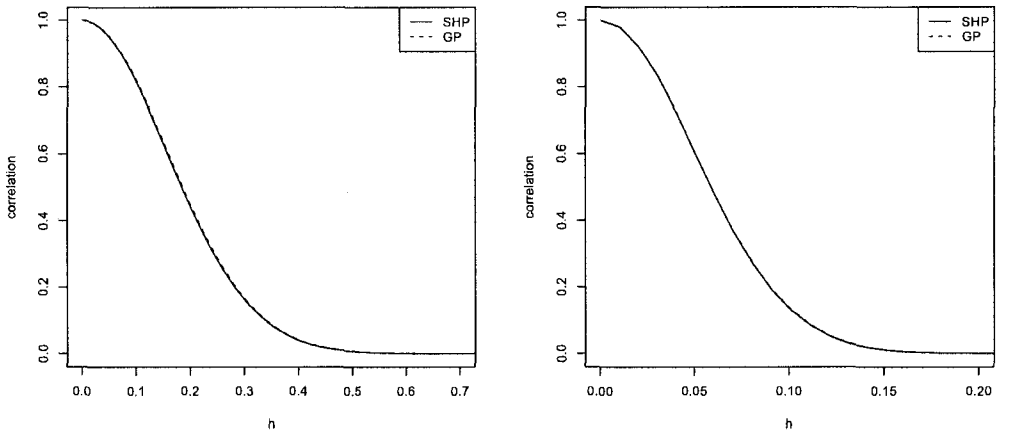


Figure 2.7: Two 1-d realizations of GP vs SHP for small value of ϕ_z. Gaussian correlation functions are used for the GP and for the latent process and random field in SHP. The same set of random noise sequences is used from panel to panel in generating the realizations.



(a) (b)

Figure 2.8: Unconditional correlation function plots. (a) The correlation function plots for GP and SHP sample paths in Figure 2.6 (a) and (b): GP with $\phi = 20$; SHP with $\tau^2 = 0.1$, $\phi_\alpha = 30$, and $\phi_z = 20$. (b) The correlation function plots for GP and SHP sample paths in Figure 2.7 (a) and (b): GP with $\phi = 200$; SHP with $\tau^2 = 0.1$, $\phi_\alpha = 50$, and $\phi_z = 200$.

correlation functions for GP and SHP. For example, the correlation functions for the GP and SHP models that generated the 2-d surfaces in panels (a) and (b) of Figure 2.9 are nearly identical, but the 2-d surfaces themselves are markedly different. The realized surface from the GP has a relatively unform degree of smoothness over the whole input domain. In contrast, the realization from the SHP model has some local volatilities. The smoothness of the SHP realization varies over different parts of the region, which is due largely to the inhomogeneity of the conditional covariance function. The local volatility is less obvious in panel (d), with smaller ϕ_α/ϕ_z, and least obvious in panel (c), with $\phi_\alpha/\phi_z < 1$.

Confounding effect of model parameters on sample paths

In the previous section, we showed the confounding effect between ϕ_α and τ^2 on the correlation functions under certain values of ϕ_z. Figure 2.10 indicates the existence of a possible confounding effect between ϕ_α and ϕ_z. Different combinations of ϕ_α and ϕ_z produce realizations quite similar to each other. For a finite sample

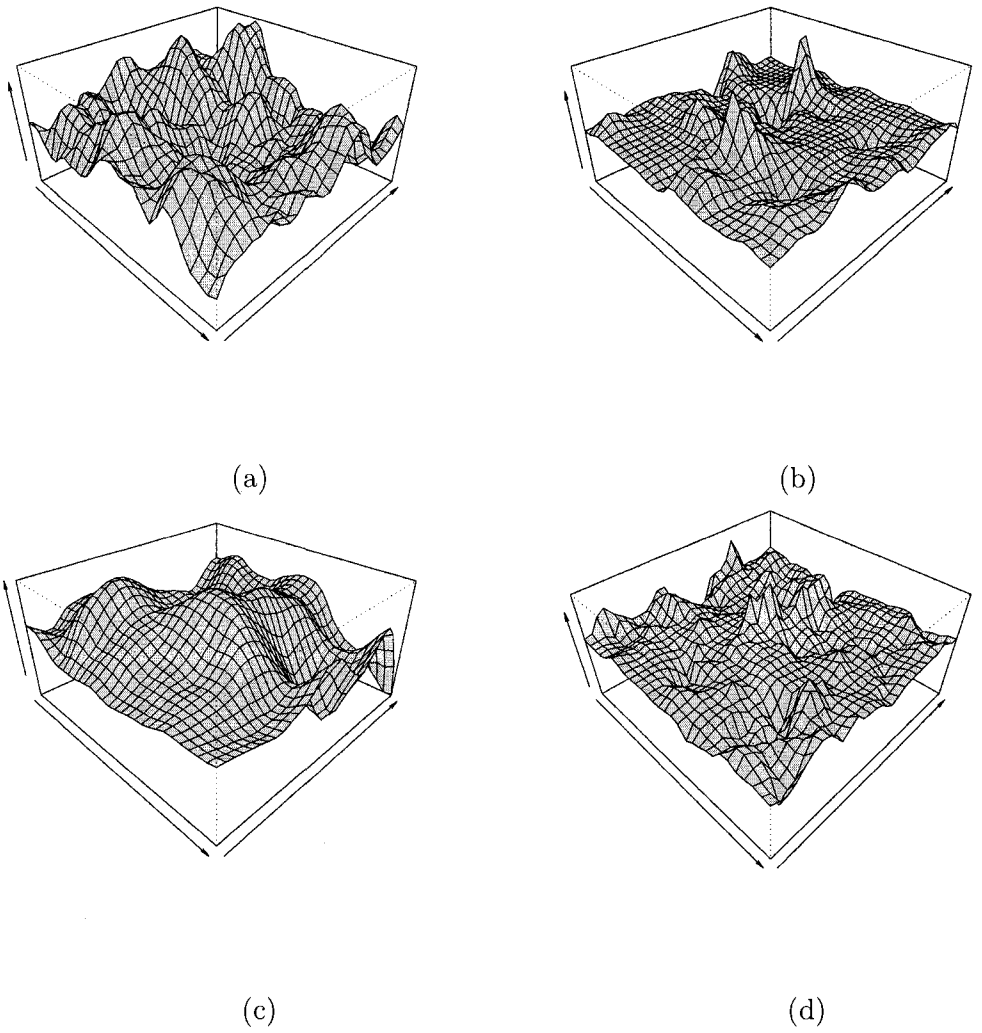


Figure 2.9: 2-d surface plots. Panel (a) is a GP surface with $\phi = 6.4$. Panel (b) is a SHP surface with $\phi_\alpha = 8.5$ and $\phi_z = 4.4$ and $\tau^2 = 2$. Panel (c) is a SHP surface with $\phi_\alpha = 1$ and $\phi_z = 3.3$ and $\tau^2 = 2$. Panel (d) is a SHP surface with $\phi_\alpha = 12$ and $\phi_z = 11$ and $\tau^2 = 2$. The unconditional correlation functions for GP and SHP in (a) and (b) are nearly identical. All correlation functions are Matérn with $\nu = 2.5$. The same set of random noise sequences is used from panel to panel in generating the realizations.

size, it will be difficult to tell which combination of ϕ_z and ϕ_α gives such a sample path. This confounding effect will lead to an identification problem in the parameter estimation. As we can see in later sections, the likelihood of the SHP model favors large values of ϕ_z. Some methods might be able to alleviate these identification problems, such as restricting different parameter ranges in the optimization process, adding a penalty in the likelihood computation for large values of ϕ_z or choosing informative priors in a fully Bayesian approach. In our study, instead of solving the identification problem in parameter estimation, we are more interested in model performance in terms of prediction accuracy. The parameter estimates are obtained by maximum likelihood.

2.3 Likelihood Calculation and Parameter Estimation

Due to the presence of the latent process α in the SHP model, there are no closed-form expressions for the likelihood function, and we consider simulation-based alternatives for its computation. We focus on importance sampling methods, which have been proven successful in related time series models (Danielsson and Richard (1993), Durbin and Koopmans (1997), Davis and Rodriguez-Yam (2005)). We describe the strategy of likelihood calculation and importance density derivation in Section 2.3.1 and Section 2.3.2, followed by parameter estimation and the latent process estimation in Section 2.3.4 and Section 2.3.5.

2.3.1 Likelihood calculation

Let $\boldsymbol{y} := (y_1, ..., y_n)$ denote the vector of observations, $\boldsymbol{\alpha} := (\alpha_1, ..., \alpha_n)$ the vector of the latent process at observed locations and $\boldsymbol{\psi} := (\boldsymbol{\theta}, \phi_\alpha)$ the model parameters. Here $\boldsymbol{\theta} := (\sigma^2, \tau^2, \phi_z, \boldsymbol{\beta})$. The joint density of $(\boldsymbol{y}, \boldsymbol{\alpha})$ of the SHP model is given by

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{\alpha}|\boldsymbol{\psi}) &= p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) p(\boldsymbol{\alpha}|\phi_\alpha) \\ &= p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) |R_\alpha|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\boldsymbol{\alpha}^T R_\alpha^{-1} \boldsymbol{\alpha}\right) (2\pi)^{-\frac{n}{2}}, \end{aligned} \tag{2.8}$$

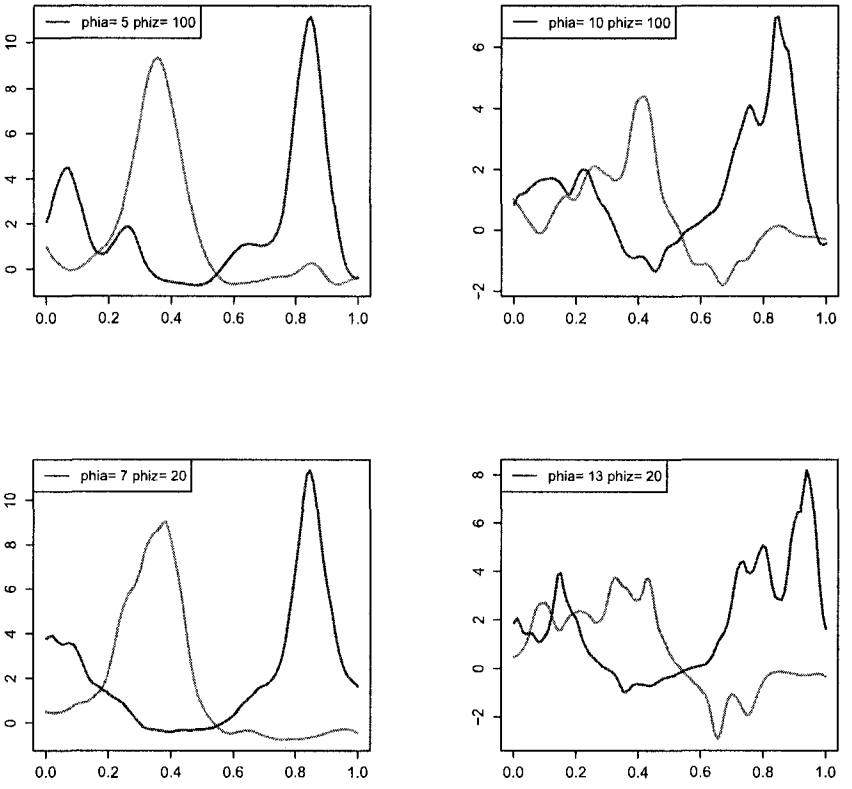


Figure 2.10: Confounding effect between ϕ_α and ϕ_z in sample paths. In spite of having different ϕ_α, ϕ_z values between the top and bottom panels, the two left panels have similar sample paths, as do the two right panels.

where R_α is the correlation matrix for $\boldsymbol{\alpha}$, which only depends on ϕ_α. The conditional distribution $p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta})$ is given by

$$p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) = \mathrm{N}\left(G^T\boldsymbol{\beta}, \sigma^2 \mathrm{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\} R_z \mathrm{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}\right), \tag{2.9}$$

where R_z is the correlation matrix of the Z process. It follows that the likelihood of the observed data is given by the n-fold integral

$$L(\boldsymbol{\psi}; \boldsymbol{y}) = \int p(\boldsymbol{y}, \boldsymbol{\alpha}|\boldsymbol{\psi}) d\boldsymbol{\alpha} = \int p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) p(\boldsymbol{\alpha}|\phi_\alpha) d\boldsymbol{\alpha}. \tag{2.10}$$

2.3.2 Importance density

The likelihood (2.10) cannot be computed explicitly. There are some simulation-based procedures in the literature to approximate the likelihood (see Robert and Casella (1999)). Monte Carlo integration is a straightforward way to approximate the likelihood. Due to the high dimensionality of the latent process in the SHP model, naive Monte Carlo integration is very inefficient. Importance sampling is introduced to increase computational efficiency and improve the accuracy of the approximation.

As mentioned in Durbin and Koopmans (1997), to achieve efficiency, the importance density should be chosen to be close to the posterior density $p(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$. It can be easily shown that if $p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$ is exactly equal to $p(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$, a sample of only $N = 1$ is required for accurate likelihood calculation. Some methodologies have been developed to find efficient importance densities; see Danielsson and Richard (1993) and Durbin and Koopmans (1997). In this paper, we refer to the likelihood approximation method in Davis and Rodriguez-Yam (2005) to obtain a density $p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$ as an approximation to $p(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$. Their work was applied to state-space models and recursive prediction algorithms, such as the Kalman recursions or innovations algorithm, to accelerate the calculation in finding an importance density.

To find the importance density $p_a(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})$, Davis and Rodriguez-Yam (2005) use a Taylor series expansion of $\log p(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})$ in a neighborhood of the posterior mode of $p(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})$. The log-density of $(\boldsymbol{y},\boldsymbol{\alpha})$ is given by

$$\log p(\boldsymbol{y},\boldsymbol{\alpha}|\boldsymbol{\psi}) = -\frac{n}{2}\log(2\pi) + \frac{1}{2}\log|R_\alpha|^{-1} + l(\boldsymbol{\theta};\boldsymbol{y}|\boldsymbol{\alpha}) - \frac{1}{2}\boldsymbol{\alpha}^T R_\alpha^{-1}\boldsymbol{\alpha}, \qquad (2.11)$$

where $l(\boldsymbol{\theta};\boldsymbol{y}|\boldsymbol{\alpha}) := \log p(\boldsymbol{y}|\boldsymbol{\alpha},\boldsymbol{\theta})$.

Let

$$\boldsymbol{k}^* := -\frac{\partial}{\partial\boldsymbol{\alpha}} l(\boldsymbol{\theta};\boldsymbol{y}|\boldsymbol{\alpha})|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}^*},$$

where $\boldsymbol{\alpha}^*$ is the mode of $p(\boldsymbol{y},\boldsymbol{\alpha}|\boldsymbol{\psi})$, which solves $\partial l(\boldsymbol{\psi};\boldsymbol{y},\boldsymbol{\alpha})/\partial\boldsymbol{\alpha} = 0$. From (2.8), it follows that $\boldsymbol{k}^* = R_\alpha^{-1}\boldsymbol{\alpha}^*$.

Hence, the second order Taylor expansion of $l(\boldsymbol{\theta};\boldsymbol{y}|\boldsymbol{\alpha})$ around $\boldsymbol{\alpha}^*$ given the latent process α is

$$l(\boldsymbol{\theta};\boldsymbol{y}|\boldsymbol{\alpha}) = h^* + \boldsymbol{\alpha}^{*T}R_\alpha^{-1}(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*) - \frac{1}{2}(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*)^T K^*(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*) + e(\boldsymbol{\alpha},\boldsymbol{\alpha}^*), \quad (2.12)$$

where $h^* := l(\boldsymbol{\theta};\boldsymbol{y}|\boldsymbol{\alpha})|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}^*}$, $K^* := -\frac{\partial^2}{\partial\alpha\partial\alpha^T} l(\boldsymbol{\theta};\boldsymbol{y}|\boldsymbol{\alpha})|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}^*}$ and $e(\boldsymbol{\alpha},\boldsymbol{\alpha}^*)$ is the corresponding remainder. Thus,

$$\begin{aligned} l(\boldsymbol{\psi};\boldsymbol{y},\boldsymbol{\alpha}) &= -\frac{n}{2}\log(2\pi) + \frac{1}{2}\log|R_\alpha|^{-1} + h^* - \frac{1}{2}\boldsymbol{\alpha}^{*T}R_\alpha^{-1}\boldsymbol{\alpha}^* \\ &\quad -\frac{1}{2}(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*)^T(K^* + R_\alpha^{-1})(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*) + e(\boldsymbol{\alpha},\boldsymbol{\alpha}^*). \qquad (2.13) \end{aligned}$$

Let $p_a(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})$ be the approximation of the posterior when the remainder term is omitted. It follows that

$$p_a(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi}) = \mathrm{N}\left(\boldsymbol{\alpha}^*, (K^* + R_\alpha^{-1})^{-1}\right) \qquad (2.14)$$

and in the SHP model,

$$\begin{aligned} &K^* = \frac{\tau^2}{4\sigma^2}(B + \mathrm{diag}\{\boldsymbol{c}\}), \\ &B = \mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}\mathrm{diag}\{\boldsymbol{y}-\boldsymbol{g}^T\boldsymbol{\beta}\}R_z^{-1}\mathrm{diag}\{\boldsymbol{y}-\boldsymbol{g}^T\boldsymbol{\beta}\}\mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}, \\ &\boldsymbol{c} = (\exp(-\tau\boldsymbol{\alpha}/2))^T\,\mathrm{diag}\{\boldsymbol{y}-\boldsymbol{g}^T\boldsymbol{\beta}\}R_z^{-1}\mathrm{diag}\{\boldsymbol{y}-\boldsymbol{g}^T\boldsymbol{\beta}\}\mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}. \end{aligned} \qquad (2.15)$$

Using $p_a(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})$ as an importance density function to implement Monte Carlo integration, the integral in (2.10) can be rewritten as

$$\begin{aligned} L(\boldsymbol{\psi};\boldsymbol{y}) &= \int \frac{p(\boldsymbol{y}|\boldsymbol{\alpha},\boldsymbol{\theta})p(\boldsymbol{\alpha}|\phi_\alpha)}{p_a(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})} p_a(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})d\boldsymbol{\alpha} \\ &= \mathrm{E}_a\left[\frac{p(\boldsymbol{y}|\boldsymbol{\alpha},\boldsymbol{\theta})p(\boldsymbol{\alpha}|\phi_\alpha)}{p_a(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})}\right]. \end{aligned} \tag{2.16}$$

If $\boldsymbol{\alpha}^{(1)},...,\boldsymbol{\alpha}^{(N)}$ are drawn from $p_a(\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi})$, then (2.16) can be approximated by

$$L(\boldsymbol{\psi};\boldsymbol{y}) \approx \frac{1}{N}\sum_{i=1}^{N}\left[\frac{p(\boldsymbol{y}|\boldsymbol{\alpha}^{(i)},\boldsymbol{\theta})p(\boldsymbol{\alpha}^{(i)}|\phi_\alpha)}{p_a(\boldsymbol{\alpha}^{(i)}|\boldsymbol{y},\boldsymbol{\psi})}\right]. \tag{2.17}$$

We use common noise sequences to draw $\boldsymbol{\alpha}^{(i)}$, $i = 1,...,N$ in order to compute the likelihood (2.17), that is, the replicates of $\boldsymbol{\alpha}^{(1)},...,\boldsymbol{\alpha}^{(N)}$ generated from the importance density are based on the same random noise denoted by $\boldsymbol{u}^{(1)},...,\boldsymbol{u}^{(N)}$. It is a common practice to use "common random numbers" in likelihood calculation using an importance density. The common random numbers ensures that the likelihood is also smooth. Due to the complexity of the likelihood calculation, we provide more details of the implementation in the following subsections.

2.3.3 Finding the mode α^*

In order to get the importance density (2.14), we need to find the posterior mode $\boldsymbol{\alpha}^*$ by maximizing (2.11). Since the dimensionality of $\boldsymbol{\alpha}$ is the same as the number of observations, it is difficult to find $\boldsymbol{\alpha}^*$ especially for a large data set. We adopt a low-rank approximation for the latent process $\boldsymbol{\alpha}$ in order to reduce the dimensionality in this optimization procedure.

An approach for low-rank approximation is to use process convolution (Higdon (2002)). One way to construct a continuous GP is through its covariance function $c(\cdot)$, so the dependence between any two points depends on the distance between them. Another way to construct a GP is to convolve a continuous white noise

process $x(\boldsymbol{s})$, $\boldsymbol{s} \in \boldsymbol{S}$ with a smoothing kernel $k(\boldsymbol{s})$ so that

$$z(\boldsymbol{s}) = \int_S k(\boldsymbol{\mu} - \boldsymbol{s}) dx(\boldsymbol{\mu}). \tag{2.18}$$

The resulting covariance function for $z(\boldsymbol{s})$ depends only on the displacement vector $\boldsymbol{d} = \boldsymbol{s} - \boldsymbol{s}'$ and is given by

$$c(\boldsymbol{d}) = \text{Cov}(z(\boldsymbol{s}), z(\boldsymbol{s}')) = \int_S k(\boldsymbol{\mu} - \boldsymbol{s}) k(\boldsymbol{\mu} - \boldsymbol{s}') d\boldsymbol{\mu} = \int_S k(\boldsymbol{\mu} - \boldsymbol{d}) k(\boldsymbol{\mu}) d\boldsymbol{\mu}. \tag{2.19}$$

The convolution of this process can be equivalently defined using some covariance function in $\mathbb{R}^d$. There is a one to one relationship between the smoothing kernel $k(\boldsymbol{d})$ and the covariance function $c(\boldsymbol{d})$, given either $\int_{R^p} k(\boldsymbol{s}) d\boldsymbol{s} < \infty$ and $\int_{R^p} k^2(\boldsymbol{s}) d\boldsymbol{s} < \infty$ or $c(\boldsymbol{s})$ is integrable and positive definite (Higdon (2002)). For example, a smoothing kernel $k(\boldsymbol{s}) \propto \exp\{-\frac{1}{2}||\boldsymbol{s}||^2\}$ has a one-to-one relationship with Gaussian covariance function $c(d) \propto \exp\{-\frac{1}{2}||\frac{d}{\sqrt{2}}||^2\}$. Note this relationship is no longer one-to-one if the process is not isotropic. In this case, multiple kernels can give rise to the same covariance function.

Higdon (2002) applies the process convolution models (2.18) in dimension reduction to handle large data sets. One can choose to restrict the latent process $x(\boldsymbol{s})$ to be nonzero at the spatial sites $\boldsymbol{\omega}_1, ..., \boldsymbol{\omega}_m$ in $\mathcal{S}$. Each $x(\boldsymbol{\omega}_j)$ is then modeled as an independent draw from a $N(0, \sigma_x^2)$ distribution. The resulting continuous GP is then approximated by

$$z(\boldsymbol{s}) = \sum_{j=1}^{m} x(\boldsymbol{\omega}_j) k(\boldsymbol{s} - \boldsymbol{\omega}_j) \tag{2.20}$$

where $k(\cdot - \boldsymbol{\omega}_j)$ is a kernel centered at $\boldsymbol{\omega}_j$.

An alternative for low-rank approximation is *low-rank kriging*. In the context of nonparametric regression, the kriging method is regarded as one kind of radial smoothing and it is BLUP for a mixed model of the form (Ruppert et al. (2003), pages 238 – 260):

$$\boldsymbol{y} = G\boldsymbol{\beta} + B_c\boldsymbol{\omega} + \boldsymbol{\epsilon}, \tag{2.21}$$

where G is the design matrix for fixed effects, and

$$B_c \equiv [C(||\boldsymbol{x}_i - \boldsymbol{x}_j||)]_{i,j=1}^n \tag{2.22}$$

for some covariance function C. The BLUP for $y(\boldsymbol{x})$ involves linear combination of

$$C(|\boldsymbol{x} - \boldsymbol{x}_j|),\ 1 \leq j \leq n.$$

The above smoother is *full-rank*. Since the smoother involves the inverse of a $n \times n$ matrix B_c, the computational cost will increase dramatically as the sample size increases. Ruppert et al. (2003) introduce a low-rank extension for radial smoothers. In what follows, we describe their method with a few changes to make their notations to be consistent with ours.

Let $\boldsymbol{\kappa}_1, ..., \boldsymbol{\kappa}_J$ be a set of knot locations. Then an approximation based on the smaller set of basis functions

$$C(|\boldsymbol{x} - \boldsymbol{\kappa}_j|),\ 1 \leq j \leq J,$$

arises from fitting the mixed model

$$\boldsymbol{y} = G\boldsymbol{\beta} + B_J\boldsymbol{\omega} + \boldsymbol{\epsilon},\ \text{Cov}(\boldsymbol{\omega}) = \sigma_\omega^2 \Omega_J^{-1}, \tag{2.23}$$

where

$$B_J \equiv \underset{1\leq i\leq n, 1\leq j\leq J}{[C(|\boldsymbol{x}_i - \boldsymbol{\kappa}_j|)]},\ \text{and}\ \Omega_J \equiv \underset{1\leq j,j'\leq J}{[C(|\boldsymbol{\kappa}_j - \boldsymbol{\kappa}_j'|)]}. \tag{2.24}$$

In this study, low-rank kriging is used to approximate the latent process α. Let $\boldsymbol{\kappa}_1, ..., \boldsymbol{\kappa}_J$ be a set of knot locations. the resulting continuous α process is then approximated by

$$\boldsymbol{\alpha} = B\boldsymbol{\omega},\ \boldsymbol{\omega} \sim \text{N}\left(\mathbf{0}, \Omega^{-1}\right), \tag{2.25}$$

where $B \equiv [\rho_\alpha(|\boldsymbol{x}_i - \boldsymbol{\kappa}_j|)]_{1\leq i\leq n, 1\leq j\leq J}$ and $\Omega \equiv [\rho_\alpha(|\boldsymbol{\kappa}_j - \boldsymbol{\kappa}_j'|)]_{1\leq j,j'\leq J}$. We substitute (2.25) into (2.11) and maximize the likelihood with respect to $\boldsymbol{\omega}$ to get $\hat{\boldsymbol{\omega}}$. Then $\boldsymbol{\alpha}^*$ is approximated by $B\hat{\boldsymbol{\omega}}$.

2.3.4 Parameter estimation

For the SHP model, simulation results show that the likelihood tends to be flat for a wide range of large σ^2 values. Unlike the GP model, σ^2 cannot be profiled out of the likelihood in (2.10). This suggests the difficulty of estimating σ^2 from the likelihood. We propose an alternative way to alleviate this problem.

Consider the sample variance calculated by

$$s^2 = \frac{1}{2n(n-1)} \sum_j \sum_k (y(\boldsymbol{x}_j) - y(\boldsymbol{x}_k))^2.$$

Incorporating the correlations in the observations, the expected value of the sample variance is

$$\begin{aligned} E(s^2) &= \frac{1}{2n(n-1)} E\left(\sum_j \sum_k (y(\boldsymbol{x}_j) - y(\boldsymbol{x}_k))^2 \right) \\ &= \frac{1}{2n(n-1)} \left(2n^2\sigma^2 \exp(\tau^2/2) - 2 \sum_j \sum_k \rho_y(j,k) \right), \end{aligned}$$

Hence, a corrected σ^2 by incorporating the sample correlation is given by

$$\sigma^2 = \frac{n(n-1)\exp(-\tau^2/2)E(s^2)}{n^2 - \sum_i \sum_j \rho_y(i,j)}. \tag{2.26}$$

In our estimation procedure, we propose to fix σ^2 at the sample variance, and maximize (2.17) only with respect to $(\tau^2, \phi_\alpha, \phi_z, \boldsymbol{\beta})$ to get estimates $(\hat{\tau}^2, \hat{\phi_\alpha}, \hat{\phi_z}, \hat{\boldsymbol{\beta}})$. Then, by substituting $E(s^2)$ with s^2 and plugging other parameter estimates into (2.26), an approximately unbiased estimator for σ^2 is given by

$$\hat{\sigma}^2 = \frac{n(n-1)\exp(-\tau^2/2)s^2}{n^2 - \sum_i \sum_j \hat{\rho}_y(i,j)}. \tag{2.27}$$

2.3.5 Estimation of function of volatility

If $\boldsymbol{\psi}$ were known, a function $f(\cdot)$ of the latent process α at observed locations can be estimated as the conditional expectation $\mathrm{E}[f(\boldsymbol{\alpha})|\boldsymbol{y}, \boldsymbol{\psi}]$, given by

$$\begin{aligned}
\mathrm{E}[f(\boldsymbol{\alpha})|\boldsymbol{y}, \boldsymbol{\psi}] &= \int f(\boldsymbol{\alpha}) p(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi}) d\boldsymbol{\alpha} \\
&= \int f(\boldsymbol{\alpha}) \frac{p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) p(\boldsymbol{\alpha}|\phi_\alpha)}{p(\boldsymbol{y}|\boldsymbol{\psi})} d\boldsymbol{\alpha} \\
&= \frac{\int f(\boldsymbol{\alpha}) p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) p(\boldsymbol{\alpha}|\phi_\alpha) d\boldsymbol{\alpha}}{\int p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) p(\boldsymbol{\alpha}|\phi_\alpha) d\boldsymbol{\alpha}} \\
&= \frac{\mathrm{E}_a[f(\boldsymbol{\alpha}) p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) p(\boldsymbol{\alpha}|\phi_\alpha)/p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})]}{\mathrm{E}_a[p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\theta}) p(\boldsymbol{\alpha}|\phi_\alpha)/p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})]}.
\end{aligned} \tag{2.28}$$

We are specifically interested in estimating $\boldsymbol{\alpha}$ and $\exp(\tau\boldsymbol{\alpha}/2)$. Once the estimates of parameters $\hat{\boldsymbol{\psi}}$ are obtained, we sample $\boldsymbol{\alpha}^{(1)}, ..., \boldsymbol{\alpha}^{(N)}$ from $p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \hat{\boldsymbol{\psi}})$ and approximate the conditional expectation in equation (2.28) by Monte Carlo integration.

2.4 Prediction

Given $y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n)$, we want to predict α and y at an unobserved location $\boldsymbol{x}_0$.

2.4.1 Latent process prediction

Let α_0 be the value of the latent process at unobserved location $\boldsymbol{x_0}$. Given ϕ_α, the joint distribution of $(\alpha_0, \boldsymbol{\alpha})$ is multivariate normal. The mean and variance of $p(\alpha_0|\boldsymbol{\alpha}, \phi_\alpha)$ are given by

$$\hat{\alpha}(\boldsymbol{x}_0) = \boldsymbol{r}_\alpha(\boldsymbol{x}_0, \boldsymbol{x}) R_\alpha^{-1} \boldsymbol{\alpha}, \tag{2.29}$$

and

$$\sigma_{\hat{\alpha}}^2(\boldsymbol{x}_0)) = 1 - \boldsymbol{r}_\alpha(\boldsymbol{x}_0, \boldsymbol{x}) R_\alpha^{-1} \boldsymbol{r}_\alpha(\boldsymbol{x}, \boldsymbol{x}_0). \tag{2.30}$$

where $\boldsymbol{r}_\alpha = [r(\boldsymbol{x}_0, \boldsymbol{x}_1), ..., r(\boldsymbol{x}_0, \boldsymbol{x}_n)]^T$ is a $n \times 1$ covariance vector between $\alpha(\boldsymbol{x}_0)$ and $\alpha(\boldsymbol{x}_1)$, ..., $\alpha(\boldsymbol{x}_n)$, and R_α is the $n \times n$ covariance matrix for $\boldsymbol{\alpha}$ at observed locations.

Given the model parameters, we are seeking the best predictor of α_0, that is, $E[\alpha_0|\boldsymbol{y},\boldsymbol{\psi}]$. Note that

$$\begin{aligned} E[\alpha_0|\boldsymbol{y},\boldsymbol{\psi}] &= E\left\{E(\alpha_0|\boldsymbol{\psi},\boldsymbol{\alpha},\boldsymbol{y})|\boldsymbol{\psi},\boldsymbol{y}\right\} \\ &= E\left\{E(\alpha_0|\boldsymbol{\psi},\boldsymbol{\alpha})|\boldsymbol{\psi},\boldsymbol{y}\right\} \\ &= E\left\{r_\alpha(\boldsymbol{x}_0,\boldsymbol{x})R_\alpha^{-1}\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi}\right\} \\ &= \boldsymbol{r}_\alpha(\boldsymbol{x}_0,\boldsymbol{x})R_\alpha^{-1}E[\boldsymbol{\alpha}|\boldsymbol{y},\boldsymbol{\psi}]. \end{aligned} \tag{2.31}$$

We can get the latent process estimates $\hat{\boldsymbol{\alpha}}$ at observed locations by Monte Carlo integration of (2.28). Plugging the estimates of $\boldsymbol{\alpha}$ and ϕ_α into (2.31), an empirical best predictor of α_0 is given by

$$\hat{\alpha}(\boldsymbol{x}_0) = \hat{\boldsymbol{r}}_\alpha(\boldsymbol{x}_0,\boldsymbol{x})\hat{R}_\alpha^{-1}\hat{\boldsymbol{\alpha}}. \tag{2.32}$$

2.4.2 y process prediction

For observations from a multivariate normal distribution, it is well known that the best predictor is the same as the best linear unbiased predictor. The SHP model is unconditionally non-Gaussian and conditionally heteroskedastic Gaussian. Given the latent process and assuming the parameters in the model are known, the best predictor (BP) and best linear unbiased predictor (BLUP) have different forms in the SHP model.

Empirical Best Predictor (EBP)

Given the latent process α, the joint distribution of $y(\boldsymbol{x}_0)$ and the observation vector $\boldsymbol{y}$ is heteroskedastic Gaussian. The conditional covariances of $(y_0,\boldsymbol{y})$ are given by

$$\begin{aligned} \mathrm{Cov}(y_0,\boldsymbol{y}|\boldsymbol{\psi},\alpha_0,\boldsymbol{\alpha}) &= \sigma^2\exp(\tau\alpha_0/2)r_z(\boldsymbol{x}_0,\boldsymbol{x})\mathrm{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}, \\ \mathrm{Cov}(\boldsymbol{y},\boldsymbol{y}|\boldsymbol{\psi},\boldsymbol{\alpha}) &= \sigma^2\mathrm{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}R_z\mathrm{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}, \\ \mathrm{Var}(y_0|\boldsymbol{\psi},\boldsymbol{\alpha}_0) &= \sigma^2\exp(\tau\alpha_0). \end{aligned} \tag{2.33}$$

It follows that the mean and variance of the conditional distribution $p(y_0|\boldsymbol{y}, \boldsymbol{\psi}, \alpha_0, \boldsymbol{\alpha})$ are

$$\begin{aligned} & E(y_0|\boldsymbol{y}, \boldsymbol{\psi}, \boldsymbol{\alpha}, \alpha_0) \\ = \; & g(\boldsymbol{x}_0)^T\boldsymbol{\beta} + \exp(\tau\alpha_0/2)\boldsymbol{r}_z(\boldsymbol{x}_0, \boldsymbol{x})R_z^{-1}\mathrm{diag}\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\}(\boldsymbol{y} - G^T\boldsymbol{\beta}), \end{aligned} \tag{2.34}$$

$$\mathrm{Var}(y_0|\boldsymbol{y}, \boldsymbol{\psi}, \alpha_0) = \sigma^2\exp(\tau\alpha_0)(1 - \boldsymbol{r}_z(\boldsymbol{x}_0, \boldsymbol{x})R_z^{-1}\boldsymbol{r}_z(\boldsymbol{x}, \boldsymbol{x}_0)). \tag{2.35}$$

To remove the conditioning on $\boldsymbol{\alpha}$ and α_0, we integrate out the latent process from (2.34) and (2.35). Therefore, we are seeking the best predictor $E(y_0|\boldsymbol{y}, \boldsymbol{\psi})$ and its predictive variance $\mathrm{Var}(y_0|\boldsymbol{y}, \boldsymbol{\psi})$. We then compute

$$\begin{aligned} E(y_0|\boldsymbol{y}, \boldsymbol{\psi}) &= E\left[E(y_0|\boldsymbol{y}, \boldsymbol{\psi}, \boldsymbol{\alpha}, \alpha_0)|\boldsymbol{y}, \boldsymbol{\psi}\right] \\ &= E\left\{E\left[E(y_0|\boldsymbol{y}, \boldsymbol{\psi}, \boldsymbol{\alpha}, \alpha_0)|\boldsymbol{y}, \boldsymbol{\alpha}, \boldsymbol{\psi}\right]|\boldsymbol{y}, \boldsymbol{\psi}\right\} \\ &= E\left\{g(\boldsymbol{x}_0)^T\boldsymbol{\beta} + E\left[\exp(\tau\alpha_0/2)|\boldsymbol{y}, \boldsymbol{\alpha}, \boldsymbol{\psi}\right]\boldsymbol{r}_z(\boldsymbol{x}_0, \boldsymbol{x})R_z^{-1}\right. \\ &\quad \left.\times\mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}(\boldsymbol{y} - G^T\boldsymbol{\beta})|\boldsymbol{y}, \boldsymbol{\psi}\right\}. \end{aligned} \tag{2.36}$$

Since the joint distribution of $(\alpha_0, \boldsymbol{\alpha})$ is normal, we get

$$E[\exp(\tau\alpha_0/2)|\boldsymbol{y}, \boldsymbol{\alpha}] = \exp\left(\frac{\tau\mu_{\alpha_0}}{2} + \frac{\tau^2 v_{\alpha_0}}{8}\right), \tag{2.37}$$

where $\mu_{\alpha_0} = \boldsymbol{r}_\alpha(\boldsymbol{x}_0, \boldsymbol{x})R_\alpha^{-1}\boldsymbol{\alpha}$ and $v_{\alpha_0} = 1 - \boldsymbol{r}_\alpha(\boldsymbol{x}_0, \boldsymbol{x})R_\alpha^{-1}\boldsymbol{r}_\alpha(\boldsymbol{x}, \boldsymbol{x}_0)$ are the mean and variance of $p(\alpha_0|\boldsymbol{\alpha}, \phi_\alpha)$. Plugging (2.37) into (2.36), we can get the best predictor of y_0.

For $\mathrm{Var}(y_0|\boldsymbol{y}, \boldsymbol{\psi})$, it is well known that

$$\mathrm{Var}(y_0|\boldsymbol{y}, \boldsymbol{\psi}) = E\{\mathrm{Var}(y_0|\boldsymbol{y}, \alpha_0, \boldsymbol{\psi})|\boldsymbol{y}, \boldsymbol{\psi}\} + \mathrm{Var}\{E(y_0|\boldsymbol{y}, \boldsymbol{\alpha}, \alpha_0, \boldsymbol{\psi})|\boldsymbol{y}, \boldsymbol{\psi}\}, \tag{2.38}$$

where

$$\begin{aligned}
& E\{\mathrm{Var}(y_0|\boldsymbol{y},\alpha_0,\boldsymbol{\psi})|\boldsymbol{y},\boldsymbol{\psi}\} \\
= \; & E\{E[\mathrm{Var}(y_0|\boldsymbol{y},\alpha_0,\psi)|\boldsymbol{y},\boldsymbol{\alpha},\boldsymbol{\psi}]|\boldsymbol{y},\boldsymbol{\psi}\} \\
= \; & \sigma^2 E\left\{E[\exp(\tau\alpha_0)|\boldsymbol{\alpha},\boldsymbol{y},\boldsymbol{\psi}](1-r_z(\boldsymbol{x}_0,\boldsymbol{x})R_z^{-1}r_z(\boldsymbol{x}_0,\boldsymbol{x})^T)|\boldsymbol{y},\boldsymbol{\psi}\right\} \\
= \; & \sigma^2 E\left\{\exp\left(\tau\mu_{\alpha_0}+\frac{\tau^2 v_{\alpha_0}}{2}\right)(1-r_z(\boldsymbol{x}_0,\boldsymbol{x})R_z^{-1}r_z(\boldsymbol{x}_0,\boldsymbol{x})^T)|\boldsymbol{y},\boldsymbol{\psi}\right\} \qquad (2.39)
\end{aligned}$$

and

$$\begin{aligned}
& \mathrm{Var}\{E(y_0|\boldsymbol{y},\boldsymbol{\alpha},\alpha_0,\boldsymbol{\psi})|\boldsymbol{y},\boldsymbol{\psi}\} \\
= \; & E\left\{E\left[E(y_0|\boldsymbol{y},\boldsymbol{\alpha},\alpha_0,\boldsymbol{\psi})-E(y_0|\boldsymbol{y},\psi)^2|\boldsymbol{y},\boldsymbol{\alpha},\boldsymbol{\psi}\right]|\boldsymbol{y},\boldsymbol{\psi}\right\} \\
= \; & E\left\{\exp\left(\tau\mu_{\alpha_0}+\frac{\tau^2 v_{\alpha_0}}{2}\right) r_z(\boldsymbol{x}_0,\boldsymbol{x})R_z^{-1}\mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}(\boldsymbol{y}-G^T\boldsymbol{\beta})\right. \\
& \left.\times(\boldsymbol{y}-G^T\boldsymbol{\beta})^T\mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}R_z^{-1}r_z(\boldsymbol{x}_0,\boldsymbol{x})^T|\boldsymbol{y},\boldsymbol{\psi}\right\} \\
& -(E(y_0|\boldsymbol{y},\boldsymbol{\psi})-g(\boldsymbol{x}_0)\boldsymbol{\beta})^2. \qquad (2.40)
\end{aligned}$$

Since (2.36) and (2.38) are functions of $\boldsymbol{\alpha}$, we can obtain the *empirical best predictor* (EBP) by plugging in estimated model parameters and evaluating the posterior means of the $\boldsymbol{\alpha}$ functions by Monte Carlo approximation through (2.28).

Empirical Best Linear Predictor (EBLUP)

As an alternative to computing the best predictor, it is simpler to compute the best linear unbiased predictor (BLUP). The mean vector and covariance matrix of the unconditional joint distribution of $(y_0, \boldsymbol{y})$ is given by

$$\begin{bmatrix} y_0 \\ \boldsymbol{y} \end{bmatrix} \sim \left(\begin{bmatrix} \boldsymbol{g}(\boldsymbol{x}_0)^T \\ \boldsymbol{G}(\boldsymbol{x})^T \end{bmatrix}\boldsymbol{\beta}, \sigma^2\begin{bmatrix} 1 & \boldsymbol{r}_y(\boldsymbol{x}_0,\boldsymbol{x}) \\ \boldsymbol{r}_y(\boldsymbol{x},\boldsymbol{x}_0) & R_y \end{bmatrix}\right), \qquad (2.41)$$

where $\boldsymbol{r}_y$ and R_y are computed by (2.2).

The BLUP (in fact, the best linear predictor) of y_0 is given by

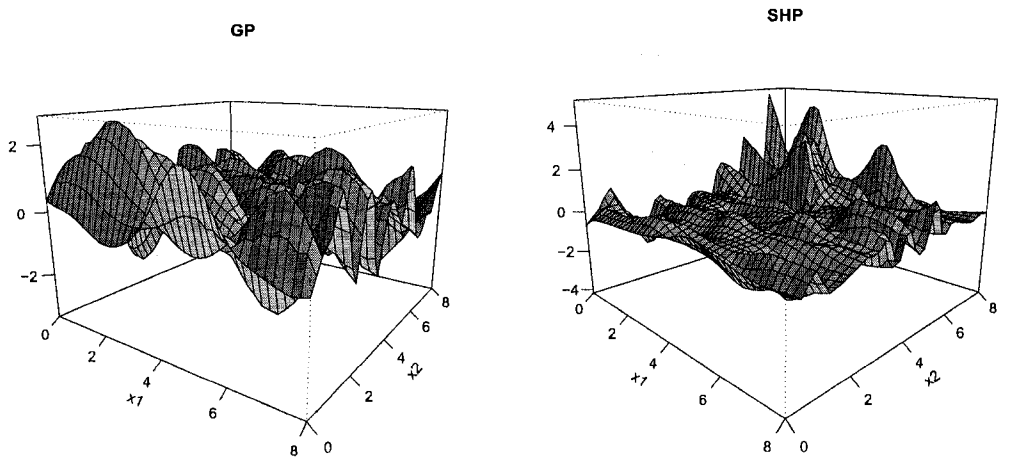
$$\hat{y_0} = g(\boldsymbol{x}_0)^T\boldsymbol{\beta} + \boldsymbol{r}_y(\boldsymbol{x}_0,\boldsymbol{x})R_y^{-1}(\boldsymbol{y}-\boldsymbol{G}(\boldsymbol{x_0})^T\boldsymbol{\beta}), \qquad (2.42)$$

where $\boldsymbol{r}_y = [r_y(\boldsymbol{x}_0, \boldsymbol{x}_1), ..., r_y(\boldsymbol{x}_0, \boldsymbol{x}_n)]^T$ is a $n \times 1$ unconditional correlation vector between $y(\boldsymbol{x}_0)$ and $y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n)$, $G = [\boldsymbol{g}(\boldsymbol{x}_1), ..., \boldsymbol{g}(\boldsymbol{x}_n)]^T$ is the $n \times p$ matrix of regressors for $\boldsymbol{y}$, and R_y is the $n \times n$ unconditional correlation matrix of $\boldsymbol{y}$ at observed locations. The empirical BLUP is calculated by plugging in the estimated $\boldsymbol{\psi}$ into (2.42).

The EBP is able to capture more heteroskedastic features of the process by incorporating the latent process. But the unconditional correlation function of SHP has its unique characteristics and can be used as a new class of correlation functions. We compare the performance of the two predictors in later chapters.

2.5 Extension to Separable SHP Model

In a traditional GP model, different range parameters can be allowed for different input directions by using separable covariance functions. Similarly, we can easily extend the isotropic SHP model into a separable SHP model by using a separable covariance function for the Z process. We keep an isotropic covariance function for the α process so that the correlation of the scale of volatility only depends on the distance between two locations. The separable SHP model is more flexible than the separable GP model by the variety of its sample paths produced. The SHP model not only produces a stationary realization similar to those from separable GP model but also has capacity to produce apparent inhomogeneous realizations. The left panel in Figure 2.11 is a realization from a separable GP with $\phi_1 = 1$ and $\phi_2 = 20$, while the surface in the right panel is a realization from a separable SHP process with the same range parameter $\phi_\alpha = 1$ in the α process and two different range parameters in the Z process: $\phi_{z1} = 1, \phi_{z2} = 20$. We can see clearly the anisotropic behavior of both surfaces. In later sections, we will compare the prediction performance of separable SHP with separable GP through some simulated 2-d realizations.



(a) (b)

Figure 2.11: Separable 2-d surfaces. (a) Separable GP with $\phi_1 = 1$ and $\phi_2 = 20$. (b) Separable SHP with $\tau^2 = 2$, $\phi_\alpha = 1$ and two different range parameters in the Z process: $\phi_{z1} = 1, \phi_{z2} = 20$.

2.6 Simulation Study

In previous sections, we proposed the stochastic heteroskedastic process (SHP) model and discussed the attractive properties of the SHP model. We derived an importance density for likelihood computation and a low-rank kriging approximation for finding the mode $\boldsymbol{\alpha}^*$ in the importance density. In this section, the estimation method is implemented through simulated data from the SHP model. The results of these simulations indicate the confounding effect of the parameters in the SHP model. Root-mean-square error (RMSE) of prediction is used as a criterion to evaluate GP-based and SHP-based predictors on 1-d and 2-d simulated data from both GP and SHP models.

2.6.1 Parameter estimation

We use 1-d simulation from SHP to explore the properties of model parameter estimates for different parameter combinations. The main procedure for parameter estimation involves:

- Obtain importance density $p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$.
  - Assign a set of finite grid points to τ^2, ϕ_α and ϕ_z respectively. Ordinary least squares estimator is used as the initial estimate for $\boldsymbol{\beta}$.
  - The importance density $p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$ is chosen by maximizing the likelihood of the joint log-density $\log p(\boldsymbol{y}, \boldsymbol{\alpha}|\boldsymbol{\psi})$ over a set of points, which are the combinations of all grid points τ^2, ϕ_α and ϕ_z. Meanwhile, the chosen τ^2, ϕ_α and ϕ_z for $p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$ are used as initial estimates for maximum likelihood estimation of model parameters.
  - The importance density $p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$ is fixed for further optimization. For this method, future iteration is not necessary according to our simulation study.
- Draw $\boldsymbol{\alpha}^{(1)}, ..., \boldsymbol{\alpha}^{(N)}$ from $p_a(\boldsymbol{\alpha}|\boldsymbol{y}, \boldsymbol{\psi})$, compute the marginal likelihood $L(\boldsymbol{\psi}|\boldsymbol{y})$ and maximize to get $\hat{\tau^2}, \hat{\phi_\alpha}, \hat{\phi_z}$ and $\hat{\boldsymbol{\beta}}$. The estimate of $\hat{\sigma^2}$ is obtained by (2.27).

We simulate 200 realizations from the SHP model for 6 different parameter combinations listed in Table 2.1. The same values of $\sigma^2 = 0.3$ and $\beta = 0$ are used in all models. The true realizations are based on 80 equally-spaced points on [0, 2]. Gaussian correlation functions are used for both the Z and α processes. The mean and standard deviation of the maximum likelihood parameter estimates are given in Table 2.1.

For models with parameters $\phi_\alpha = 30$ and $\phi_z = 100$, (i.e., the true latent α process is smoother than the Z process), the parameter estimates are quite accurate.

For other models where the latent process is at least as rough as the Z process, the estimates for ϕ_z are positively biased and the estimates for ϕ_α are negatively biased. This is because of the dominance of ϕ_z in the likelihood, so that the Z process accounts for the major roughness of the y process. In previous sections, we have discussed the possible confounding effect between ϕ_α and ϕ_z from the simulated sample path. For a finite sample size, the likelihood favors the model with a rougher Z process and a smoother latent α process.

A slight negative relationship might exist for estimates of ϕ_α and τ^2, which can be explained by their confounding effects in correlation plots in Chapter 2. An increase in ϕ_α or τ^2 leads to correlation decrease. But the major confounding comes from ϕ_α and ϕ_z. For such a flexible model with a finite sample size, we cannot expect highly accurate estimates for model parameters based on likelihood estimation. The confounding effect might be able to be reduced in applications if a Bayesian approach were applied with informative priors. Such priors might come from some pilot study or expert opinion. This will be an interesting direction for future research.

The parameter estimation procedure is different from Davis and Rodriguez-Yam (2005) or Durbin and Koopmans (1997). In their approaches, iteration is taken until the parameter estimates converge. In our approach, the procedure stops after one iteration. We do not continue the iteration until the estimates converge. But even with iteration until likelihood value converges, there is not much improvement on parameter estimation. For concerns of computation time, we ignore further iteration steps in this study.

2.6.2 1-d simulation assessment

In our study, we aim at finding a flexible model that can capture more inhomogeneous features of a surface than the traditional GP model. Out of 80 data points, we sample 30 regularly-spaced points as training data and use the remaining

Table 2.1: Mean and standard deviation of parameter estimates for 1-d SHP simulation based on 500 realizations. Sample size is $n = 80$.

	σ^2	τ^2	ϕ_α	ϕ_z	β
True	0.3	0.3	30	100	0
Mean	0.28	0.38	27.02	100.87	0.0008
Stdev	0.13	0.33	13.86	9.57	0.17
True	0.3	0.3	100	30	0
Mean	0.25	0.58	66.52	71.22	0.0026
Stdev	0.15	0.38	19.70	13.11	0.25
True	0.3	4	30	100	0
Mean	0.32	3.66	24.83	104.21	-0.003
Stdev	0.62	1.65	9.01	12.95	0.17
True	0.3	4	100	30	0
Mean	0.28	4.23	56.55	120.06	-0.0083
Stdev	0.43	1.84	11.96	17.80	0.21
True	0.3	4	30	30	0
Mean	0.50	3.18	22.73	46.54	-0.0033
Stdev	0.97	1.61	7.62	7.57	0.15
True	0.3	4	100	100	0
Mean	0.29	4.19	58.19	164.46	-0.0017
Stdev	0.44	1.94	15.58	20.54	0.16

50 points as test data. Both GP and SHP models are used to fit the training data and RMSE of predicted values at test locations is computed for each realization. For SHP modeling, we use EBP and EBLUP. The RMSE ratios of GP/SHP(EBP or EBLUEP) are summarized in Table 2.2. The RMSE boxplots for GP modeling and SHP modeling are plotted in Figure 2.12.

For the SHP simulated data, the GP sometimes performs comparable to SHP, but most of the time it is worse than the SHP. When $\tau^2 = 0.3, \phi_\alpha = 100$ and $\phi_z = 30$, the smoothness of realizations from SHP are similar to that from GP with $\phi = 30$, as shown in the top panels in Figure 2.13. When $\tau^2 = 0.3, \phi_\alpha = 30$ and $\phi_z = 100$, the smoothness of realizations from SHP are similar to that from

GP with $\phi = 100$ shown in the bottom panels in Figure 2.13. Since the SHP model produces realizations that have homogeneous features similar to features from a GP, the Gaussian prediction accuracy is close to that of the SHP model for these two cases. See Figure 2.12 (a) and (b) and Table 2.2.

On the other hand, when τ^2 increases, the inhomogeneous appearance of realizations increases (Figure 2.14), and the stationary GP lacks the flexibility to handle the inhomogeneous. Thus, the SHP model performs better in prediction. For all the realizations, the SHP model with EBLUP has similar RMSE values as GP model as shown in Figure 2.12. This assures that the unconditional SHP covariance function can be used as a new class of stationary covariance functions in GP modeling.

To have a fair comparison, we simulate 200 realizations from smooth and rough stationary GP models. We want to evaluate the performance of the SHP modeling and prediction when the true model is a stationary GP. Gaussian correlation functions are used in the simulation of the GP. The true realizations are based on 80 equally-spaced points on [0, 2]. Out of 80 data points, we sample 30 regularly-spaced data points as training data and use the remaining 50 points as test data. Both GP and SHP models are used to fit the training data and the RMSE of predicted values at test locations is computed for each realization. For SHP modeling, we use EBP and EBLUP. The RMSE ratios of GP/SHP(EBP or EBLUP) are summarized in Table 2.3. The RMSE boxplots for GP modeling and SHP modeling are plotted in Figure 2.15. We can see that SHP model performs almost identically in prediction to stationary GP model. We conclude from the simulation results that SHP performs similar to GP for homoskedastic 1-d functions and outperforms GP model for heteroskedastic 1-d functions.

2.6.3 2-d separable simulation

In this subsection, we want to compare the prediction accuracy for a 2-d separable GP model with a separable SHP model. In a separable model, the surfaces

Table 2.2: Summary of RMSE ratios (GP/SHP) for 200 realizations generated from 1-d SHP model. For all realizations, $\sigma^2 = 0.3$ and $\beta = 0$. The true realizations are based on 80 equally-spaced points on [0,2]. A sample of 30 regularly-spaced points is used as training data. The out-of-sample RMSE is computed for each realization. The 6-number summary statistics of RMSE ratio for GP/SHP are listed. EBP represents the empirical best predictor for SHP and EBLUP represents the empirical best linear predictor for SHP. The last column is the percentage of the 200 realizations preferring SHP model.

τ^2	ϕ_α	ϕ_z		min	25%	median	mean	75%	max	percent
0.3	30	100	GP/EBP	0.43	0.98	1.00	1.00	1.01	1.75	56
			GP/EBLUP	0.74	0.99	1.00	1.00	1.01	1.18	52
0.3	100	30	GP/EBP	0.35	0.90	1.02	1.05	1.17	1.97	57
			GP/EBLUP	0.72	0.98	1.01	1.02	1.07	1.45	60
4	30	100	GP/EBP	0.20	0.91	1.09	1.31	1.42	6.99	65
			GP/EBLUP	0.41	0.98	1.00	1.03	1.05	4.72	53
4	100	30	GP/EBP	0.33	1.00	1.25	1.42	1.72	5.31	74
			GP/EBLUP	0.51	0.97	1.01	1.01	1.04	1.50	57
4	30	30	GP/EBP	0.37	1.22	1.73	2.34	3.07	15.93	87
			GP/EBLUP	0.25	0.95	1.01	1.01	1.09	1.78	55
4	100	100	GP/EBP	0.15	0.89	1.08	1.20	1.37	7.53	62
			GP/EBLUP	0.67	0.97	1.00	1.02	1.02	4.45	50

are allowed to have different smoothness in different input directions. We first generate 200 realizations from separable SHP model with $\sigma^2 = 0.3, \tau^2 = 8, \phi_\alpha = 0.1$, $\phi_{z1} = 0.1, \phi_{z2} = 5$ and $\beta = 0$. Gaussian correlation functions are used for the α and Z processes. The true surface is based on 21×21 grid points on $[0, 8] \times [0, 8]$. Fifty randomly sampled points from the surface are used as training data and the rest as test data. Both GP and SHP models are used to fit the training data and RMSE of predicted values at test locations is computed for each realization. For SHP prediction, we use EBP and EBLUP. The RMSE ratios of GP/SHP(EBP or EBLUEP) are summarized in Table 2.4. The separable SHP model with BP has

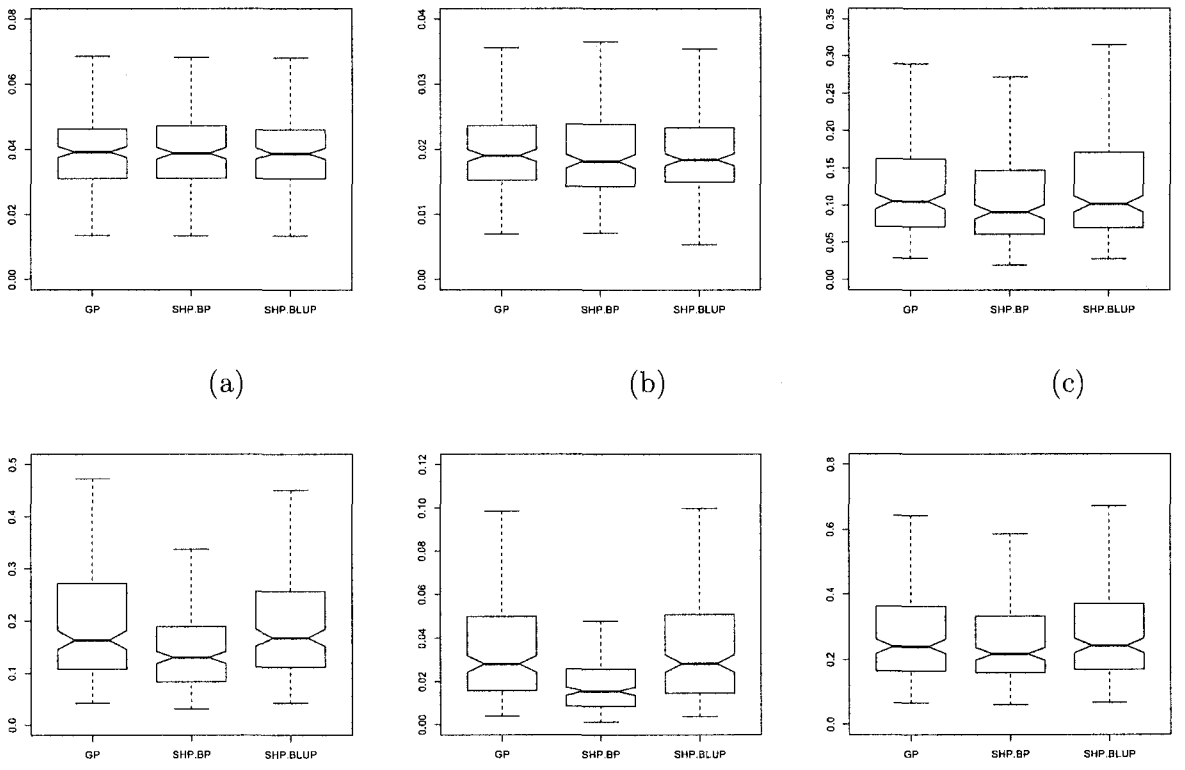


Figure 2.12: RMSE boxplots for GP and SHP modeling and prediction of modeling SHP realizations. (a) $\tau^2 = 0.3$, $\phi_\alpha = 30$ and $\phi_z = 100$. (b) $\tau^2 = 0.3$, $\phi_\alpha = 100$ and $\phi_z = 30$. (c) $\tau^2 = 4$, $\phi_\alpha = 30$ and $\phi_z = 100$. (d) $\tau^2 = 4$, $\phi_\alpha = 100$ and $\phi_z = 30$. (e) $\tau^2 = 4$, $\phi_\alpha = 30$ and $\phi_z = 30$. (f) $\tau^2 = 4$, $\phi_\alpha = 100$ and $\phi_z = 100$. For (a)$--$(f), $\sigma^2 = 0.3$ and $\beta = 0$.

smaller RMSE in 148 out of the 200 trials. The RMSE boxplots for GP and SHP models are plotted in panel (a) of Figure 2.16. We can see that separable SHP model performs better than stationary separable GP model.

We also check the prediction performance of separable GP and separable SHPfor 200 realizations from a 2-dimensional GP, which uses a separable Gaussian correlation function with $\sigma^2 = 0.3, \phi_1 = 0.1, \phi_2 = 5$ and constant mean $\beta = 0$. The RMSE ratios of GP/SHP(EBP or EBLUEP) prediction with GP realizations are summarized in Table 2.5 and the RMSE boxplots are plotted in panel (b) of

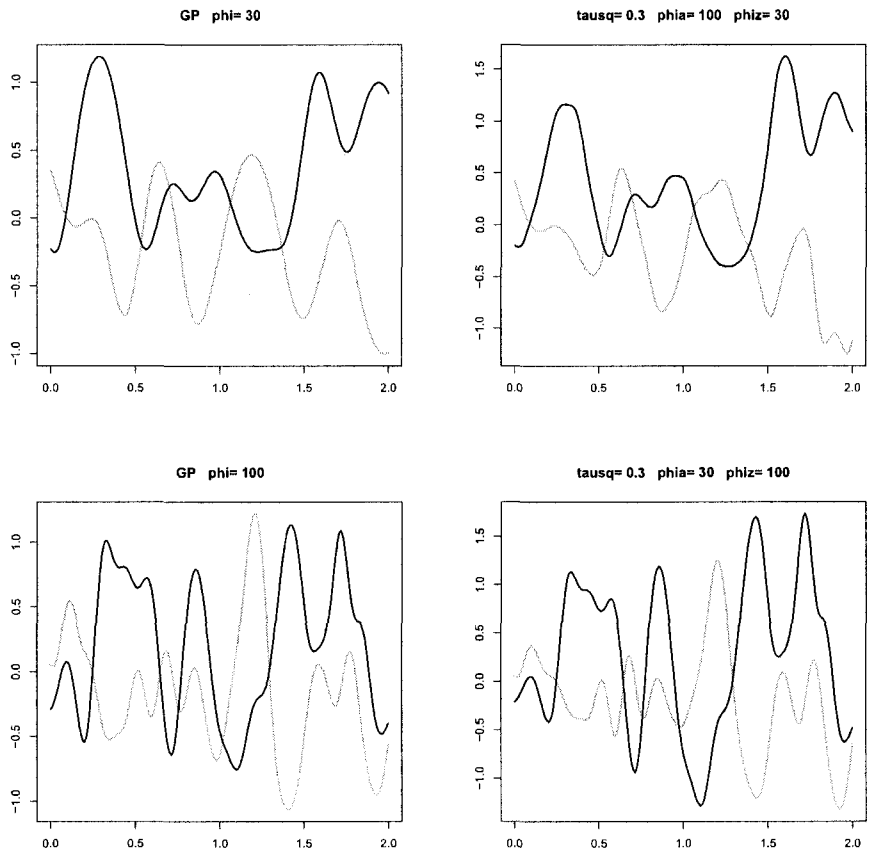


Figure 2.13: Two realizations from 1-d GP and SHP model with different model parameters. The same set of random noise sequences is used from panel to panel in generating the realizations.

Figure 2.16. The separable SHP model performs comparable with the separable GP model, indicating that the separable SHP model can recover the realizations from the separable GP model.

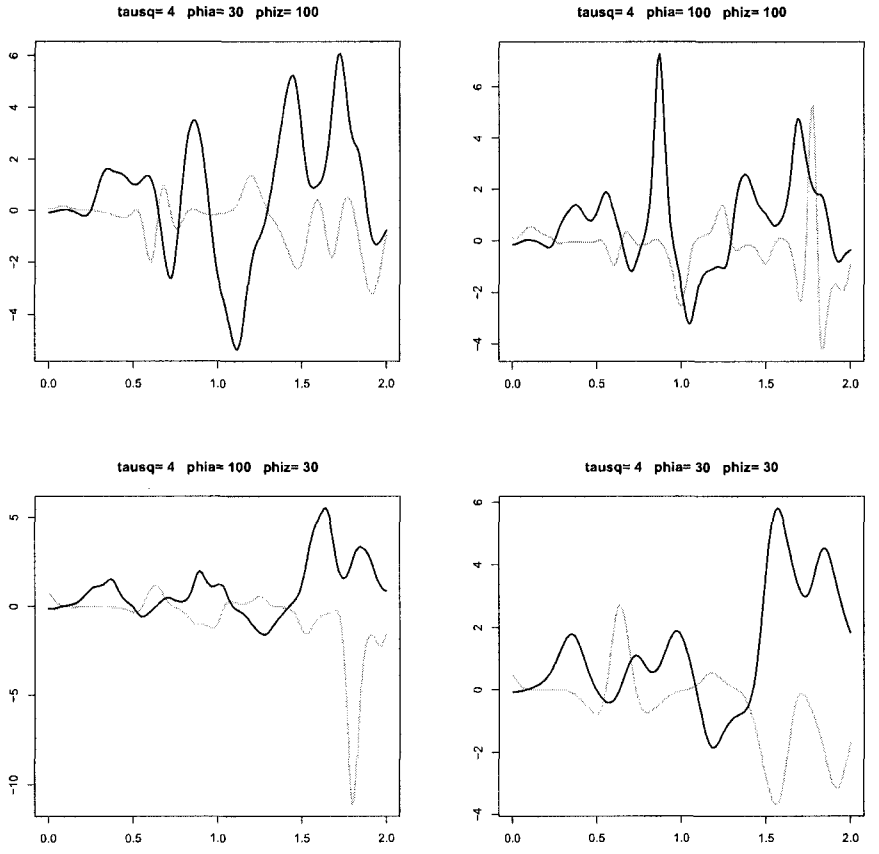
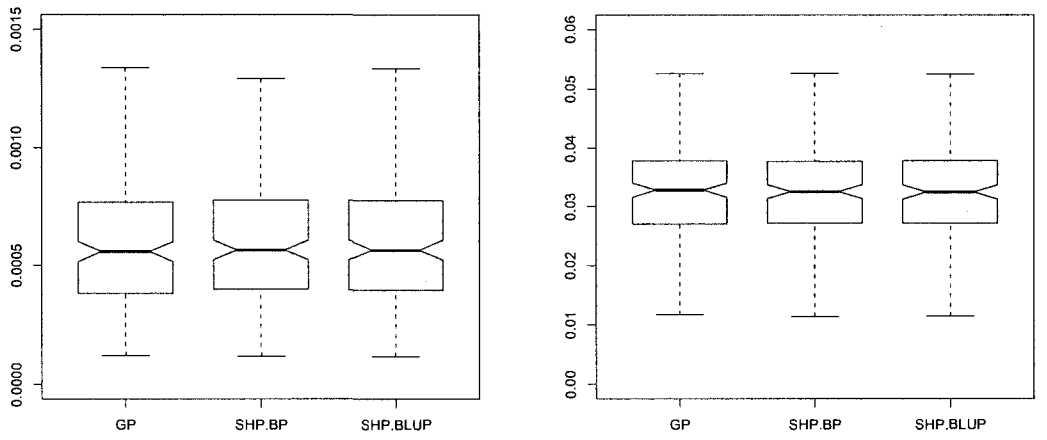


Figure 2.14: Two realizations from 1-d SHP model with different model parameters. The same set of random noise sequences is used from panel to panel in generating the realizations.

Table 2.3: Summary of RMSE ratios (GP/SHP) for 200 realizations generated from 1-d GP model. For all realizations, $\sigma^2 = 0.3$, $\phi = 100$ and $\beta = 0$. The true realizations are based on 80 equally-spaced points on [0,1]. A sample of 30 regularly-spaced points is used as training data. The out-of-sample RMSE is computed for each realization. The 6-number summary statistics of RMSE ration for GP/SHP are listed. EBP represents the empirical best predictor for SHP and EBLUP represents the empirical best linear predictor for SHP. The last column is the percentage of the 200 realizations preferring SHP model.

ϕ		min	25%	median	mean	75%	max	percent
30	GP/EBP	0.61	0.96	1.00	0.99	1.03	1.26	48
	GP/EBLUP	0.36	0.96	1.00	0.99	1.03	1.29	48
100	GP/EBP	0.90	1.00	1.00	1.00	1.00	1.37	49
	GP/EBLUP	0.90	1.00	1.00	1.00	1.00	1.38	43



(a) (b)

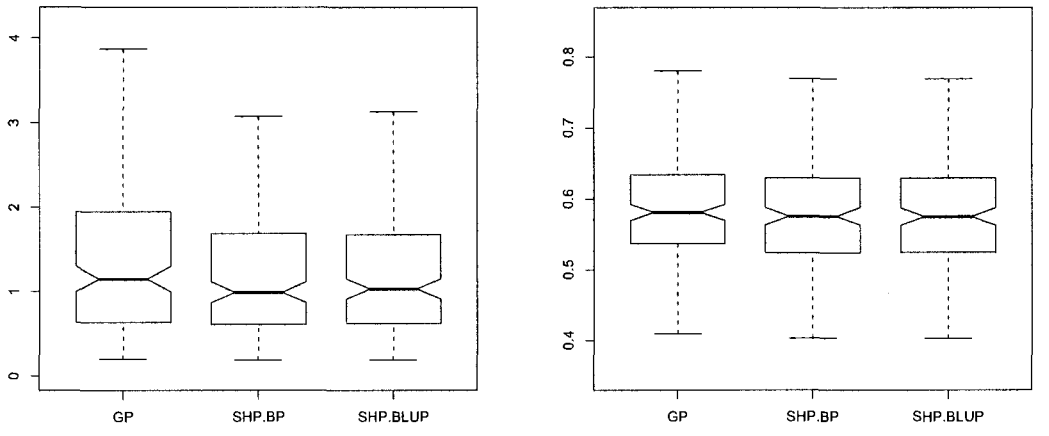
Figure 2.15: RMSE boxplots for GP and SHP modeling and prediction of GP realizations. (a) $\phi = 30$. (b) $\phi = 100$.

Table 2.4: Summary of RMSE ratios (GP/SHP) for 200 realizations generated from 2-d separable SHP model with $\sigma^2 = 0.3$, $\tau^2 = 8$ and $\beta = 0$. The true realization is based on 21×21 grid points on $[0, 8] \times [0, 8]$. Fifty randomly sampled points are used as training data. The out-of-sample RMSE is computed for each realization. The 6-number summary statistics of RMSE ratios for GP/SHP are listed. EBP represents the empirical best predictor for SHP and EBLUP represents the empirical best linear predictor for SHP. The last column is the percentage of the 200 realizations preferring SHP model.

ϕ_α	ϕ_{z1}	ϕ_{z2}		min	25%	median	mean	75%	max	percent
0.1	0.1	5	GP/EBP	0.40	1.00	1.09	1.19	1.23	4.73	74
			GP/EBLUP	0.71	0.99	1.03	1.18	1.13	6.51	70

Table 2.5: Summary of RMSE ratios (GP/SHP) for 200 realizations generated from 2-d anisotropic GP model with $\sigma^2 = 0.3$ and $\beta = 0$. The true realization is based on 21×21 grid points on $[0, 8] \times [0, 8]$. Fifty randomly sampled points are used as training data. The out-of-sample RMSE is computed for each realization. The 6-number summary statistics of RMSE ratios for GP/SHP are listed. EBP represents the empirical best predictor for SHP and EBLUP represents the empirical best linear predictor for SHP. The last column is the percentage of the 200 realizations preferring SHP model.

ϕ_1	ϕ_2		min	25%	median	mean	75%	max	percent
0.1	5	GP/EBP	0.69	0.99	1.00	1.01	1.01	1.40	51
		GP/EBLUP	0.69	0.99	1.00	1.00	1.00	3.17	53



(a) (b)

Figure 2.16: RMSE boxplots for GP and SHP model fitting and prediction of separable GP and SHP realizations. (a) SHP: $\sigma^2 = 0.3$, $\tau^2 = 8$, $\phi_\alpha = 0.1$, $\phi_{z1} = 0.1$, $\phi_{z2} = 5$ and $\beta = 0$. (b) GP: $\sigma^2 = 0.3$, $\phi_1 = 0.1$, $\phi_2 = 5$ and $\beta = 0$.

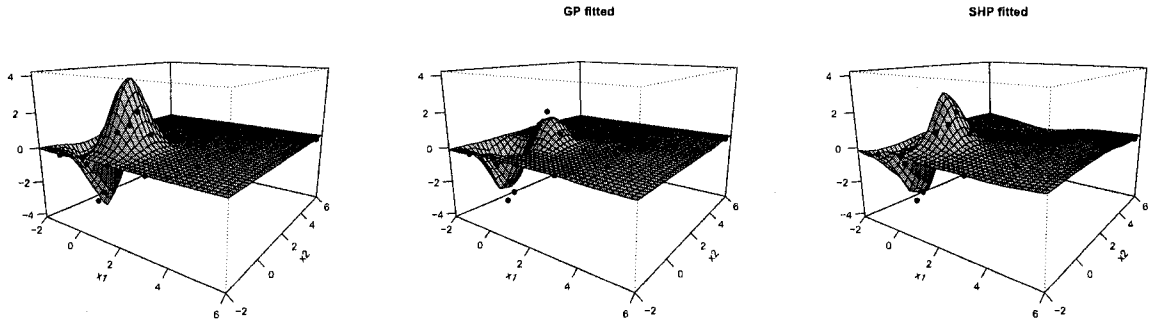
Chapter 3

APPLICATIONS

In this section, we use a two-dimensional mathematical function and higher-dimensional computer experiment examples to compare the prediction accuracy of the SHP model with GP model. RMSE of prediction is used as the criterion to assess the prediction performance.

3.1 Two-Dimensional Test Function

The true function is $f(\boldsymbol{x}) = 10x_1 \exp(-x_1^2 - x_2^2)$, which is ten times the function used in Gramacy et al. (2004) to evaluate the treed GP fitting procedure of that paper. The function is evaluated on a uniform 21×21 grid on $[-2, 6] \times [-2, 6]$. The surface is plotted in Figure 3.1 (a).



(a) (b) (c)

Figure 3.1: (a) True response surface. (b) GP predicted surface based on 20 inputs. (c) SHP predicted surface based on 20 inputs.

To compare the prediction accuracy of SHP with GP, a training data set of size 20 was chosen from the grid points. We used Latin hypercube sampling (LHS) to place 12 points in the quadrant $[-2, 2] \times [-2, 2]$ and 8 points in other areas. The sampling was implemented through the `R` package `tgp` (Gramacy (2007)). After fitting the 20 points with a GP model and a SHP model (using isotropic Gaussian covariance functions for the GP and for α and Z in the SHP), we predicted the other 421 grid points and computed the RMSE. This process of sampling, fitting, and predicting was repeated 100 times. We then computed the 100 RMSE ratios of GP/SHP. The summary statistics of the 100 RMSE ratios are summarized in Table 3.1. The SHP model with BP given by (2.36) has smaller RMSE in 83 of the 100 trials. We give an example of fitted surfaces in Figure 3.1 (b) and (c). The plots show that the SHP model is able to catch peaks better than the GP model. In this example, the latent process $\boldsymbol{\alpha}$ does a good job of capturing the inhomogeneous features of the function.

Table 3.1: Summary statistics for 100 replicates of RMSE ratios for GP/SHP(EBP). The third column indicates the percentage of RMSE ratios being greater than 1 out of 100 replicates.

	MEDIAN	MEAN	PERCENT	AVE(GP RMSE)	AVE(SHP RMSE)
$n = 20$	1.302	1.474	75	0.513	0.418

The GP model is a popular approach for metamodeling in computer experiments not only because it can fit complex functional forms, but also because it provides a measure of prediction uncertainty given by the prediction error variance. The prediction error variance of SHP model can be obtained by (2.38). These prediction error variances can be used in adaptive sampling to select the subsequent sample points. We systematically explore adaptive sampling schemes with SHP in Chapter 4, and compare the efficiency of SHP adaptive sampling to GP adaptive sampling.

3.2 Four-Dimensional Computer Experiment

To illustrate the potential of SHP in metamodeling, we consider an example of a computer simulation experiment given in Qian et al. (2006). The data consist of the outputs from computer simulations for a heat exchanger used in electronic cooling applications. The response y of interest is the steady heat transfer rate depending on four inputs: the mass flow rate of entry air $\dot{m}$, the temperature of entry air T_{in}, the temperature of heat source T_{wall} and solid material thermal conductivity k. There are two types of simulations used in their study: an approximate simulation (AS) for 64 input points and a detailed simulation (DS) for 22 out of these 64 input points. Following their notation, $\boldsymbol{y}_a$ represents the AS outputs and $\boldsymbol{y}_d$ represents the DS outputs.

To explore the relationship between the design factors and heat transfer rate, Qian et al. (2006) proposed a two-step approach to build a surrogate model that can produce predictions close to the DS data. The first step uses 64 AS data to build a base surrogate model and the second step is to adjust the fitted model in step 1 with 22 DS data to create the final surrogate model for DS runs. The two-step models are given by

$$\begin{aligned} y_a(\boldsymbol{x}) &= \beta_{a0} + \sum_{h-1}^{d} \beta_{ah} x_h + \epsilon_a(\boldsymbol{x}), \\ y_d(\boldsymbol{x}) &= \rho(\boldsymbol{x}) y_a(\boldsymbol{x}) + \delta(\boldsymbol{x}), \end{aligned} \tag{3.1}$$

where $\epsilon_a(\boldsymbol{x})$ is a stationary GP with zero mean and separable Gaussian covariance function, $\delta(\boldsymbol{x})$ is a stationary GP with unknown constant mean and separable Gaussian covariance function, and $\rho(\boldsymbol{x}) = \rho_0 + \sum_{j=1}^{d} \rho_j x_j$. For more modeling and engineering details, see Qian et al. (2006).

Qian et al. (2006) use 64 AS data to build their base surrogate model. We use these same 64 AS data to build base surrogate models using isotropic and separable

GP models and the SHP model. We compare the three approaches using leave-one-out cross-validation. The RMSE boxplot for 64 leave-one-out data sets is plotted in Figure 3.2. The cross validation score for SHP model with EBP is 0.311, which is 39% smaller than the isotropic Gaussian model (0.509) and 52% smaller than the separable Gaussian model (0.642). The cross validation score for SHP model with EBLUP is 0.427, which is 16% smaller than the isotropic Gaussian model (0.509) and 33% smaller than the separable Gaussian model (0.642). The mean and and standard deviation of SHP model parameter estimates across the 64 leave-one-out data sets are listed in Table 3.2.

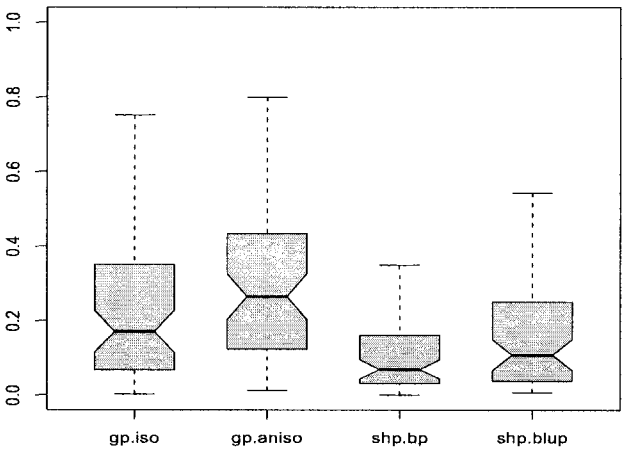


Figure 3.2: RMSE boxplots of different models for leave-one-out cross-validation of 64 AS data $\boldsymbol{y}_a$.

Table 3.2: Mean and standard deviation for parameter estimates of SHP modeling 64 leave-one-out data sets of AS data $\boldsymbol{y}_a$.

	τ^2	ϕ_α	ϕ_z	β_0	β_1	β_2	β_3	β_4
mean	8.772	0.153	0.572	21.169	0.333	-2.304	0.359	5.575
sd	2.219	0.021	0.288	0.075	0.041	0.041	0.050	0.065

Qian et al. (2006) use a separable GP model in their study. For AS data approximation, the separable GP model performs worse than the isotropic GP model in terms of prediction accuracy. This is due to the three additional model parameters in the separable model, resulting in increased variation in estimation and prediction with this small data set.

However, one important advantage of the separable GP model is that it allows different correlation information of different design factors with respect to responses. The choice of model depends on the objective of the study. The goal of this example is to build a good final surrogate model for DS runs. The mathematical model for DS data is different from the mathematical model for AS data. The correlation of each input variable for AS data may not be exactly the same as DS data. In the final surrogate model, the accuracy for AS data is related to the accuracy of DS data. This is why we only fit the isotropic SHP model with AS data. We will use the same model in Qian et al. (2006) for DS data approximation. The bias term ($\boldsymbol{\delta}$) between AS and DS data is modeled as a GP with separable Gaussian covariance function. Note the advantage of isotropic SHP model used in this paper: it offers a sensitivity study on volatile areas, leading to a more effective quantification of prediction uncertainty and sampling strategy like we showed in Section 4.2. Moreover, the isotropic SHP model can be easily extended to a separable structure by using a separable covariance function in the Z process.

The goal in Qian's study is to combine AS information into modeling to improve the prediction accuracy of DS data. Among the 64 AS data, 22 of them have DS simulation results. From equation (3.1), we can see that conditioning on $\boldsymbol{y}_a$, the distribution of $\boldsymbol{y}_d$ is multivariate normal. Once parameter estimates are obtained, Qian et al. (2006) compute the bias term $\delta(\boldsymbol{x})$ at 22 observed DS locations by $\delta(\boldsymbol{x}) = y_d(\boldsymbol{x}) - \hat{\rho}(\boldsymbol{x}) y_a(\boldsymbol{x})$. EBLUPs for $\delta(\boldsymbol{x})$ at test locations are easily obtained. We fit the original 64 AS data with the SHP model and predict 14 set-aside AS data. We then

plug the 14 predicted AS values into the final surrogate model of Qian et al. (2006), given in equation (3.1), to predict 14 DS values. The proposed SHP model provides significant improvement in terms of prediction accuracy for both the 14 AS and 14 DS values. For AS, the RMSE based on SHP is 2.073, which is 20% smaller than the RMSE from Qian's result (2.588). For DS, the RMSE based on SHP-predicted inputs is 3.133, which is 17% smaller than Qian's result (3.795).

Using the SHP model to fit AS data improves the prediction accuracy for DS data. This example illustrates the potential of SHP modeling in application of multi-level computer experiments and model validation. The SHP model cannot only be used to model lower-level outputs but also the bias term among different level outputs or the bias between computer code output and physical data in model validation. We did not use SHP to model the bias term in this example because there are only 22 DS data in a 4-dimensional space. For a very small data set evenly distributed over input space, the GP model can do a comparable job to the SHP model.

Sensitivity analysis

To evaluate the sensitivity of the four inputs on the fitted approximate data, we use the 64 approximated runs as a training set and create a test set of 2000 input values for sensitivity analysis. The extended fast method proposed in Satelli et al. (1999) was used to compute the first-order and total sensitivity indices S_i and T_i for the fitted GP and SHP models. The sensitivity analysis is implemented using the `R` package `sensitivity` (Team (2005)). Table 3.3 lists the first-order sensitivity index and total sensitivity index for each input. The importance of order of inputs agrees for the GP fitted model and SHP fitted model. The most important factors are the two temperatures T_{in} and T_{wall} since the sum of their first-order sensitivity indices is over 90% of the total variation in the response. Even though the main effect of

the mass flow rate of entry air $\dot{m}$ is not significant, as its first-order sensitivity index is less than 1% of the total variation, its interaction with other variables is about 4% of the total variation, indicating that the mass flow rate of entry air $\dot{m}$ may also be an important variable and has a more complex relationship with the response.

Table 3.2 lists the maximum likelihood estimates for model parameters. The values of the estimated coefficients are large for T_{in} and T_{wall}, assuring significant main effects for these two factors. The small values of ϕ_α and ϕ_z indicate large correlation between two nearby points. Since the sensitivity and coefficient values don't indicate the main effect of the mass flow rate of entry air $\dot{m}$ and the solid material thermal conductivity k, we assume the relationship between the response and these two factors is non-linear.

Table 3.3: First-order and total sensitivity indices for predicted AS data of GP and SHP models.

		GP		SHP	
Input	Symbol	S_i	Ti	S_i	Ti
Mass flow rate of entry air	$\dot{m}$	0.00876	0.0375	0.00709	0.0429
Temperature of entry air	T_{in}	0.195	0.229	0.154	0.167
Solid material thermal conductivity	k	0.0147	0.0389	0.000648	0.0156
Temperature of heat source	T_{wall}	0.717	0.720	0.767	0.803

3.3 SIR Model

An SIR (Susceptible-Infected-Resistant) model describes the time dynamics of a contagious disease through a system of ordinary differential equations, with one equation for susceptible individuals, one for infected, and one for resistant. We use one example from Estep and Neckels (2006) to investigate the behavior of an SIR model that allows for birth, death due to natural causes, death due to disease, and the possibility that the offspring of the resistant class may inherit the resistance.

This SIR model is described by

$$\begin{cases} \dot{S} = r_n(1 - \frac{N}{k})(S + I + (1 - p_R)R) - d_n S - r_I SI, \\ \dot{I} = r_I SI - (d_n + d_I)I - a_R I, \\ \dot{R} = p_R r_n(1 - \frac{N}{k})R - d_n R + a_R I, \\ S(0) = S_0, I(0) = I_0, R(0) = R_0, \end{cases}$$

where

- $S(t)$ = the number of susceptible individuals in the population at time t,
- $I(t)$ = the number of infected individuals in the population at time t,
- $R(t)$ = the number of resistant individuals in the population at time t,
- $N = S(t) + I(t) + R(t)$, the population size,
- $\dot{S}, \dot{I}, \dot{R}$ denote time derivatives,
- and $\boldsymbol{x} = (a_R, r_n, k, p_R, d_n, r_I, d_I)'$ are input parameters.

The input space for this SIR model is seven-dimensional with domains for each input parameter specified in Table 3.4. Responses of interest for the SIR could be some functional of S, I, R. We consider three responses: the average number of infected individuals, the average number of susceptible individuals and the average number of resistant individuals over a time interval $[0, T]$ with $T = 10$:

$$q(\boldsymbol{x})_1 = \frac{1}{T}\int_0^T S(s, \boldsymbol{x})ds,$$

$$q(\boldsymbol{x})_2 = \frac{1}{T}\int_0^T I(s, \boldsymbol{x})ds,$$

$$q(\boldsymbol{x})_3 = \frac{1}{T}\int_0^T R(s, \boldsymbol{x})ds.$$

Note that these responses are not expensive to compute, so exact results can be obtained and compared to predictions based on models fitted to samples of inputs. We model the three quantities of interest individually.

Table 3.4: Domains for input parameters of SIR model.

Input	Symbol	Domain
Recovery rate	a_R	[0.1,0.3]
Natural growth rate	r_n	[0.3,1.7]
Carrying capacity	k	[95,105]
Probability of inheriting resistance	p_R	[0.09,0.11]
Natural death rate	d_n	[0.1,0.3]
Contraction rate	r_I	[0.1,0.3]
Death rate from disease	d_I	[0.3,1.7]

Using the generalized Green's function and a variational analysis, Estep and Neckels (2006) compute not only the quantity of interest but also the derivatives at sampled input points. This derivative information is used in Estep and Neckels (2006) to create what they refer to as the "higher-order parameter sampling" method, or HOPS, to approximate the quantity of interest at untried locations. We compare stochastic modeling using GP and SHP to the HOPS method.

We first use Latin hypercube sampling to select 70 data points at random from the input domain, which is standardized to $[0,1]^7$. We then fit using HOPS, GP and SHP and predict values of the quantities of interest at 1000 points uniformly distributed in the input domain. These predictions are compared to the true values of the quantities for those 1000 points. The process of sampling, fitting, and predicting is repeated 100 times. Table 3.5 shows the summary statistics for 100 replicates of RMSE ratios for each response variables.

Table 3.5 shows that the SHP model outperforms the traditional GP model and HOPS globally for each quantity of interest. It is also of interest to investigate the performance of predictors locally, for sub-domains in the parameter space. In the SIR setting, a potential sub-domain of interest might be a "good population" with high recovery rate, high natural growth rate and high probability of inheriting resistance, but low natural death rate, low contraction rate and low death rate from

Table 3.5: Summary statistics of 100 replicates of RMSE ratios for HOPS/SHP and GP/SHP for $q1$, $q2$ and $q3$.

		min	25^{th}	median	mean	75^{th}	max	percentage
$q1$	HOPS/SHP	4.280	6.348	7.160	7.385	8.385	11.540	100
	GP/SHP	0.747	1.072	1.242	1.260	1.426	1.848	89
$q2$	HOPS/SHP	0.985	1.374	1.551	1.556	1.743	2.313	99
	GP/SHP	0.868	1.042	1.118	1.135	1.209	1.504	83
$q3$	HOPS/SHP	1.663	2.444	2.709	2.751	3.088	3.979	100
	GP/SHP	0.665	0.990	1.064	1.081	1.148	1.519	70

disease. Another sub-domain of interest might be a "bad" population with low recovery rate, low natural growth rate and low probability of inheriting resistance, but high natural death rate, high contraction rate and high death rate from disease. Among the 1000 test data values, there are 11 in the good sub-domain and 9 in the bad sub-domain. For the 100 repeated training data sets, the ratios of RMSEs for GP/SHP for each quantity of interest in the good and bad sub-domains are summarized in Table 3.6. In this example, the SHP predictions outperform GP not only globally but also locally.

Table 3.6: Summary statistics of 100 RMSE ratios for GP/SHP in the "good" and "bad" sub-domains for the three quantities of interest.

	min	25^{th}	median	mean	75^{th}	max	percentage
$q1$.good	0.529	0.990	1.241	1.369	1.629	3.604	72
$q1$.bad	0.588	0.967	1.058	1.128	1.261	2.394	63
$q2$.good	0.375	0.937	1.116	1.122	1.303	1.736	67
$q2$.bad	0.521	1.045	1.348	1.392	1.569	4.116	79
$q3$.good	0.385	0.923	1.120	1.178	1.320	2.463	64
$q3$.bad	0.499	0.871	1.188	1.366	1.668	3.663	63

An important problem in science and engineering is the determination of the effect of variation in input parameters on the uncertainty of output. This kind of uncertainty analysis is a major objective in Estep and Neckels (2006). In particular,

Estep and Neckels (2006) use a 64-point HOPS to approximate the distribution of $q2$ when the input parameters are independently distributed as uniform on their respective ranges. The exact cumulative distribution function (cdf) for the quantity of interest was approximated by Estep and Neckels (2006) with a massive Monte-Carlo simulation of 30,000 points. We used LHS to select 64 points from the SIR model (input and output), fitted the 64 data points with HOPS, SHP and GP, and then predicted the 30,000 randomly selected points. Figure 3.3 shows the log-odds ratio of the empirical cdf's for the "exact" distribution (MC30000) and the three approximation methods. Compared to the MC30000 distribution, the two-sample Kolmogorov-Smirnov (K-S) test statistics $\max_{\boldsymbol{x}\in\mathbb{R}^d}|F_q - F_{\hat{q}}|$ for HOPS, GP, and SHP are 0.057, 0.053 and 0.044, respectively. Thus, the SHP model outperforms GP and HOPS, indicating the potential of stochastic modeling in the uncertainty analysis.

Sensitivity analysis on SIR model

The extended fast method proposed in Satelli et al. (1999) was used to compute the first-order and total sensitivity indices S_i and T_i for the three quantities of interest in the SIR model. We use maximin LHS design to generate a data set of size 70 as training data. We fitted the training data with GP and SHP models, predicted the responses at 7000 test locations, and then performed the sensitivity analysis on each quantity of interest. The test data set and sensitivity analysis were implemented using the `R` package `sensitivity` (Team (2005)). The estimated S_i and T_i are listed in Tables 3.7 and 3.9. The sensitivity results from the fitted GP model are very similar to the fitted SHP model. For the average susceptible population $q1$, the important factors are contraction rate r_I and death rate from disease d_I. For the average infected population $q2$, the important factors are recovery rate a_R, natural growth rate r_n and death rate from disease d_I. For the average resistant population $q3$, the important factors are recovery rate a_R, natural growth

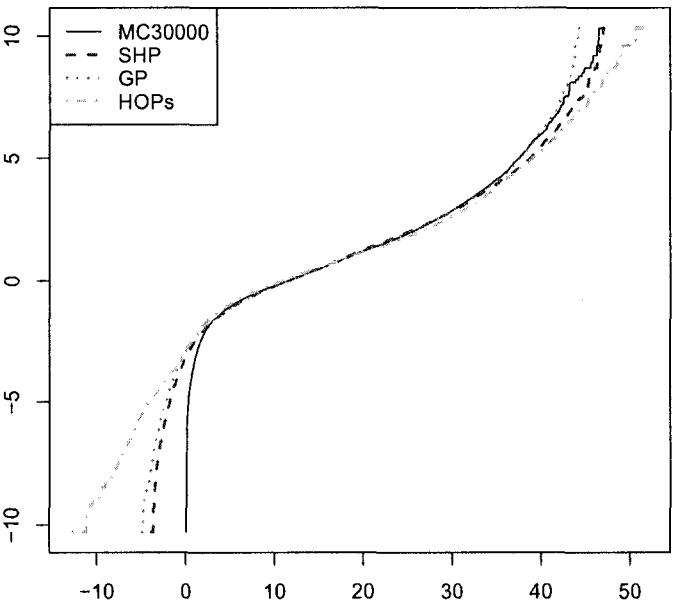


Figure 3.3: Log-odds ratio of empirical cumulative distribution functions, $\log(F_{\hat{q}}/(1-F_{\hat{q}}))$, for "exact" (MC30000) and three approximation methods (HOPS, GP and SHP).

rate r_n, natural death rate d_n and death rate from disease d_I. Though the important factors are different for different quantities of interest, the probability of inheriting resistance p_R and carrying capacity k are not important for any of them. For further study of SIR model with parameter ranges given in Table 3.4, the probability of inheriting resistance p_R and carrying capacity k might be fixed at their reference values. Studies could focus on varying the other five input dimensions.

Table 3.7: First-order and total sensitivity indices for $q1$ by fitting GP and SHP model.

		GP		SHP	
Input	Symbol	S_i	Ti	S_i	Ti
Recovery rate	a_R	0.01	0.014	0.0081	0.015
Natural growth rate	r_n	0.024	0.036	0.0283	0.045
Carrying capacity	k	0.0009	0.011	0.0012	0.011
Probability of inheriting resistance	p_R	0.00007	0.006	0.00005	0.004
Natural death rate	d_n	0.003	0.011	0.0034	0.010
Contraction rate	r_I	0.599	0.635	0.612	0.66
Death rate from disease	d_I	0.296	0.328	0.284	0.319

Table 3.8: First-order and total sensitivity indices for $q2$ by fitting GP and SHP model.

		GP		SHP	
Input	Symbol	S_i	Ti	S_i	Ti
Recovery rate	a_R	0.0024	0.0073	0.003	0.012
Natural growth rate	r_n	0.701	0.740	0.673	0.721
Carrying capacity	k	0.00017	0.0022	0.0007	0.0065
Probability of inheriting resistance	p_R	0.00019	0.0035	0.0001	0.0038
Natural death rate	d_n	0.0198	0.0247	0.0209	0.0275
Contraction rate	r_I	0.0016	0.0048	0.001	0.0047
Death rate from disease	d_I	0.234	0.274	0.241	0.288

Table 3.9: First-order and total sensitivity indices for $q3$ by fitting GP and SHP model.

		GP		SHP	
Input	Symbol	S_i	Ti	S_i	Ti
Recovery rate	a_R	0.096	0.128	0.098	0.132
Natural growth rate	r_n	0.588	0.648	0.589	0.649
Carrying capacity	k	0.0002	0.0032	0.0002	0.0022
Probability of inheriting resistance	p_R	0.0007	0.0036	0.0003	0.0026
Natural death rate	d_n	0.095	0.118	0.0964	0.116
Contraction rate	r_I	0.0006	0.004	0.00052	0.0032
Death rate from disease	d_I	0.158	0.194	0.155	0.192

Chapter 4

ADAPTIVE SAMPLING

Given a set of training data points $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, the GP and SHP models provide the best predictor $\hat{y}(\boldsymbol{x}_0)$ for the response at an untried location $\boldsymbol{x}_0$. Furthermore, the GP and SHP provide the estimate of the predictive variance for $\hat{y}(\boldsymbol{x}_0)$ under the respective models. The predictive variances can be used to quantify the model uncertainty, which in turn can suggest where to sample more points to learn about the model behaviors.

4.1 An Motivating Example

Recall the two-dimensional example used in the last chapter. We have compared the prediction accuracy of SHP and GP based on 100 data sets of size 20. We next want to compare the effectiveness of adaptive sampling when prediction uncertainty is quantified by GP or SHP models. We fit each of the 100 data sets of size 20 with the SHP model and quantify the prediction error variances. Another 20 points are adaptively selected from the grid with probabilities proportional to the SHP model prediction error variances. This sampling without replacement is implemented by the `R` function `sample` (Team (2005)). Once we update the sample size to 40, we fit the 40 points with SHP and GP models, compute the RMSE at the remaining 401 grid points for each model, and select an additional 20 points with probabilities proportional to the updated SHP prediction error variances. We fit the 60 points with the SHP and GP again and compute their RMSE at the remaining 381 grid points. These results are displayed in rows SHP40 and SHP60 of Table 4.1. The

whole process was repeated using GP model prediction error variances in increasing the sample from 20 to 40 and then to 60. These results are displayed in rows GP40 and GP60 of Table 4.1.

Regardless of the way the points are adaptively sampled, SHP dominates GP by producing smaller RMSEs. Further, the SHP does an excellent job of guiding the selection of new points in adaptive sampling. SHP adaptive sampling produces smaller RMSEs, regardless of whether predictions are computed from GP or SHP. The paired boxplots of RMSEs in Figure 4.2 give a graphical, side-by-side comparison of the performance of GP and SHP for prediction and adaptive sampling.

Table 4.1: Summary statistics for 100 replicates of RMSE ratios for GP/SHP(EBP) with different sample sizes and sampling strategies. The third column indicates the percentage of RMSE ratios being greater than 1 out of 100 replicates.

	MEDIAN	MEAN	PERCENT	AVE(GP RMSE)	AVE(SHP RMSE)
tgp20[1]	1.302	1.474	75	0.513	0.418
GP40	1.889	2.133	83	0.371	0.221
SHP40	1.414	1.566	66	0.165	0.114
GP60	1.767	2.272	80	0.274	0.172
SHP60	1.08	1.477	57	0.075	0.060

We give an example of fitted surfaces in Figure 4.1: (a) is GP-predicted surface based on 20 inputs, (b) is GP-predicted surface based on 40 inputs with the extra 20 points selected via GP adaptive sampling, (c) is SHP-predicted surface based on 40 inputs in (b), (d) is SHP-predicted surface based on 20 inputs in (a), (e) is

[1]tgp20 refers to GP modeling (GP20) versus SHP modeling (SHP20) using 20 initial data points sampled through R package tgp (Gramacy (2007)). GP40 refers to RMSE ratio of GP to SHP with 40 data points where the extra 20 points are adaptively sampled using GP20 prediction variance, and SHP40 with the extra 20 points adaptively sampled using SHP20 prediction variance. GP60 refers to RMSE ratio of GP to SHP with 60 data points where the extra 20 points are adaptively sampled using GP40 prediction variance, and SHP60 with the extra 20 points adaptively sampled using SHP40 prediction variance. We also predict the data by SHP model with EBLUP. The prediction behavior is not much different from GP model.

GP-predicted surface based on 40 inputs with the extra 20 points selected via SHP adaptive sampling, (f) is SHP-predicted surface based on 40 in (e). The plots show that the SHP model is able to catch peaks better than the GP model. Furthermore, the GP fitted surface shows considerable improvement with SHP adaptive sampling. In this example, the latent process α does a good job of capturing the inhomogeneous features of the function.

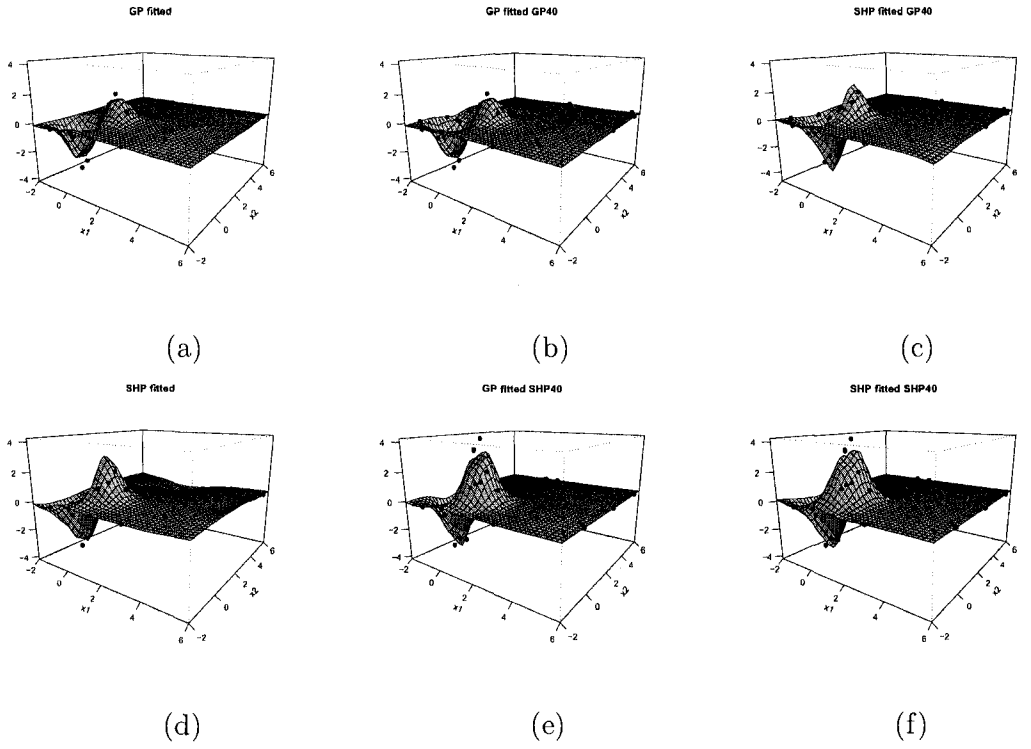


Figure 4.1: (a) is GP-predicted surface based on 20 inputs, (b) is GP-predicted surface based on 40 inputs with the extra 20 points selected via GP adaptive sampling, (c) is SHP-predicted surface based on 40 inputs in (b), (d) is SHP-predicted surface based on 20 inputs in (a), (e) is GP-predicted surface based on 40 inputs with the extra 20 points selected via SHP adaptive sampling, (f) is SHP-predicted surface based on 40 in (e).

The plots in Figure 4.3 and Figure 4.4 illustrate how GP and SHP, respectively, generate adaptive samples. Figure 4.3(a) is the image plot of true absolute errors

$|\boldsymbol{y} - \hat{\boldsymbol{y}}|$ for the GP model based on one set of 20 inputs, and Figure 4.3(b) is the image plot of the GP prediction error variances, which are fairly uniform away from initial sample locations. Accordingly, GP adaptive sampling selects 20 new inputs in a fairly uniform way across the previously unsampled part of the input space. This pattern is repeated in the second row of plots, Figures 4.3(c) and (d), as the sample is extended from 40 to 60 via GP adaptive sampling.

Similarly, Figure 4.4(a) is the image plot of true absolute errors $|\boldsymbol{y} - \hat{\boldsymbol{y}}|$ for the SHP model based on one set of 20 inputs, and Figure 4.4(b) is the image plot of the SHP prediction error variances. In contrast with Figure 4.3(b), these prediction error variances are far from uniform away from initial sample locations, and instead have hot spots of high uncertainty. Accordingly, SHP adaptive sampling selects 20 new inputs intensively in the hot spots. This pattern is repeated in the second row of plots, Figures 4.4(c) and (d), as the sample is extended from 40 to 60 via SHP adaptive sampling. In this example, with clear inhomogeneity in the surface, the SHP not only produces better predictors but also provides a much better adaptive sampling scheme.

The above example shows the potential of using SHP predictive variances in an adaptive sampling scheme, which motivates us to explore it in a more systematic way. In this study, we restrict the problem of adaptive sampling as following: given an initial set of training data points $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$, where $\boldsymbol{x}_i \in \boldsymbol{X}$, a model is selected to describe the relationship between input $\boldsymbol{x}$ and output y. The model is able to iteratively select a new input $\tilde{\boldsymbol{x}}$, observe the corresponding output $\tilde{y}$, and incorporate the new example $(\tilde{\boldsymbol{x}}, \tilde{y})$ into its training set.

The adaptive sampling involves several choices. The choice of initial set of sample points is one important decision. This can be regarded as the first-stage data from which we obtain initial information on the entire response surface. Since the maximin distance LHS designs, discussed in Section 1.1, have a "space-filling"

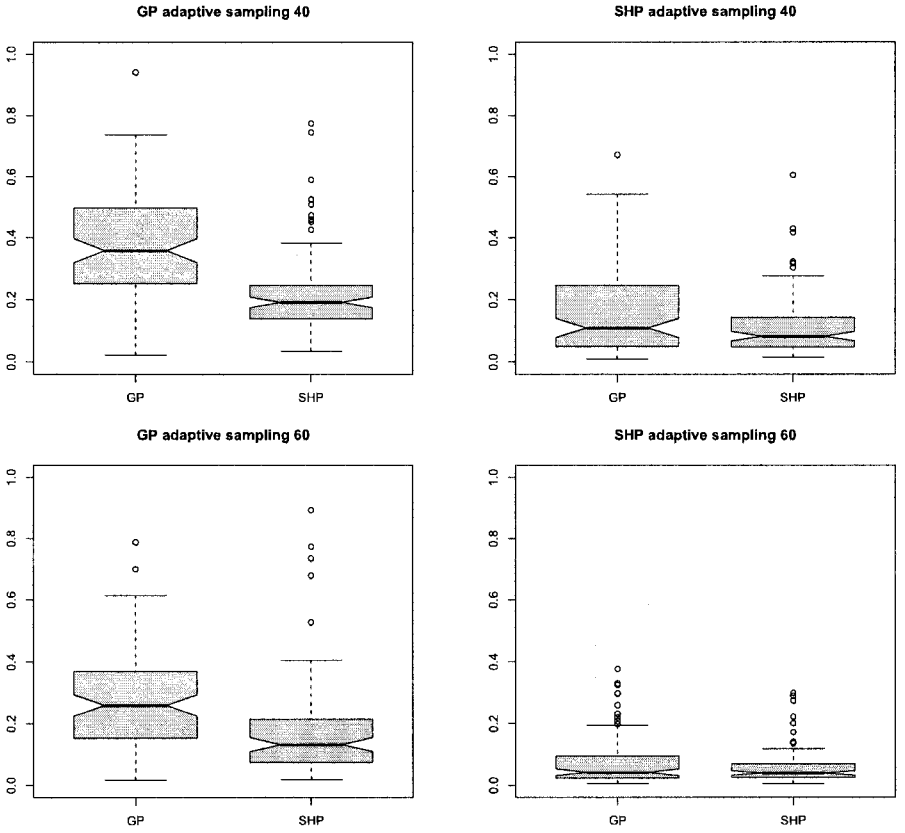


Figure 4.2: Boxplots of 100 RMSEs for GP and SHP model with different sample sizes and sampling strategies.

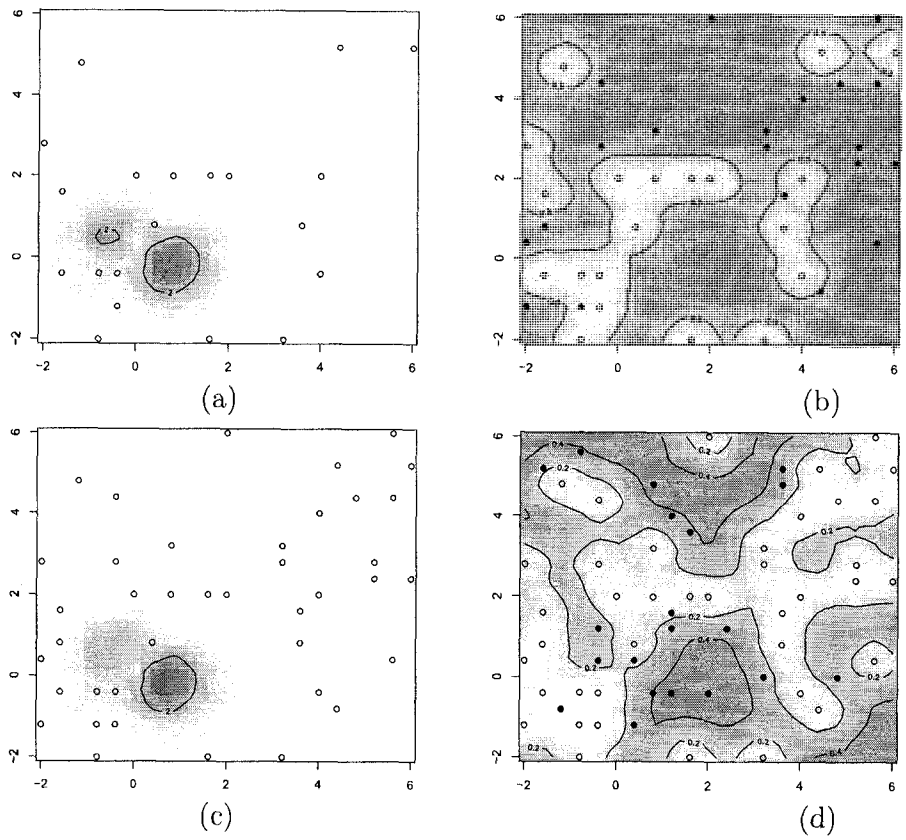


Figure 4.3: (a) Image plot of absolute error $|\boldsymbol{y} - \hat{\boldsymbol{y}}|$ from the GP model using 20 initial sample points given by the open circles. (b) Image plot of GP prediction error variance based on the 20 initial locations (open circles). Solid dots are the 20 locations adaptively sampled using GP. (c) Image plot of absolute error from GP model based on 40 points in (b). (d) Image plot of GP prediction error variance based on the 40 input locations (open circles) in (c). Solid dots are the next 20 locations adaptively sampled using GP.

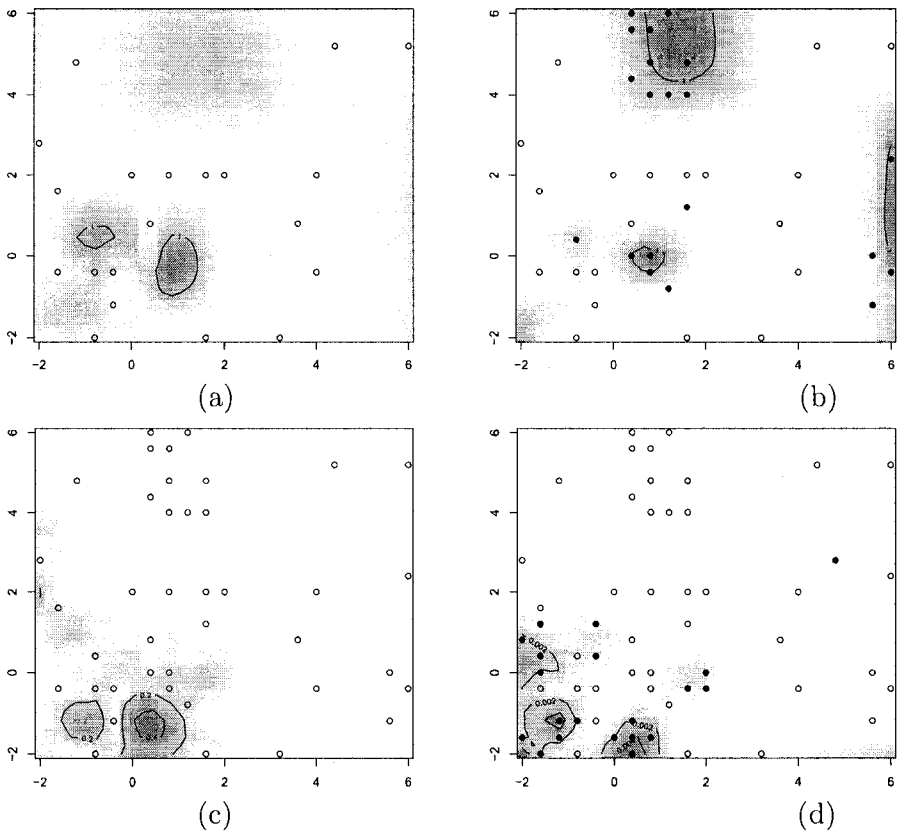


Figure 4.4: (a) Image plot of absolute error $|\boldsymbol{y} - \hat{\boldsymbol{y}}|$ from the SHP model using 20 initial sample points given by the open circles. (b) Image plot of SHP prediction error variance based on the 20 initial locations (open circles). Solid dots are the 20 locations adaptively sampled using SHP. (c) Image plot of absolute error from SHP model based on 40 points in (b). (d) Image plot of SHP prediction error variance based on the 40 input locations (open circles) in (c). Solid dots are the next 20 locations adaptively sampled using SHP.

property in higher-dimensional space and are easy to implement, they are used to obtain the initial data sets in this thesis.

Another important choice is a set of candidate points from which to select the additional sample points. One major goal of computer experiments is to build the functional relationship between the response of a computer code and input variables. The candidate points need to have a good coverage of the entire space. The candidate set can come from a dense grid, which is not feasible in higher-dimensional input space. Again, maximin distance LHS designs are used to create a set of candidate points in this work. This candidate set can also be used as a test data set to evaluate the model prediction accuracy.

Last but not least, we must choose a strategy or algorithm to guide the decision for adaptive sampling. There are many ways to choose $\tilde{\boldsymbol{x}}$, including choosing locations where there is no data, where the model performs poorly, or where the model is expected to change (Cohn et al. (1996)). Statistical criteria for optimal design of experiments such as D-optimality can be adopted in the sequential design, but they are usually computational intense due to the involvement of inverses and determinants of large covariance matrices (Gramacy et al. (2004)). In our study, we consider two criteria, ALM and ALC, in active data selection.

4.2 Two Active Learning Algorithms

There are two useful algorithms for active data selection in machine learning. The first one, called ALM for Active Learning Mackay (Mackay (1992)), has been aimed to maximize the expected information gain about the model parameter values $\boldsymbol{\psi}$ when we receive new data at location $\tilde{\boldsymbol{x}}$. Mackay proved that we will learn most about the model by selecting the data location $\tilde{\boldsymbol{x}}$ with largest predictive variance in the input space. One non-ideal property of this criterion is that the error bars are largest at the most extreme points where data have been gathered for most

models. This lead to us to repeatedly sample data at the edges of the input space (Mackay (1992)). But this disadvantage can be reduced by querying data from a set of candidates $\tilde{\boldsymbol{X}}$ in a defined region of interest or spread out over the space. Thus the candidate point $\tilde{\boldsymbol{x}}$ is selected by

$$\tilde{\boldsymbol{x}} = \underset{\boldsymbol{x}\in\tilde{\boldsymbol{X}}}{\operatorname{argmax}}\, \sigma^2_{\hat{y}}(\boldsymbol{x}).$$

The second approach, called ALC for Active Learning Cohn (Cohn (1996)), is to select the location $\tilde{\boldsymbol{x}}$ to minimize the expectation of the mean square error over input space χ. Let $\hat{y}_n(\boldsymbol{\xi})$ be the predicted value of y at a reference point $\boldsymbol{\xi}$ given current data set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ and let $\hat{y}_{n+1}(\boldsymbol{\xi})$ be the predicted value of y at a reference point $\boldsymbol{\xi}$ when a new input $\tilde{\boldsymbol{x}}$ is added to current data set. The mean square error of $\hat{y}_n(\boldsymbol{\xi})$ and $\hat{y}_{n+1}(\boldsymbol{\xi})$ are given by

$$\begin{aligned}
\mathrm{MSE}\{\hat{y}_n(\boldsymbol{\xi})\} &= E\left[(\hat{y}_n(\boldsymbol{\xi}) - y(\boldsymbol{\xi}))^2 | y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n)\right], \\
\mathrm{MSE}\{\hat{y}_{n+1}(\boldsymbol{\xi})\} &= E\left[(\hat{y}_{n+1}(\boldsymbol{\xi}) - y(\boldsymbol{\xi}))^2 | y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n), y(\tilde{\boldsymbol{x}})\right].
\end{aligned}$$

The mean square error can be decomposed into a variance and bias term, then

$$\begin{aligned}
&\mathrm{MSE}\{\hat{y}_n(\boldsymbol{\xi})\} \\
&= E\left[(\hat{y}_n(\boldsymbol{\xi}) - E(\hat{y}_n(\boldsymbol{\xi})))^2 | y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n)\right] + \left[(E(\hat{y}_n(\boldsymbol{\xi})) - y(\boldsymbol{\xi}) | y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n))\right]^2 \\
&= \sigma^2_{\hat{y}_n}(\boldsymbol{\xi}) + \mathrm{bias}(\hat{y}_n(\boldsymbol{\xi}))^2,
\end{aligned}$$

and

$$\mathrm{MSE}\{\hat{y}_{n+1}(\boldsymbol{\xi})\} = \sigma^2_{\hat{y}_{n+1}}(\boldsymbol{\xi}) + \mathrm{bias}(\hat{y}_{n+1}(\boldsymbol{\xi}))^2, \tag{4.1}$$

where $\sigma^2_{\hat{y}_n}(\boldsymbol{\xi})$ is the predictive variance of $\hat{y}_n(\boldsymbol{\xi})$ and $\sigma^2_{\hat{y}_{n+1}}(\boldsymbol{\xi})$ is the predictive variance of $\hat{y}_{n+1}(\boldsymbol{\xi})$. Assuming the bias in the model is small compared to the variance, we neglect the bias term. The expected mean square error is then approximated by the expected predictive variance, i.e.,

$$\begin{aligned}
E[MSE(\hat{y}_n(\boldsymbol{\xi}))] &\approx E[\sigma^2_{\hat{y}_n}(\boldsymbol{\xi})] \\
E[MSE(\hat{y}_{n+1}(\boldsymbol{\xi}))] &\approx E[\sigma^2_{\hat{y}_{n+1}}(\boldsymbol{\xi})].
\end{aligned}$$

The ALC algorithm aims at selecting $\tilde{\boldsymbol{x}}$ to minimize the expected mean square error, which is equivalent to maximizing the expected reduction in variance by adding a new input $\tilde{\boldsymbol{x}}$. Let $\Delta\sigma^2_{\xi}(\tilde{\boldsymbol{x}})$ be the reduction in predictive variance at reference location $\boldsymbol{\xi}$ given that location $\tilde{\boldsymbol{x}}$ is added into the data, then the expected reduction in predictive variance can be obtained by averaging over the reduction in predictive variance at other referenced locations:

$$\begin{aligned} \Delta\sigma^2(\tilde{\boldsymbol{x}}) &= E[\Delta\sigma^2_{\xi}(\tilde{\boldsymbol{x}})] \\ &= E[\sigma^2_{\hat{y}_n}(\boldsymbol{\xi}) - \sigma^2_{\hat{y}_{n+1}}(\boldsymbol{\xi})]. \end{aligned} \tag{4.2}$$

Equation (4.2) is a function of the candidate location $\tilde{\boldsymbol{x}}$ and it must integrate $\boldsymbol{\xi}$ over the input domain to compute the integrated average change in variance of the model. In practice, Monte Carlo approximation of this integral is used to compute equation (4.2) at a number of reference points drawn according to the distribution of $\boldsymbol{x}$. For simplicity, the set of reference points can be chosen the same as the candidate set $\tilde{X}$.

4.2.1 Active learning in GP regression

The two active learning criteria have been used in GP regression and treed GP model (see Seo et al. (2000) and Gramacy et al. (2004)). The GP model provides the distribution of a predictor response y at an untried location $\boldsymbol{x}_0$ given a training set. The conditional mean and variance are used as the predictor and predictive variance for $\hat{y}(\boldsymbol{x}_0)$:

$$\begin{aligned} \hat{y}(\boldsymbol{x}_0) &= \boldsymbol{g}(\boldsymbol{x}_0)^T\boldsymbol{\beta} + \boldsymbol{r}(\boldsymbol{x}_0, \boldsymbol{x})R^{-1}(\boldsymbol{y} - G^T\boldsymbol{\beta}), \\ \sigma^2_{\hat{y}}(\boldsymbol{x}_0) &= \sigma^2(1 - r(\boldsymbol{x}_0, \boldsymbol{x})R^{-1}r(\boldsymbol{x}, \boldsymbol{x}_0)). \end{aligned} \tag{4.3}$$

Since the variance in (4.3) can be easily computed, implementation of the ALM is straightforward. One merely selects the input $\tilde{\boldsymbol{x}}$ with greatest predictive variance.

In the ALC algorithm, the predictive variances $\sigma^2_{\hat{y}_n(\xi)}$ and $\sigma^2_{\hat{y}_{n+1}(\xi)}(\tilde{\boldsymbol{x}})$ with a new input $\tilde{\boldsymbol{x}}$ are defined as:

$$\begin{aligned}\sigma^2_{\hat{y}_n(\xi)} &= \sigma^2(1 - r_n^T(\boldsymbol{\xi})R_n^{-1}r_n(\boldsymbol{\xi})),\\ \sigma^2_{\hat{y}_{n+1}(\xi)}(\tilde{\boldsymbol{x}}) &= \sigma^2(1 - r_{n+1}^T(\boldsymbol{\xi})R_{n+1}^{-1}r_{n+1}(\boldsymbol{\xi})),\end{aligned}$$

where $\boldsymbol{r}_n = [\gamma(\boldsymbol{x}_1, \boldsymbol{\xi}), ..., \gamma(\boldsymbol{x}_n, \boldsymbol{\xi})]$ is the correlation between the training data and response at a reference location $\boldsymbol{\xi}$ and $r_{n+1} = [\gamma(\boldsymbol{x}_1, \boldsymbol{\xi}), ..., \gamma(\boldsymbol{x}_n, \boldsymbol{\xi}), \gamma(\tilde{\boldsymbol{x}}, \boldsymbol{\xi})]$, R_n is the correlation matrix of the training data and R_{n+1} is the correlation matrix of training data with new response at location $\tilde{\boldsymbol{x}}$. Since the correlation matrix R_{N+1} can be expressed in terms of R_n, the change of variance $\Delta\sigma^2_{\hat{y}(\xi)}(\tilde{\boldsymbol{x}})$ at a reference point $bm\xi$ if the candidate $\tilde{\boldsymbol{x}}$ is added to the training set can be calculated as (Seo et al. (2000)):

$$\Delta\sigma^2_{\hat{y}(\xi)}(\tilde{\boldsymbol{x}}) = \sigma^2_{\hat{y}_n(\xi)} - \sigma^2_{\hat{y}_{n+1}(\xi)}(\tilde{\boldsymbol{x}}) = \frac{(\boldsymbol{k}_n C_n^{-1}\boldsymbol{m} - \gamma(\tilde{\boldsymbol{x}}, \boldsymbol{\xi}))^2}{(\gamma(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}}) - \boldsymbol{m}^T C_n^{-1}\boldsymbol{m})}, \tag{4.4}$$

where $\boldsymbol{k}_n = [\gamma(\boldsymbol{x}_1, \boldsymbol{\xi}), ..., \gamma(\boldsymbol{x}_n, \boldsymbol{\xi})] \in \mathbb{R}^n$ is the vector of covariances between the training data and a response at reference data point $\boldsymbol{\xi}$, $m = [\gamma(\boldsymbol{x}_1, \tilde{\boldsymbol{x}}), ..., \gamma(\boldsymbol{x}_n, \tilde{\boldsymbol{x}})] \in \mathbb{R}^N$ is the vector of covariances between the training data and a candidate data point at $\tilde{\boldsymbol{x}}$, $\gamma(\tilde{\boldsymbol{x}}, \boldsymbol{\xi})$ is the covariance between the response $y(\tilde{\boldsymbol{x}})$ at a candidate point and the response $y(\boldsymbol{\xi})$ at a reference point, $\gamma(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}})$ is the variance of response $y(\tilde{\boldsymbol{x}})$ at a candidate point, and C_n is the covariance matrix for observed data $\boldsymbol{x}_1, ..., \boldsymbol{x}_n$. The average reduction in variance, given that $\tilde{\boldsymbol{x}}$ is added to the data, is obtained by averaging $\Delta\sigma^2_{\hat{y}(\xi)}(\tilde{\boldsymbol{x}})$ over all other locations in χ:

$$\begin{aligned}\Delta\sigma^2_{\hat{y}} &= \int_{\chi} \Delta\sigma^2_{\hat{y}(\xi)}(\tilde{\boldsymbol{x}})\\ &= \int_{\chi} \frac{(\boldsymbol{k}_n C_n^{-1}\boldsymbol{m} - \gamma(\tilde{\boldsymbol{x}}, \boldsymbol{\xi}))^2}{(\gamma(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}}) - \boldsymbol{m}^T C_n^{-1}\boldsymbol{m})}.\end{aligned}$$

In practice, the expected reduction is averaged over a test data set. The candidate data set $\tilde{\boldsymbol{X}}$ can be used as the test data set.

4.2.2 Active learning in SHP model

As we noted in the previous section, the two active learning criteria ALM and ALC have been used in GP regression and treed GP model (Seo et al. (2000), Gramacy et al. (2004)). Given the latent process α, the SHP is a nonstationary GP with covariance function (2.7). The conditional mean (2.34) and variance (2.35) are used as predictor and predictive variance of y at untried location $\boldsymbol{x}_0$. Hence, the ALM and ALC algorithm used in GP regression (Seo et al. (2000)) can be easily extended to the SHP model if α is known. For the ALM algorithm, we select a $\tilde{\boldsymbol{x}}$ from candidate set $\tilde{\boldsymbol{X}}$ with largest value of predictive variance (2.35).

In a SHP model, given the latent process $\boldsymbol{\alpha}$ and model parameters $\boldsymbol{\psi} = (\sigma^2, \tau^2, \phi_\alpha, \phi_z, \boldsymbol{\beta})$, we have

$$\begin{aligned}
\boldsymbol{k}_n &= \mathrm{Cov}[y(\boldsymbol{\xi}), \boldsymbol{y}|\alpha_\xi, \boldsymbol{\alpha}, \boldsymbol{\psi}] = \sigma^2 \exp(\tau\alpha_\xi/2) r_z(\boldsymbol{\xi}, \boldsymbol{x}) \mathrm{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}, \\
C_n &= \mathrm{Cov}[\boldsymbol{y}, \boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{\psi}] = \sigma^2 \mathrm{diag}\left\{\exp(\tau\alpha/2)\right\} R_z \mathrm{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}, \\
\boldsymbol{m} &= \mathrm{Cov}[y(\tilde{\boldsymbol{x}}), \boldsymbol{y}|\alpha_{\tilde{x}}, \boldsymbol{\alpha}, \boldsymbol{\psi}] = \sigma^2 \exp(\tau\alpha_{\tilde{x}}/2) r_z(\boldsymbol{x}_0, \boldsymbol{x}) \mathrm{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}, \\
\gamma(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}}) &= \mathrm{Cov}[y(\tilde{\boldsymbol{x}}), y(\tilde{\boldsymbol{x}})|\alpha_{\tilde{x}}, \boldsymbol{\psi}] = \sigma^2 \exp(\tau\alpha_{\tilde{x}}), \\
\gamma(\tilde{\boldsymbol{x}}, \boldsymbol{\xi}) &= \mathrm{Cov}[y(\tilde{\boldsymbol{x}}), y(\boldsymbol{\xi})|\alpha_{\tilde{x}}, \alpha_\xi, \boldsymbol{\psi}] = \sigma^2 \exp(\tau\alpha_{\tilde{x}}/2) r_z(\tilde{\boldsymbol{x}}, \xi) \exp(\tau\alpha_\xi/2).
\end{aligned}$$

Substituting the above equation into (4.4), the change of variance at a reference point $\boldsymbol{\xi}$ if $\tilde{\boldsymbol{x}}$ is added into training set for SHP model is

$$\begin{aligned}
\Delta\sigma^2_\xi(\tilde{\boldsymbol{x}}) &= \sigma^2_{\hat{y}_n}(\boldsymbol{\xi}) - \sigma^2_{\hat{y}_{n+1}}(\boldsymbol{\xi}) \\
&= \sigma^2 \exp(\tau\alpha(\boldsymbol{\xi})) \frac{(\boldsymbol{r}_z(\boldsymbol{\xi}) R_z^{-1} \boldsymbol{r}_z(\tilde{\boldsymbol{x}})^T - r_z(\tilde{\boldsymbol{x}}, \boldsymbol{\xi}))^2}{(1 - \boldsymbol{r}_z(\tilde{\boldsymbol{x}}) R_z^{-1} \boldsymbol{r}_z(\tilde{\boldsymbol{x}})^T)},
\end{aligned} \tag{4.5}$$

where $\boldsymbol{r}_z(\boldsymbol{\xi}) = [r_z(\boldsymbol{x}_1, \boldsymbol{\xi}), ..., r_z(\boldsymbol{x}_n, \boldsymbol{\xi})] \in \mathbb{R}^n$ is the vector of covariances between the Z process at training locations and the Z value at a reference data points $\boldsymbol{\xi}$, $\boldsymbol{r}_z(\tilde{\boldsymbol{x}}) = [r_z(\boldsymbol{x}_1, \tilde{\boldsymbol{x}}), ..., r_z(\boldsymbol{x}_n, \tilde{\boldsymbol{x}})] \in \mathbb{R}^n$ is the vector of covariances between the Z process at the training locations and the Z value at a candidate data points $\tilde{\boldsymbol{x}}$,

$r_z(\tilde{\boldsymbol{x}}, \boldsymbol{\xi})$ is the covariance of the Z process at a candidate point $\tilde{\boldsymbol{x}}$ and a reference point $\boldsymbol{\xi}$, $r_z(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}})$ is the variance of the Z process at a candidate point $\tilde{\boldsymbol{x}}$, and R_z is the covariance matrix for the Z process at $\boldsymbol{x}_1, ..., \boldsymbol{x}_n$. The ALC value for SHP can be obtained by averaging $\Delta\sigma^2_{\xi}(\tilde{\boldsymbol{x}})$ over a reference data set.

To illustrate the potential of SHP model in active data selection, a realization from SHP with known parameters and latent process is generated. The true realization, as shown in Figure 4.5, is simulated based on 500 equally-spaced points on $[0, 3]$. We randomly select 200 points out of 500 and use them as candidate data, $\tilde{\boldsymbol{X}}$. Another set of 300 randomly selected points out of 500 is used as reference data set in ALC algorithm. The test data set contains 400 randomly selected points out of 500 and is used to compare the performance of ALM, ALC with random selection.

To initiate either algorithms, the first point was chosen randomly from the candidate set $\tilde{\boldsymbol{X}}$. Given the known model parameters and latent process α, the conditional SHP predictive variances are calculated by at the rest of 199 candidate locations. The next sampling point is selected either with the largest predictive variance (ALM) or with the largest reduction of average variance over the reference locations (ALC). We continue sampling through ALM or ALC until a sample size of 50 is achieved.

The entire process above was repeated 20 times with different initial sample locations, candidate data set and test data set. The two sampling strategies, ALM and ALC, are compared by the average (across 20 replicates) of the average (across locations) of the predictive variance, and the average (across the 20 replicates) of the RMSE for the test data set. The adaptive sampling methods are also compared with random selection of the sample points from candidate locations. Figure 4.6 shows the relative efficiency of adaptive sampling to random sampling. The top panel is the average RMSE ratio for ALM/Random and ALC/Random at each sample size. The bottom panel is the ratio for the average of mean predictive variances over

the test locations at each sample size. The ALC, ALM and random selection have the same performance at the first stage since they start with the same initial point. The adaptive sampling methods obviously outperform the random sampling method when sample size increases. The efficiency of the adaptive sampling methods keep increasing until a large sample size is achieved. The ALC method performs similar to ALM in this example.

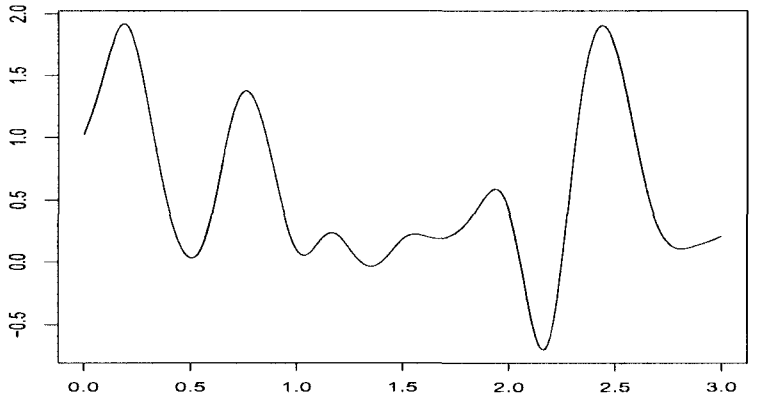


Figure 4.5: A realization from SHP with $\beta = 0, \sigma^2 = 0.1, \tau^2 = 1, \phi_\alpha = 10, \phi_z = 5$.

To see how the adaptive sampling outperforms random sampling, Figure 4.7 — Figure 4.12 give one example of the first 18 points sampled by different methods. In each subplot, the open circles are the current-stage data set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ and the solid dot is $(\boldsymbol{x}_{n+1}, y_{n+1})$, which is going to be sampled next. The simple random sampling method, also known as pseudo-Monte Carlo sampling (Metropolis and Ulam (1949)), is very easy to implement but exhibits some clustering and fails to explore a large proportion of the input domain. In Figure 4.7 — Figure 4.12, if dominant features of the model are likely to be in one part of the space as in other parts, a design with points that are evenly spread out is preferable. The ALM and ALC agree on the interesting area of large volatility most of the time. But sampling differences exists on where exactly to be sampled next. For ALM, the sample data

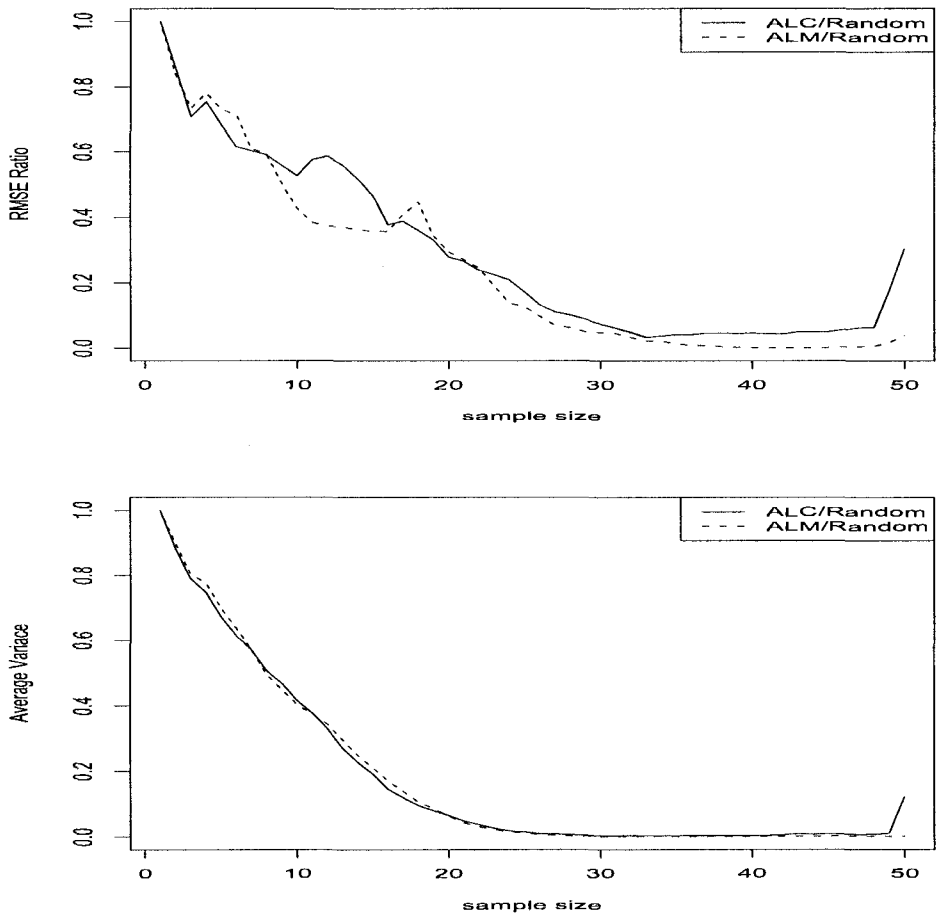


Figure 4.6: Plots for relative efficiency of adaptive sampling with respect to random sampling. Top panel: The ratio of average RMSE over 20 replicates as a function of sample size. Bottom panel: The ratio of the average of the mean predictive variance as a function of sample size.

point is placed at the largest volatility. In this example, since the large volatility area is closer to the boundary, for a small sample size, the ALM is more likely to place the sample points close to the boundary while the ALC has a better coverage over the interior region of the input space.

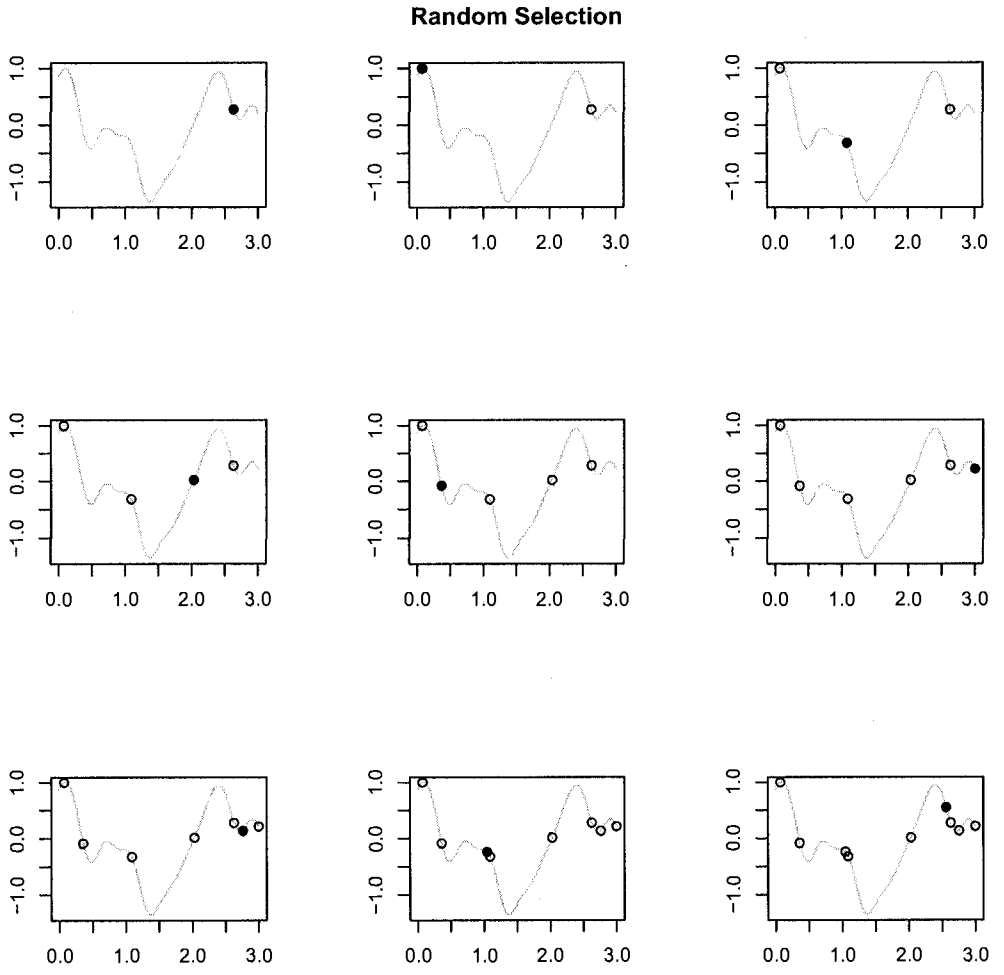


Figure 4.7: The 1^{st} to 9^{th} randomly selected points. The gray line is the true latent curve. In each subplot, the open cycles are the current-stage data set, and the solid dot is the latent process of response to be sampled next.

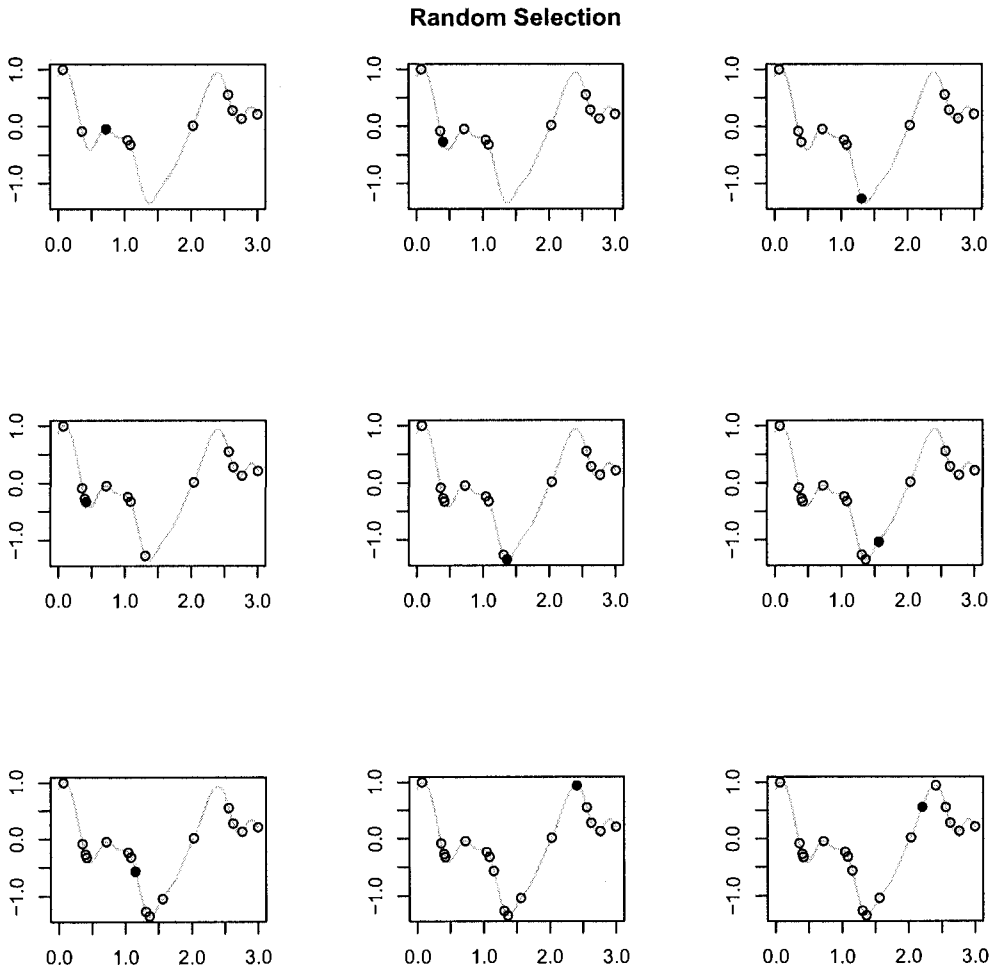


Figure 4.8: The 10^{th} to 18^{th} randomly selected points. The gray line is the true latent curve. In each subplot, the open cycles are the current-stage data set, and the solid dot is the latent process of response to be sampled next.

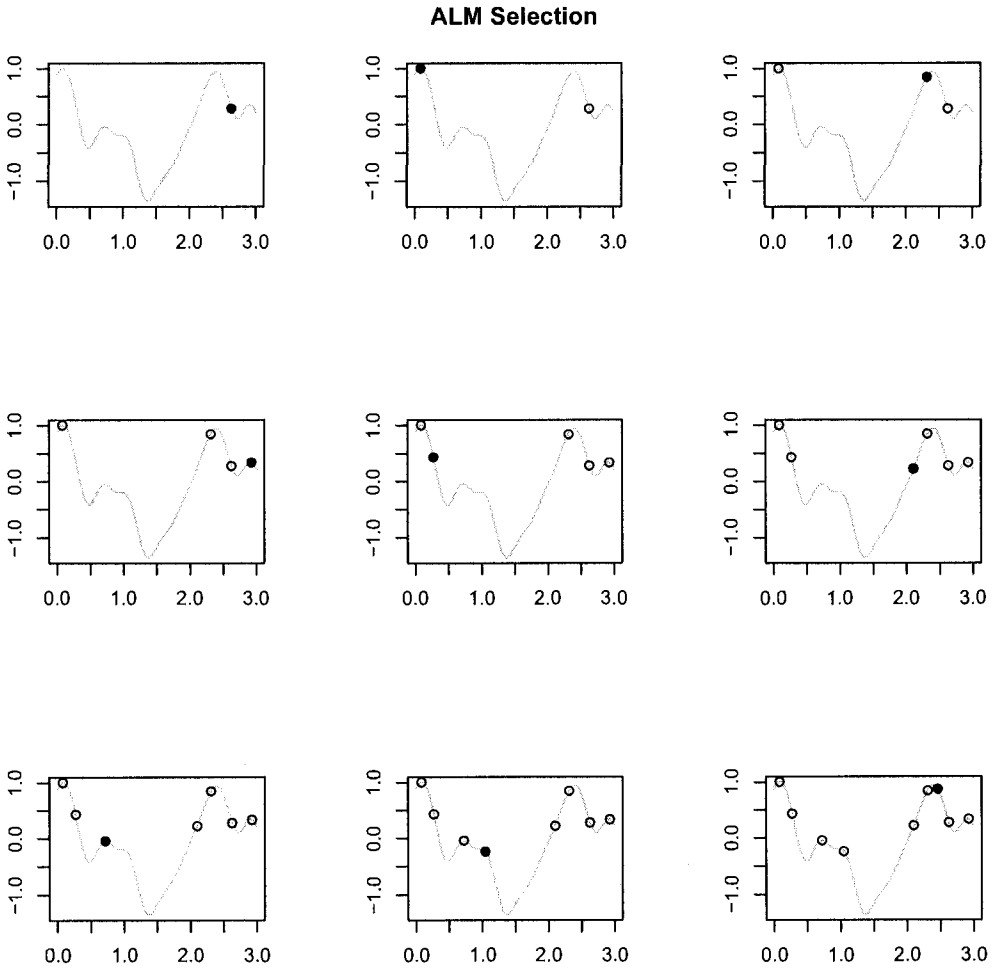


Figure 4.9: The 1^{st} to 9^{th} ALM selected points. The gray line is the true latent curve. In each subplot, the open cycles are the current-stage data set, and the solid dot is the latent process of response to be sampled next.

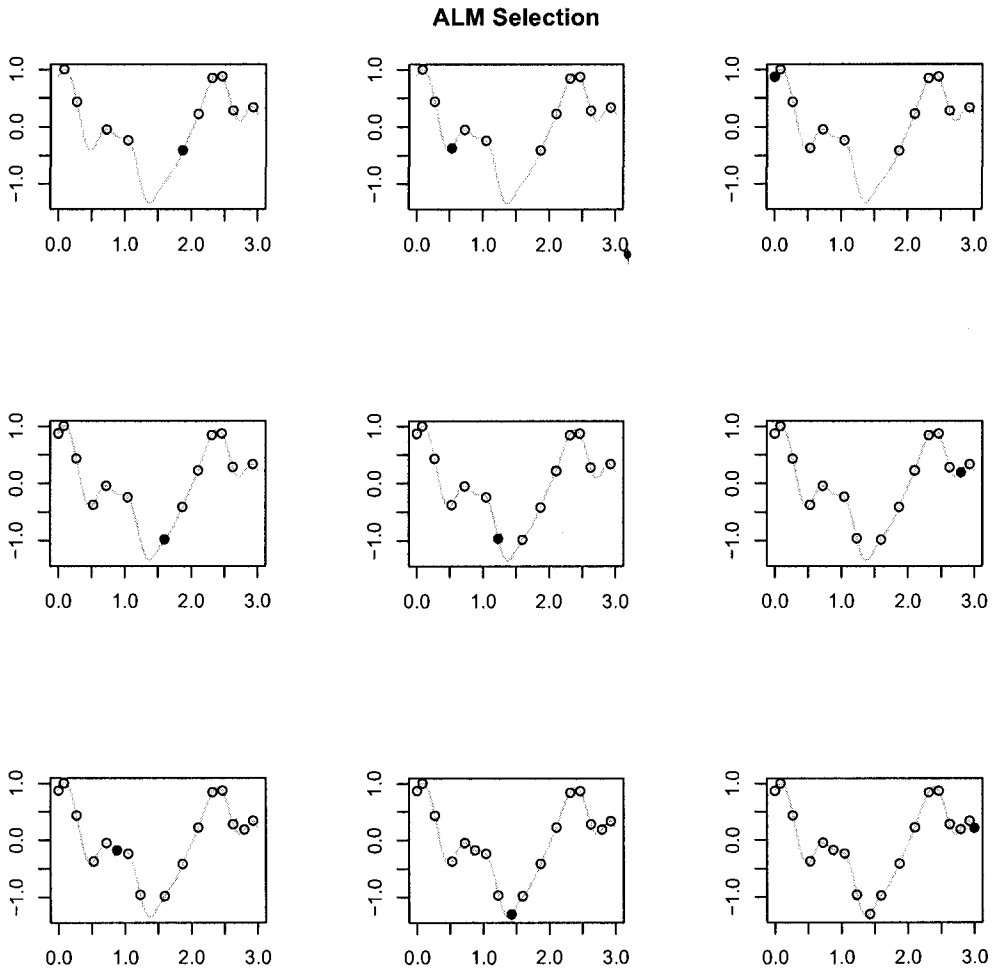


Figure 4.10: The 10^{th} to 18^{th} ALM selected points. The gray line is the true latent curve. In each subplot, the open cycles are the current-stage data set, and the solid dot is the latent process of response to be sampled next.

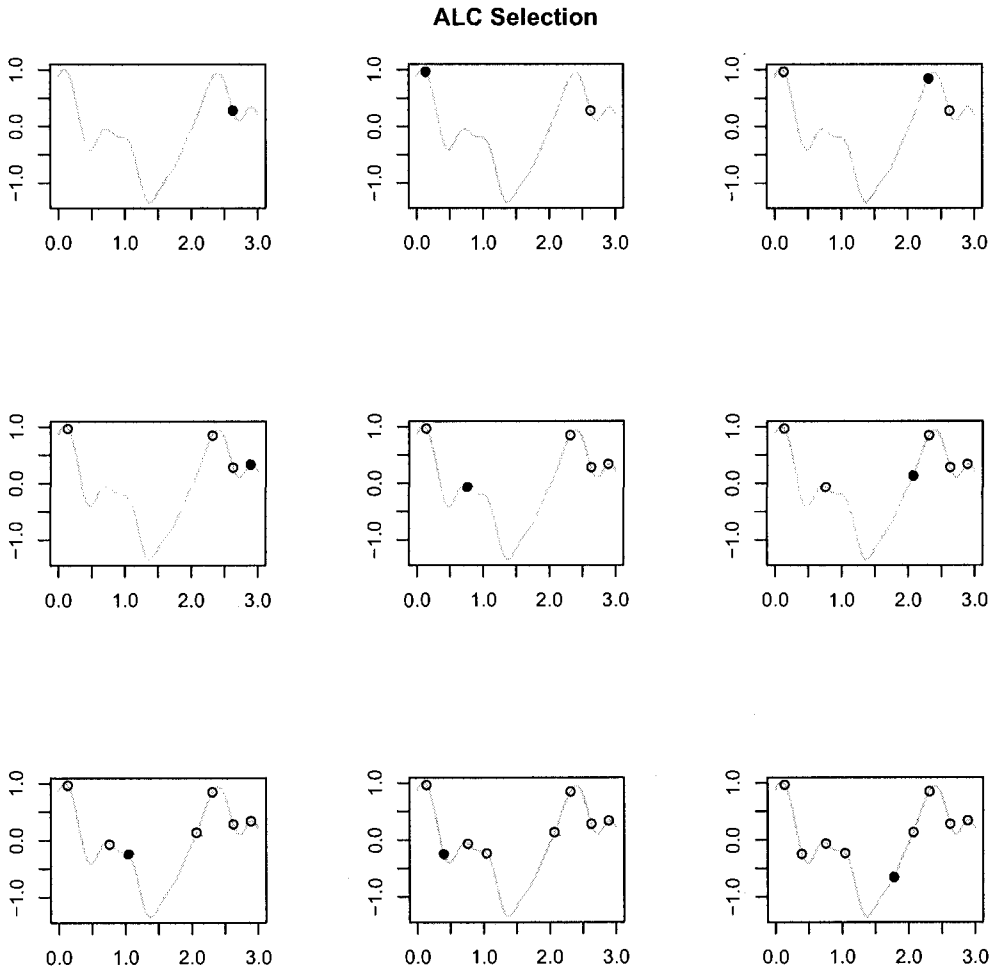


Figure 4.11: The 1^{st} to 9^{th} ALC selected points. The gray line is the true latent curve. In each subplot, the open cycles are the current-stage data set, and the solid dot is the latent process of response to be sampled next.

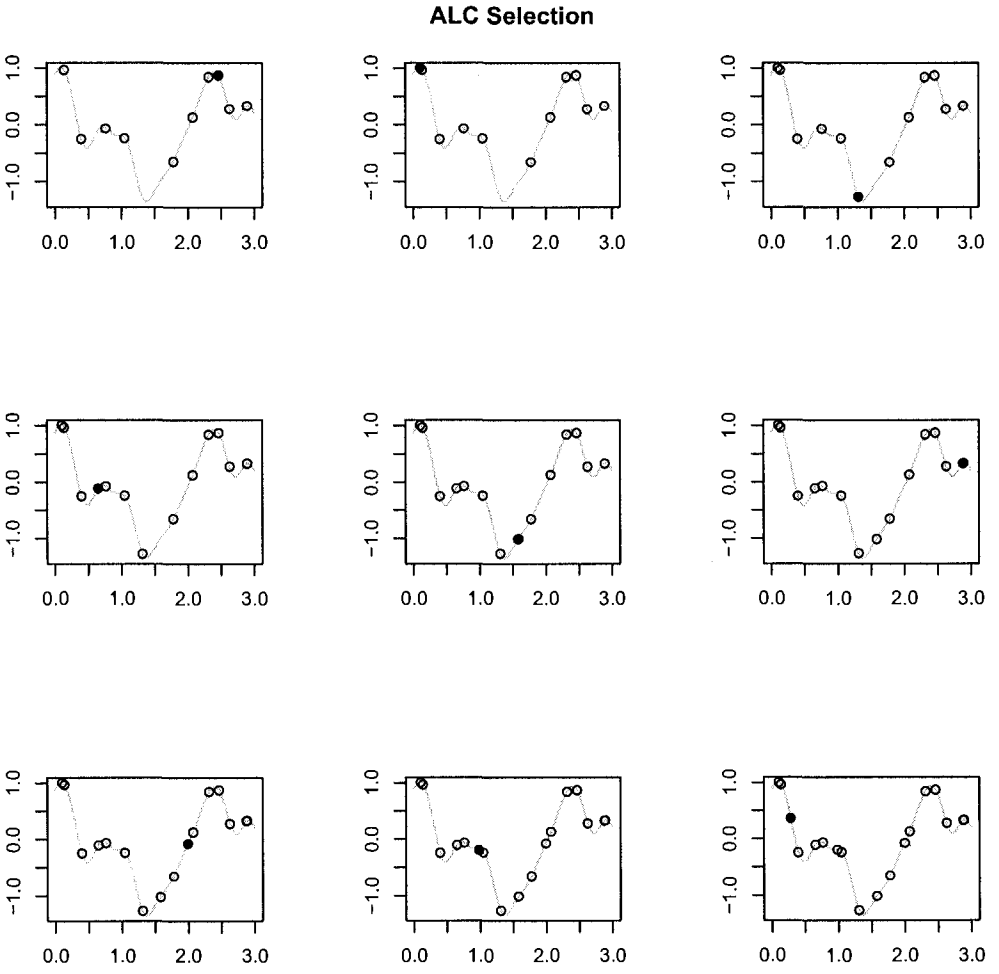


Figure 4.12: The 10^{th} to 18^{th} ALC selected points. The gray line is the true latent curve. In each subplot, the open cycles are the current-stage data set, and the solid dot is the latent process of response to be sampled next.

4.2.3 Adaptive sampling procedure with SHP

In the previous subsection, we illustrate the potential of the SHP model in active data learning using the conditional mean (2.34) and variance (2.35). Unfortunately, the latent process is unknown in reality. We need to integrate out the latent process from (2.34) and (2.35). Therefore, we use the best predictor $E(y_0|\boldsymbol{y},\boldsymbol{\psi})$ with its predictive variance $\mathrm{Var}(y_0|\boldsymbol{y},\boldsymbol{\psi})$ in adaptive sampling. Recall that the best predictor $E(y_0|\boldsymbol{y},\boldsymbol{\psi})$ is given by

$$\begin{aligned} E(y_0|\boldsymbol{y},\boldsymbol{\psi}) &= E\left[E(y_0|\boldsymbol{y},\boldsymbol{\psi},\boldsymbol{\alpha},\alpha_0)|\boldsymbol{y},\boldsymbol{\psi}\right] \\ &= E\left\{g(\boldsymbol{x}_0)^T\boldsymbol{\beta} + E\left[\exp(\tau\alpha_0/2)|\boldsymbol{y},\boldsymbol{\alpha}\,\boldsymbol{\psi}\right]\boldsymbol{r}_z(\boldsymbol{x}_0,\boldsymbol{x})R_z^{-1}\right. \\ &\quad \left.\times\mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}(\boldsymbol{y}-G^T\boldsymbol{\beta})|\boldsymbol{y},\boldsymbol{\psi}\right\}, \end{aligned} \tag{4.6}$$

where $\mu_{\alpha_0} = \boldsymbol{r}_\alpha(\boldsymbol{x}_0,\boldsymbol{x})R_\alpha^{-1}\boldsymbol{\alpha}$ and $v_{\alpha_0} = 1-\boldsymbol{r}_\alpha(\boldsymbol{x}_0,\boldsymbol{x})R_\alpha^{-1}\boldsymbol{r}_\alpha(\boldsymbol{x},\boldsymbol{x}_0)$ are the mean and variance of $p(\alpha_0|\boldsymbol{\alpha},\phi_\alpha)$.

The predictive variance is given by

$$\mathrm{Var}(y_0|\boldsymbol{y},\boldsymbol{\psi}) = E\{\mathrm{Var}(y_0|\boldsymbol{y},\alpha_0,\boldsymbol{\psi})|\boldsymbol{y},\boldsymbol{\psi}\} + \mathrm{Var}\{E(y_0|\boldsymbol{y},\boldsymbol{\alpha},\alpha_0,\psi)|\boldsymbol{y},\boldsymbol{\psi}\}, \tag{4.7}$$

where

$$\begin{aligned} &E\left\{\mathrm{Var}(y_0|\boldsymbol{y},\alpha_0,\boldsymbol{\psi})|\boldsymbol{y},\boldsymbol{\psi}\right\} \\ =\ &\sigma^2 E\left\{\exp\left(\tau\mu_{\alpha_0}+\frac{\tau^2 v_{\alpha_0}}{2}\right)(1-r_z(\boldsymbol{x}_0,\boldsymbol{x})R_z^{-1}r_z(\boldsymbol{x}_0,\boldsymbol{x})^T)|\boldsymbol{y},\boldsymbol{\psi}\right\} \end{aligned} \tag{4.8}$$

and

$$\begin{aligned} &\mathrm{Var}\left\{E(y_0|\boldsymbol{y},\boldsymbol{\alpha},\alpha_0)|\boldsymbol{y},\boldsymbol{\psi}\right\} \\ =\ &E\left\{E\left[E(y_0|\boldsymbol{y},\boldsymbol{\alpha},\alpha_0,\boldsymbol{\psi})-E(y_0|\boldsymbol{y},\psi)^2|\boldsymbol{y},\boldsymbol{\alpha},\boldsymbol{\psi}\right]|\boldsymbol{y},\boldsymbol{\psi}\right\} \\ =\ &E\left\{\exp(\tau\mu_{\alpha_0}+\tau^2 v_{\alpha_0}/2)r_z(\boldsymbol{x}_0,\boldsymbol{x})R_z^{-1}\mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}(\boldsymbol{y}-G^T\boldsymbol{\beta})\right. \\ &\left.\times(\boldsymbol{y}-G^T\boldsymbol{\beta})^T\mathrm{diag}\left\{\exp\left(-\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}R_z^{-1}r_z(\boldsymbol{x}_0,\boldsymbol{x})^T|\boldsymbol{y},\boldsymbol{\psi}\right\} \\ &-(E(y_0|\boldsymbol{y},\psi)-g(\boldsymbol{x}_0)\boldsymbol{\beta})^2. \end{aligned} \tag{4.9}$$

Given the maximum likelihood estimates, the best predictive variance (4.7) can be approximated by Monte Carlo integration. Thus, the ALM procedure can be easily implemented and the next candidate point is selected by

$$\tilde{\boldsymbol{x}} = \underset{\boldsymbol{x}\in\tilde{\boldsymbol{x}}}{\operatorname{argmax}}\, \sigma^2_{\hat{y}}(\boldsymbol{x})$$

where $\sigma^2_{\hat{y}}(\boldsymbol{x}) = \mathrm{Var}(y(\boldsymbol{x})|\boldsymbol{y}, \boldsymbol{\psi})$ is the predictive variance of $\hat{y}(\boldsymbol{x})$ given in equation (4.7).

As for ALC procedure, the change of variance at a reference point $\boldsymbol{\xi}$ if $\tilde{\boldsymbol{x}}$ is added into training set for SHP model is

$$\begin{aligned}
\Delta\sigma^2_{\xi}(\tilde{\boldsymbol{x}}) &= \mathrm{Var}(y_{\xi}|\boldsymbol{y}_n, \boldsymbol{\psi}) - \mathrm{Var}(y_{\xi}|\boldsymbol{y}_{n+1}, \boldsymbol{\psi}) \\
&= E\left\{\mathrm{Var}(y_0|\boldsymbol{y}_n, \alpha_0, \boldsymbol{\psi})|\boldsymbol{y}_n, \boldsymbol{\psi}\right\} - E\left\{\mathrm{Var}(y_0|\boldsymbol{y}_{n+1}, \alpha_0, \boldsymbol{\psi})|\boldsymbol{y}_{n+1}, \boldsymbol{\psi}\right\} + \\
&\quad \mathrm{Var}\left\{E(y_0|\boldsymbol{y}_n, \boldsymbol{\alpha}_n, \alpha_0)|\boldsymbol{y}_n, \boldsymbol{\psi}\right\} - \mathrm{Var}\left\{E(y_0|\boldsymbol{y}_{n+1}, \boldsymbol{\alpha}_{n+1}, \alpha_0)|\boldsymbol{y}_{n+1}, \boldsymbol{\psi}\right\}
\end{aligned} \tag{4.10}$$

Unlike the GP, the change of variance in the SHP using best predictive variance involves the response values at all candidate locations. This is impossible in practice. The advantage of ALC is that it provides a better coverage of the interior of the input space while ALM is more likely to place points on the boundary. But ALM can perform similarly to ALC if the candidate data set is chosen uniformly over the domain (Gramacy et al. (2004)). In this study, the candidate set in the example is chosen by maximin distance LHS design which has a good coverage over the input space. Therefore, we only implement ALM strategy in the SHP model and compare its performance with GP ALM and GP ALC.

Because of the computational cost of the ALM with the SHP model, the model is not updated at every increased sample size. Instead, the model is updated each time the sample size equals a multiple of $k > 1$. Thus, the computation time can be reduced by updating the model on at every k^{th} step. Another reason for doing

so is due to the curse of dimensionality. For a small data set in a high-dimensional input space, the data points are far away from each other so there is not enough information for reconstructing the latent process and quantifying the uncertainty in specific regions. A single new data point added by ALM at the current stage will not help much in improving the model in a high-dimensional space. In this study, we choose $k = 10$. The ALM procedure for fitting data with SHP model is summarized as follows:

1. Choose a set of initial sample points and a set of candidate points. The initial sample is used to get preliminary information about the entire response surface. The set of candidate points for further sampling needs to cover the input space in order to have a good exploration over the domain.

2. Fit the current data set $(\boldsymbol{x}_1, y_1), ..., (\boldsymbol{x}_n, y_n)$ with the SHP model, record the parameter estimates $\hat{\boldsymbol{\psi}}$, the importance samples $\boldsymbol{\alpha}^{(1)}, ..., \boldsymbol{\alpha}^{(N)}$ and the corresponding importance weights $p(\boldsymbol{y}|\boldsymbol{\alpha}^{(i)}, \boldsymbol{\theta})p(\boldsymbol{\alpha}^{(i)}|\phi_\alpha)/p_a(\boldsymbol{\alpha}^{(i)}|\boldsymbol{y}, \boldsymbol{\psi})$ for $i = 1, ..., N$.

3. Choose $\boldsymbol{x}_{n+1}$ with the largest predictive variance, i.e.

$$\boldsymbol{x}_{n+1} = \underset{\tilde{\boldsymbol{x}} \in \tilde{X}}{\operatorname{argmax}} \operatorname{Var}(y(\tilde{\boldsymbol{x}})|\boldsymbol{y}, \boldsymbol{\psi}),$$

4. Fix estimated model parameters, augment the latent process from n to $n+1$ for each Monte Carlo sample $\boldsymbol{\alpha}^{(i)}, i = 1, ..., N$. The augmentation is implemented by predicting $\alpha_{n+1}^{(i)}$ at sampled candidate location $\tilde{\boldsymbol{x}}$ given $\boldsymbol{\alpha}^{(i)}$ and $\hat{\boldsymbol{\psi}}$, that is, for each i,

$$\hat{\alpha}_{n+1}^{(i)}(\tilde{\boldsymbol{x}}) = \boldsymbol{r}_n^{(i)}(\tilde{\boldsymbol{x}})^T C_{\alpha_n^{(i)}}^{-1} \hat{\boldsymbol{\alpha}}_n^{(i)}.$$

5. Use the same importance weights for each augmented $\boldsymbol{\alpha}^{(i)}$ and compute the predictive variance at other candidate locations based on the increased sample. Choose $\boldsymbol{x}_{n+2}$ with the largest predictive variance.

6. Repeat steps 4 and 5 until the sample size increases to $n + k$.

7. Update model parameters at sample size $n + k$, repeat step 2, 3, 4 and 5 until the desired sample size is achieved or the prediction accuracy is met.

For fair comparison, the GP model is also updated at every 10^{th} increased sample size.

4.3 Results and Discussion

In this section, adaptive sampling based on ALM criterion is used for previous 2-d test function data with the SHP model. The efficiency is compared with the GP model using ALM and ALC criteria. Finally, the motivating SIR example is revisited with GP and SHP ALM adaptive sampling.

4.3.1 2-d example revisited

In Section 4.1, we have done the "ALM-like" adaptive sampling in which instead of sampling one at a time with largest predictive variance, we sampled 20 at a time using selection probabilities proportional to the predictive variances. Two observations were obtained: one is that the SHP model has a better prediction accuracy than the GP model in terms of smaller RMSE values; the other is that given a latent process α, the SHP model provides a better quantification of model uncertainty, leading to a more efficient sampling scheme. In this subsection, we explore the adaptive sampling of the SHP model using ALM criterion. The results are compared with the adaptive sampling of GP model using both ALM and ALC criteria.

To compare the sampling efficiency of the SHP with the GP, a training data set of size 20 was chosen. We use a D-optimal design to place 12 points in the quadrant $[-2, 2] \times [-2, 2]$ and 8 points in other areas. A candidate date set of size 100 was chosen evenly by D-optimal design over the domain $[-2, 6] \times [-2, 6]$. The sampling

was implemented through the R package tgp (Gramacy (2007)). The 21×21 grid points on $[-2, 6] \times [-2, 6]$ were used as a test data set to evaluate the prediction performance.

After fitting the 20 points with a SHP model, we compute the predictive variances at 100 candidate locations and select the next point with largest predictive variance. We keep increasing the sample size until it achieves 60. For each increased sample size, we fit the data with a GP model and a SHP model, predict the 441 test points and compute the RMSE. This process of ALM adaptive sampling, fitting, and prediction was repeated 20 times. A different random seed is used at each time to generate the initial data set and the candidate data set. We compute the average RMSE over 20 replicates at different sample sizes. The average RMSE curves as a function of sample size for SHP and GP model fitting with SHP ALM adaptively sampled data points are plotted in Figure 4.13.

Similarly, we fit the each of the 20 initial data sets with the GP model and increase the sample size to 60 using ALM and ALC with GP-computed predictive variances. The whole process of fitting and predicting was repeated for 20 replicates. The average RMSE curves as a function of sample size for SHP and GP model fitting with GP ALM and GP ALC adaptively sampled data points are plotted in Figure 4.13.

Figure 4.13 gives a graphical comparison of the performance of GP and SHP for adaptive sampling and prediction. In this figure, RMSE is plotted as a function of sample size for SHP and GP models over 20 replicates. Strategies are denoted by [adaptive sampling method]-[prediction method]. For example, GP(ALM)-GP uses GP ALM adaptive sampled data, and fits a GP model for prediction.

Regardless of the way the points are adaptively sampled, SHP dominates the GP by producing smaller RMSEs. Further, the SHP model does an excellent job of guiding the selection of new points in ALM adaptive sampling. SHP adaptive

sampling produces smaller RMSEs, regardless of whether predictions are computed from GP or SHP.

Figure 4.13 tells us that adaptive sampling with GP model guided by ALC criterion performs slightly differently than when guided by ALM criterion. The average RMSE curve for ALC adaptively sampled points decreases faster than ALM adaptively sampled points at a smaller sample size, i.e., 30, while the average RMSE curve for ALM adaptively sampled points has a faster decrease than ALC adaptively sampled points at a larger sample size, 50. As for SHP model, the RMSE curves with SHP ALM adaptive sampling decreases faster than those with GP adaptive sampling procedures. The rates of decrease of RMSE curve relate to the sampling efficiency. The faster the RMSE curve decreases, the better the adaptive sampling efficiency.

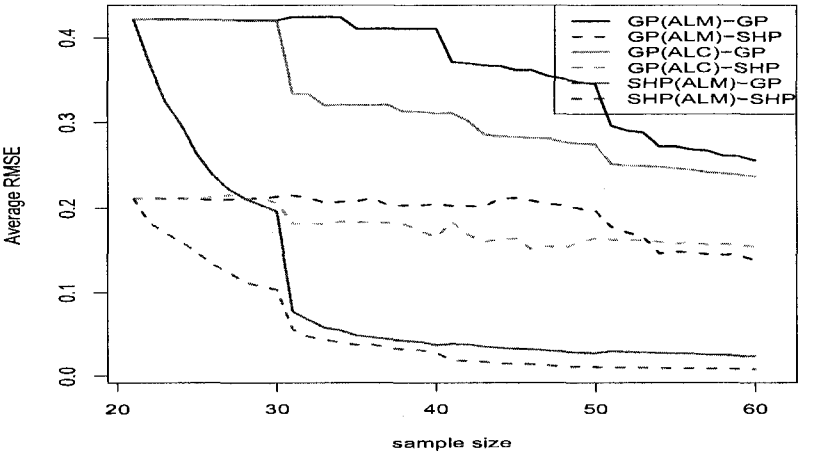


Figure 4.13: RMSE plots as a function of sample size for SHP and GP models over 20 replicates. Strategies are denoted by [adaptive sampling method]-[prediction method]. For example, GP(ALM)-GP uses GP ALM adaptive sampled data, and fits a GP model for prediction.

We give an example of the initial data set, candidate data set and test data set in Figure 4.14. The open squares are the locations for the initial data set. The solid

dots are the 100 candidate locations for further adaptive sampling and the gray dots are the test locations. The data points in the candidate set are evenly spread out over the input space by the property of D-optimal design. Figure 4.15 plots the GP and SHP fitted surfaces for the initial data set. Even though the fitted surfaces capture the major feature of the true function, there is still room for improvement.

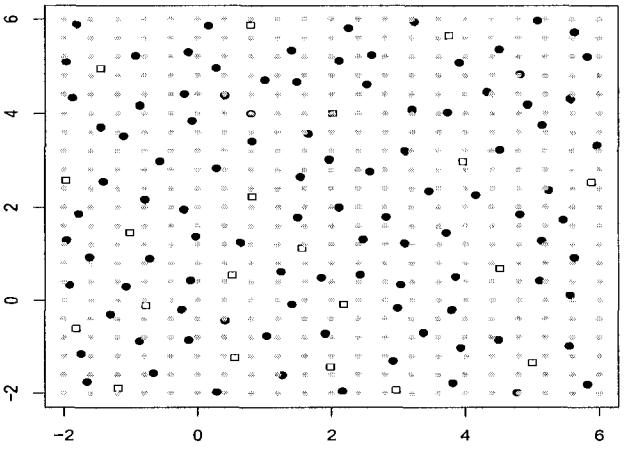
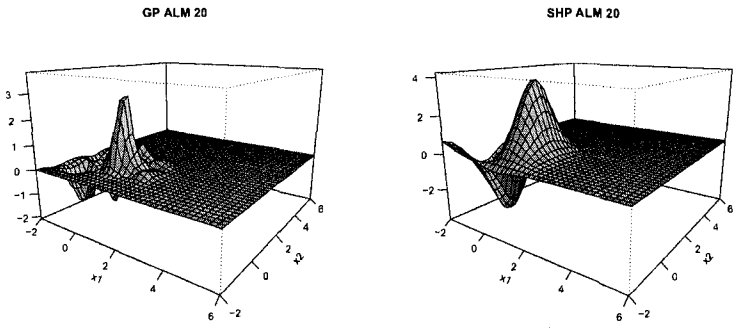


Figure 4.14: Data locations for 2-d test function. The open squares are the 20 locations for initial data set. The solid dots are the candidate locations for adaptive sampling and the gray dots are the test locations.

The ALM criterion was used in the SHP model to guide further sampling, while both ALM and ALC criteria were used in the GP model to guide further sampling. Figure 4.16 (a) and (b) are the ALM and ALC surface for the GP model based on 20 initial inputs. The ALM and ALC surfaces for the GP model agree on the area of large predictive variances, which are fairly uniform away from initial sample locations. But difference exists about which point to sample next. That location is marked by 1 in Figure 4.17 (a) and (b). Even though these two points are located at different places, they are away from the initial sample data points.

Figure 4.16 (c) is the ALM surface for the SHP model based on 20 initial inputs. In contrast with Figure 4.16 (a) and (b), these prediction error variances



(a) (b)

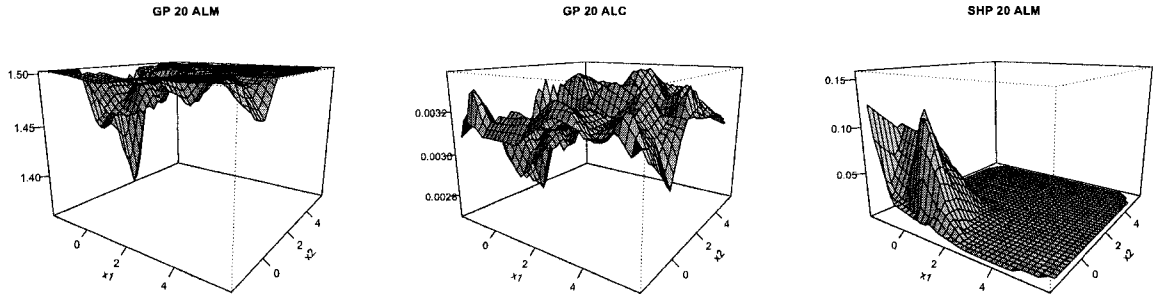
Figure 4.15: (a) Fitted surface for SHP model based on 20 initial data points. (b) Fitted surface for GP model based on 20 initial data points.

are far from uniform away from initial sample locations, and instead have hot spots of high uncertainty. The SHP selects its next sample point, marked by 1 in Figure 4.17 (c), within this hot spot.

As the sample size increases from 20 to 30, the more differences arise among GP ALM, GP ALC and SHP ALM in this example. The SHP model puts all extra points in the area of interest, i.e. the first quadrant. GP ALC puts 10 extra points both in the first quadrant and some locations away from previous sample points. And GP ALM puts all 10 extra points in a fairly uniform way across the previously unsampled part of the input space. This results confirms that ALC algorithm is more likely to put the data points inside the input space than ALM does. These 10 extra points in Figure 4.17 explains why the average RMSE curves for different sampling methods have different rate of decrease at sample size 30.

As the sample is extended, the difference between ALC and ALM for GP diminishes since the candidate data set is evenly distributed over the input space; the SHP ALM starts to select the sample points at unsampled parts of the input space. This can be seen from the ALM and ALC surfaces in Figure 4.18.

We stop sampling the process at sample size 60 (40 adaptively sampled points). These 40 sampled locations with different methods are marked and ordered with



(a) (b) (c)

Figure 4.16: (a) ALM surface for GP model fitted with 20 initial data points. (b) ALC surface for GP model fitted with 20 initial data points. (c) ALM surface for SHP model fitted with 20 initial data points.

numbers in Figure 4.19. As we mentioned before, even though the ALM method is more likely to place the data points around the boundary than the ALC method, this difference can be reduced by using a candidate data set that is evenly spread out over the input space. In this example, with clear inhomogeneity in the surface, the SHP provides a much better adaptive sampling scheme than the GP. The ALM or ALC adaptive sampling with a GP selects the new inputs in a fairly uniform way across the previously unsampled part of the input space, while the ALM adaptive sampling with SHP selects the new input in the area of large volatility, i.e. the first quadrant.

4.3.2 SIR model revisited

The previous examples show the potential of adaptive sampling with the SHP model in low-dimensional cases. We now look at adaptive sampling with SHP in a higher-dimensional example, the SIR model of Section 3.3. Recall that the SIR model is a 7-dimensional example with three quantities of interest: the average number of susceptible individuals $q(\boldsymbol{x})_1 = \frac{1}{T}\int_0^T S(s,\boldsymbol{x})ds$, the average number of

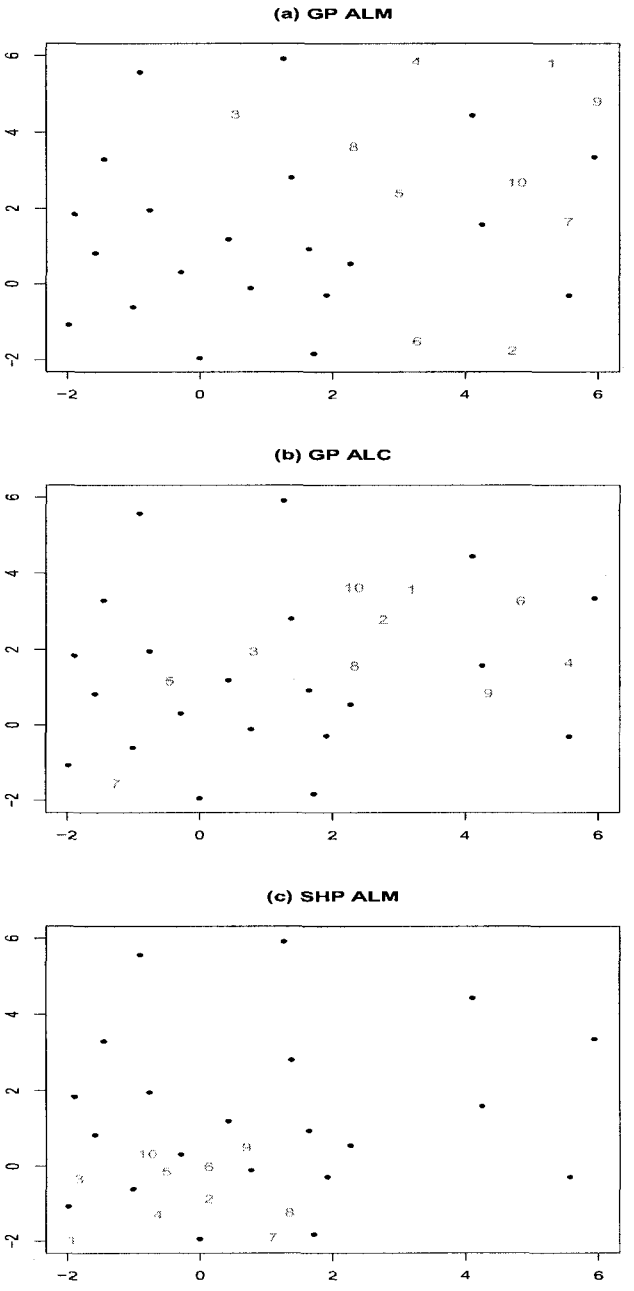


Figure 4.17: Locations for the first 10 adaptively sampled points via GP ALM (top), GP ALC (middle) and SHP ALM (bottom) methods. The solid dots are the locations for the 20 initial points. The numbers represent the 10 adaptively sampled locations in order of selection.

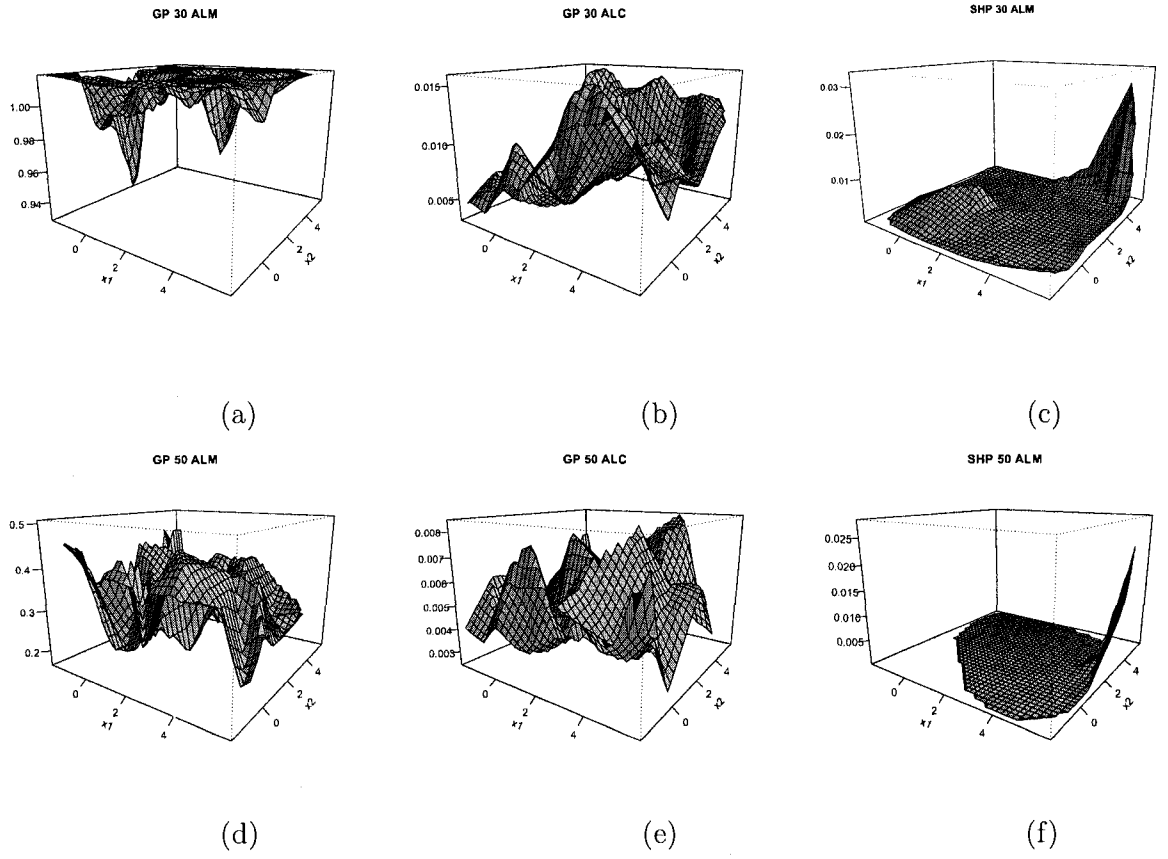


Figure 4.18: (a) ALM surface for GP model based on 30 points with 10 extra points adaptively sampled by GP ALM. (b) ALC surface for GP model based on 30 points with 10 extra points adaptively sampled by GP ALC. (c) ALM surface for SHP model based on 30 points with 10 extra points adaptively sampled by SHP ALM. (d) ALM surface for GP model based on 50 points with 30 extra points adaptively sampled by GP ALM. (e) ALC surface for GP model based on 50 points with 30 extra points adaptively sampled by GP ALC. (f) ALM surface for SHP model based on 50 points with 30 extra points adaptively sampled by SHP ALM.

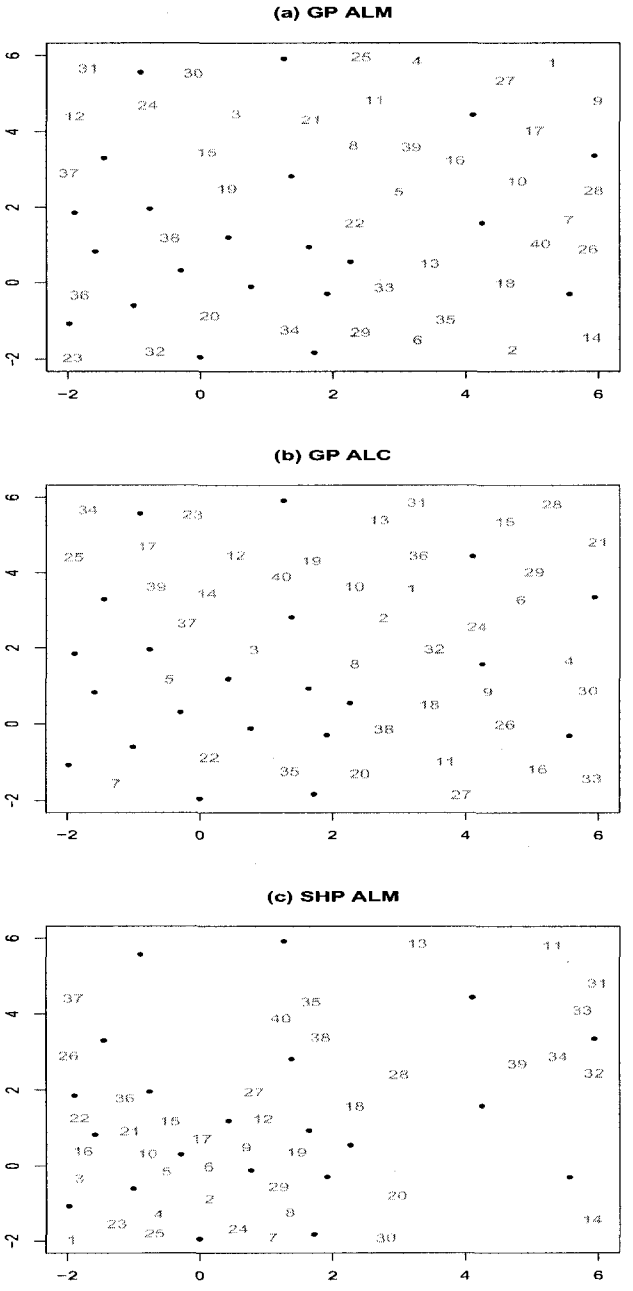


Figure 4.19: Locations for the 40 adaptively sampled points via GP ALM (top), GP ALC (middle) and SHP ALM (bottom) methods. The solid dots are the locations for the initial 20 points. The numbers represent the 40 adaptively sampled locations in order of selection.

infected individuals $q(\boldsymbol{x})_2 = \frac{1}{T}\int_0^T I(s,\boldsymbol{x})ds$, and the average number of resistant individuals $q(\boldsymbol{x})_3 = \frac{1}{T}\int_0^T R(s,\boldsymbol{x})ds$ over a time interval $[0,T]$.

We use a random LHS design to sample 40 points in the input space as the initial sample. A set of 1000 data points sampled by maximin distance LHS design is used as candidate data for further adaptive sampling. The candidate data set is also used as the test data set. Given the design we choose, the candidate data are spread out over the 7-dimensional space. Even though the ALM method is more likely to place the sample points close to the boundary (Gramacy et al. (2004)), it can be similar to ALC method if the candidate data points are spread out. For simplicity, we only use ALM method to do the adaptive sampling in this high-dimensional example. The three response variables are studied separately. The model is updated each time the sample size equals a multiple of 10.

For each quantity of interest, we fit the 40 initial points with a SHP model, compute the predictive variances at 1000 candidate locations and select the next point with largest predictive variance. We keep sampling other points sequentially through the ALM procedure as described above until the sample size increased to 100 locations. With each new sample observation, we re-compute the predictors for the remaining test points using the GP model and the SHP model, and compute the RMSE. This process of SHP ALM adaptive sampling, fitting, and prediction was repeated 20 times, with different random seeds used at each time to generate the initial data set. We compute the average RMSE over the 20 replicates at different sample sizes.

Similarly, we fit each of the 40 initial data sets with the GP model and increase the sample size to 100 using the ALM criterion with GP model predictive variances. The whole process of fitting and predicting was replicated 20 times. The average RMSE curves as a function of sample size for SHP and GP model fitting with SHP and GP ALM adaptively sampled data points for each quantity of interest are plotted in Figure 4.20 to Figure 4.22.

Figure 4.20 to Figure 4.22 give a graphical comparison of the performance of GP and SHP for prediction and adaptive sampling of each quantity of interest. In each figure, strategies are denoted by [ALM sampling method]-[prediction method]. For example, GP(ALM)-SHP is SHP model fitting and prediction with GP-ALM adaptively sampled data.

For each quantity of interest, the SHP model fitting with SHP ALM adaptively sampled data outperforms the GP model fitting with GP ALM adaptively sampled data. However, the SHP ALM is not as efficient as with the previous 2-d test function. One reason is that there do not exist obvious inhomogeneities on the true surfaces for $q1$, $q2$, and $q3$. For homogeneous surfaces, the GP can perform similarly to SHP. Another reason is that 1000 evenly-distributed test data points are not dense enough in the 7-d input space. The adaptive sampling methods would require more dense test data to capture local behavior.

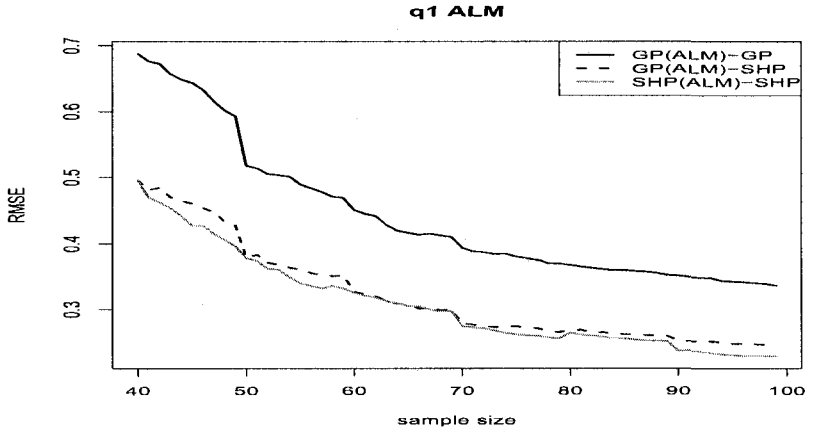


Figure 4.20: Average RMSE as a function of sample size for SHP and GP models for $q1$ in SIR. The GP and SHP models are updated each time the sample size equals a multiple of 10. Strategies are denoted by [ALM sampling method]-[prediction method]. For example, GP(ALM)-SHP is SHP model fitting and prediction with GP-ALM adaptively sampled data.

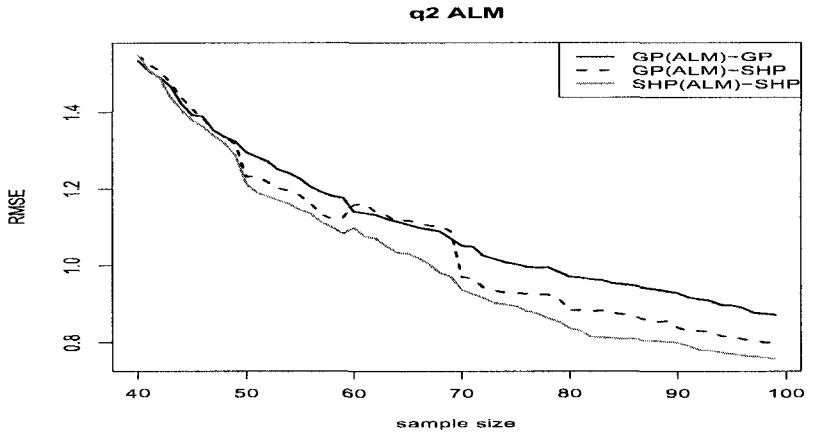


Figure 4.21: Average RMSE as a function of sample size for SHP and GP models for $q2$ in SIR. The GP and SHP models are updated each time the sample size equals a multiple of 10. Strategies are denoted by [ALM sampling method]-[prediction method]. For example, GP(ALM)-SHP is SHP model fitting and prediction with GP-ALM adaptively sampled data.

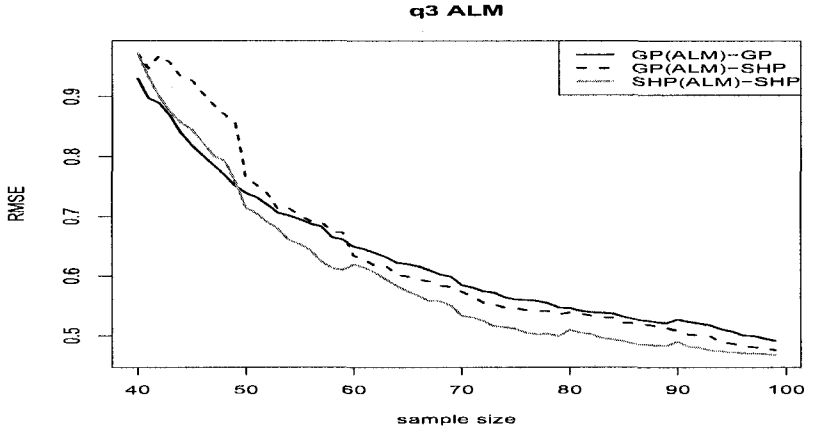


Figure 4.22: Average RMSE as a function of sample size for SHP and GP models for $q3$ in SIR. The GP and SHP models are updated each time the sample size equals a multiple of 10. Strategies are denoted by [ALM sampling method]-[prediction method]. For example, GP(ALM)-SHP is SHP model fitting and prediction with GP-ALM adaptively sampled data.

Chapter 5

MODELING LOCAL SENSITIVITY

Some computer experiments provide both $y(\cdot)$ and its first partial derivatives at observed inputs $\boldsymbol{x}$. These partial derivatives are also called local sensitivities. These local derivatives can provide additional information about the surface that is useful in reconstructing the whole surface. In the computer experiments, most mathematical models are based on systems of differential equations. The system usually has a large number of input parameters and is expensive to execute. Local sensitivity provides the slope of the calculated model output at a given set of values in the input space. The information is useful as a cheap approximation for the output from the model. In their study, Estep and Neckels (2006) introduce the procedure *Higher Order Parameter Sampling* (HOPS) for approximating a quantity of interest based on a system of differential equations using local sensitivity. Section 5.1 reviews HOPS method.

In Section 5.2, we introduce the traditional GP simultaneous modeling of response and derivatives. Similarly, we can extend the SHP to model the response and its derivative. Properties of SHP derivatives are discussed in section 5.3. Due to the high-dimensional integration in the likelihood calculation, we propose the low-rank SHP model for fitting the response and derivatives in Section 5.4. A low-rank importance density is developed to improve the efficiency of likelihood computation and the estimation of the latent process. In Section 5.5, EBP is used for predicting the response at an untried location based on the combined information of response and derivatives. For using derivative information, we evaluate the performance of

HOPS approximation, GP model and SHP model through two 1-dimensional and two 2-dimensional test functions. We also revisit the SIR example. The stochastic modeling of response and derivative outperforms HOPS, and the SHP model outperforms the GP model.

In the work of Estep and Neckels (2006), the authors develop an adaptive sampling method, Fast Adaptive Parameter Sampling (FAPS), using derivative information. The method of FAPS also provides sensitivity analysis of input variables and can be used as a sampling method in stochastic modeling of response and derivatives. The method is illustrated via the SIR model. We use SHP and GP to fit the response and derivatives at the FAPS adaptively sampled locations and compare the results from the HOPS approximation. One of the disadvantages of stochastic modeling is the existence of numerical problem associated with calculating determinant of the large joint covariance matrix of response and derivatives in the optimization process for the high-dimensional case. Some approaches to alleviate this problem are proposed.

5.1 High Order Parameter Sampling (HOPS)

Estep and Neckels (2006) provide a fast and reliable method for approximating the model output from a set of differential equations. The following is a detailed description of their method. We use their notation in the description.

Consider the problem of determining the effects of variations in inputs on a quantity of interest computed from the solution of the initial value problem

$$\begin{cases} \dot{\boldsymbol{x}}(t;\boldsymbol{\lambda}) &= \boldsymbol{f}(\boldsymbol{x}(t;\boldsymbol{\lambda});\boldsymbol{\lambda}_1), \ t>0, \\ \boldsymbol{x}(0;\boldsymbol{\lambda}) &= \boldsymbol{\lambda}_0, \end{cases} \tag{5.1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{f} : \mathbb{R}^{n+p} \to \mathbb{R}^n$. The parameter $\boldsymbol{\lambda} = (\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_0)^T \in \mathbb{R}^d$ with $d = n + p$, where $\boldsymbol{\lambda}_1 \in \mathbb{R}^p$ represents parameters in the model $\boldsymbol{f}$ and $\boldsymbol{\lambda}_0 \in \mathbb{R}$ represents the initial conditions (which also are considered as parameters, though

they are sometimes fixed). The goal is to quantify how variation in $\boldsymbol{\lambda}$ affects the solution to (5.1).

We consider the practical goal of computing a quantity of interest, which can be represented as a linear functional of the form

$$q(\boldsymbol{\lambda}) = \int_0^T \langle \boldsymbol{x}(s;\boldsymbol{\lambda}), \boldsymbol{\psi}(s)\rangle ds, \tag{5.2}$$

where $\langle \cdot, \cdot \rangle$ represents the inner product. We consider $\boldsymbol{\lambda} = \boldsymbol{\lambda}(\omega)$ as a random vector on a probability space $(\Omega, \mathcal{B}, \mathcal{P})$ and $\boldsymbol{\psi}$ is a function of time corresponding to the quantity of interest. Some common choices for $\boldsymbol{\psi}$ are

- $\boldsymbol{\psi} = \delta(s-t)(0,...,1,0,...)'$, which yields the i^{th} component of $\boldsymbol{x}(t,\omega)$ at time t.
- $\boldsymbol{\psi} = (1,...,1)'/T$, which yields the time average over $[0,T]$ of the sum of all components.
- $\boldsymbol{\psi} = (0,...,1,0,...)'/T$, which yields the time average over $[0,T]$ of a particular component of the solution.

The generalized Green's function is introduced to solve the adjoint problem to the linearized equation

$$\begin{cases} -\dot{\boldsymbol{\phi}}(t) - A^T(t)\,\boldsymbol{\phi}(t) & = \boldsymbol{\psi}(t), \;\; T \geq t \geq 0, \\ \boldsymbol{\phi}(T) & = 0, \end{cases} \tag{5.3}$$

where

$$\begin{aligned} A(t) &= D_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{y}(t);\boldsymbol{\mu}_1) \\ &= \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}. \end{aligned}$$

Estep and Neckels have shown that the quantity of interest can be approximated by

$$\begin{aligned} q(\boldsymbol{\lambda}) &= \int_0^T \langle \boldsymbol{x}, \boldsymbol{\psi} \rangle ds \\ &\approx \int_0^T \langle \boldsymbol{y}, \boldsymbol{\psi} \rangle ds + \langle \boldsymbol{\lambda}_0 - \boldsymbol{\mu}_0, \boldsymbol{\phi}(0) \rangle \\ &+ \int_0^T \langle D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \boldsymbol{\mu}_1)(\boldsymbol{\lambda}_1 - \boldsymbol{\mu}_1), \boldsymbol{\phi}(\boldsymbol{\mu}) \rangle ds. \end{aligned} \tag{5.4}$$

The representation in (5.4) is a linear approximation to the quantity of interest q at the reference value $\boldsymbol{\mu}$. Then the *one-point HOPS* approximation can be written as

$$q(\boldsymbol{\lambda}) \approx q(\boldsymbol{\mu}) + \langle \nabla q(\boldsymbol{\mu}), (\boldsymbol{\lambda} - \boldsymbol{\mu}) \rangle, \tag{5.5}$$

and the local sensitivity can be computed as

$$\begin{aligned} \nabla q(\boldsymbol{\mu}) &= \int_0^T \langle D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \boldsymbol{\mu}_1), \boldsymbol{\phi}(\boldsymbol{\mu}) \rangle ds \\ &= \int_0^T D_{\boldsymbol{\lambda}_1} \boldsymbol{f}(\boldsymbol{y}; \boldsymbol{\mu}_1)^T \boldsymbol{\phi}(\boldsymbol{\mu}) ds. \end{aligned}$$

We can use (5.4) to obtain a good approximation of the distribution of $q(\boldsymbol{\lambda})$, provided $\boldsymbol{\lambda}$ has small variance about $\boldsymbol{\mu}$ and the function q is approximated well by its linearization at $\boldsymbol{\mu}$. In order to obtain an accurate global approximation, we need to combine HOPS approximations computed at multiple reference values, which leads to the *multi-point Higher Order Parameter Sampling* approximation.

We choose a sample $\{\boldsymbol{\mu}_i\}_{i=1}^N$ of the parameter space and partition the parameter space into a collection of generalized rectangles $\{R_i\}_{i=1}^N$ with $\boldsymbol{\mu}_i \in R_i$ for all i. The corresponding piecewise linear HOPS approximation is defined by

$$q(\boldsymbol{\lambda}) \approx \tilde{q}(\boldsymbol{\lambda}) = \sum_{i=1}^{N} (q(\boldsymbol{\mu}_i) + \langle \nabla q(\boldsymbol{\mu}_i), (\boldsymbol{\lambda} - \boldsymbol{\mu}_i) \rangle) \chi_{R_i}(\boldsymbol{\lambda}). \tag{5.6}$$

The implementation procedure of multi-point HOPS approximation is summarized as following:

- Choose multiple reference points $\{\boldsymbol{\mu}_i\}_{i=1}^N$

- Solve the forward system (5.1) to get the solution $y(t;\boldsymbol{\mu}_1),...,y(t;\boldsymbol{\mu}_N)$
- Calculate $D_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{y}_i;\boldsymbol{\mu}_i)$ and $D_{\boldsymbol{\lambda}_1}\boldsymbol{f}(\boldsymbol{y}_i;\boldsymbol{\mu}_i)$ for $i=1,...,N$
- Solve the adjoint system (5.4) to get $\phi(t;\boldsymbol{\mu}_1),...,\phi(t;\boldsymbol{\mu}_N)$
- Calculate the gradient information $\int_0^T D_{\boldsymbol{\lambda}_1}\boldsymbol{f}(\boldsymbol{y}_i;\boldsymbol{\mu}_i)^T\boldsymbol{\phi}(\boldsymbol{\mu}_i)ds$ for $i=1,...,N$
- Sample $\boldsymbol{\lambda}$ from input space and use equation (5.6) to calculate $\tilde{q}(\boldsymbol{\lambda})$

HOPS is a very cheap approximation for the quantity of interest. If the response surface is very smooth, HOPS might be a good metamodel since derivatives can help prediction a great deal when data set is small. However the computer code output can be nonlinear and inhomogeneous. The equation for the HOPS approximation has the form of the first-order Taylor expansion around the reference value $\boldsymbol{\mu}$. This leads to one disadvantage of HOPS, which requires many reference points in general to have a good global approximation. A large data set is not feasible for expensive computer experiments. But HOPS provides a way to compute the local estimate of the error.

If we write the one-point HOPS approximation as

$$q(\boldsymbol{\lambda})-q(\boldsymbol{\mu})\approx\langle\nabla q(\boldsymbol{\mu}),(\boldsymbol{\lambda}-\boldsymbol{\mu})\rangle,$$

we see that the derivative information $\langle\nabla q(\boldsymbol{\mu}),(\boldsymbol{\lambda}-\boldsymbol{\mu})\rangle$ provides a local estimate of the error that results from using the sample value $q(\boldsymbol{\mu})$ in place of the actual value $q(\boldsymbol{\lambda})$ for $\boldsymbol{\lambda}$ near $\boldsymbol{\mu}$.

If we view a sample as a piecewise constant approximation, then

$$\tilde{q}(\boldsymbol{\lambda})=\sum_{i=1}^{N}(q(\boldsymbol{\mu}_i)+\langle\nabla q(\boldsymbol{\mu}_i),(\boldsymbol{\lambda}-\boldsymbol{\mu}_i)\rangle)\chi_{R_i}(\boldsymbol{\lambda}). \tag{5.7}$$

Using a Taylor expansion in each of the R_i as in HOPS, we obtain an approximation of the expected value of the error in the L^1 norm,

$$\epsilon^{pc}=\sum_{i=1}^{N}\int_{R_i}|\langle\nabla q(\boldsymbol{\mu}_i),(\boldsymbol{z}-\boldsymbol{\mu}_i)\rangle|d\mu_\lambda(\boldsymbol{z}). \tag{5.8}$$

5.1.1 Fast adaptive parameter sampling (FAPS)

The accurate estimate of the error for a given sample offers the possibility of optimizing the sample process in order to reach a desired accuracy with minimal computational cost. The adaptive strategy works in an iterative fashion. Given a current set of sample points, we estimate the local contribution to the error for the sample and choose additional sample points in regions in which the contributions to the error are estimated to be largest.

One approach of adaptive sampling described in Estep and Neckels (2006) is analogous to the standard h-refinement strategy in adaptive finite element methods. Define ϵ_i^{pc} to be approximate contribution to the error bound from rectangle R_i, i.e.,

$$\epsilon_i^{pc} = \int_{R_i} |\langle \nabla q(\boldsymbol{\mu}_i), (\boldsymbol{z} - \boldsymbol{\mu}_i) \rangle| d\mu_\lambda(\boldsymbol{z}).$$

The adaptive strategy is to refine some fraction of the rectangles on which ϵ_i^{pc} is largest. We refine along one dimension at a time since q may be very sensitive to changes in one parameter, but not in others. To measure the contribution to the error bound from each dimension, we define $\epsilon_{i,k}^{pc}, k = 1, ..., d$ by

$$\epsilon_{i,k}^{pc} = \int_{R_i} |\partial_{\lambda_k} q(\boldsymbol{\mu}_i)(z^k - \mu_i^k)| d\mu_\lambda(\boldsymbol{z}), \tag{5.9}$$

where $\boldsymbol{z} = (z^1, ..., z^d)^T$ and $\boldsymbol{\mu}_i = (\mu_i^1, ..., \mu_i^d)^T$ so

$$\epsilon^{pc} \leq \sum_{i=1}^{N} \epsilon_i^{pc} \leq \sum_{i=1}^{N} \sum_{k=1}^{d} \epsilon_{i,k}^{pc}.$$

For a rectangle where ϵ_i^{pc} is large enough for refinement, the maximum contribution $\epsilon_{i,k}^{pc}, k = 1, ..., d$ is identified and the rectangle is divided along this dimension. Since the value at the center of the rectangle is known, we split in thirds along this dimension, compute the values at the centers of the two new rectangles and iterate till a given error tolerance level (TOL) or a desired sample size is achieved. By looking at number of splits by input dimension, we can obtain sensitive analysis of

input variables. But the FAPS method is limited by the dimensionality of the input space since the number of partitions will increase exponentially as the dimension of the parameter space increases.

5.2 GP Modeling of Derivatives

The alternative to HOPS is to use stochastic modeling of response and derivatives. As we have mentioned in the introduction, given the differentiability of the covariance function, the derivative of a GP is still a Gaussian covariance and can be modeled simultaneously with the response (Morris et al. (1993)). In this section, we will give the equations for GP modeling responses and derivatives in the one-dimensional case with constant mean. For simplicity, we consider a GP with constant mean and Gaussian covariance function. Denote the vector of the first partial derivatives at sampled points as $y'(x)$. The model for response and derivative is:

$$\begin{aligned} y(x) &= \beta + Z(x) \\ y'(x) &= Z'(x), \end{aligned}$$

where $\mathrm{Cov}(Z(x_i), Z(x_j)) = \exp(-\phi(x_i - x_j)^2)$. By taking the derivative of the covariance function with respect to x, we can get the covariance function of y and y' and the covariance function of y' as

$$\begin{aligned} \mathrm{Cov}(y(x_i), y'(x_j)) &= 2\phi(x_i - x_j)\exp(-\phi(x_i - x_j)^2) \\ \mathrm{Cov}(y'(x_i), y(x_j)) &= -2\phi(x_i - x_j)\exp(-\phi(x_i - x_j)^2) \\ \mathrm{Cov}(y'(x_i), y'(x_j)) &= (2\phi - 4\phi(x_i - x_j)^2)\exp(-\phi(x_i - x_j)^2). \end{aligned} \tag{5.10}$$

Given a data set at a coarse grid locations $\{(x_i, y_i, y_i')\}_{i=1}^n$, $x_i \in \mathbb{R}$, we want to predict the responses at an unobserved location x_0. Since the joint distribution of response and derivative is still multivariate GP, the conditional distribution of

$y(x_0)$ given responses and derivatives is normal. The mean and variance of the conditional distribution can be used as predictor and predictive variance for $y(x_0)$. This approach can be easily extended to higher dimensional space (Morris et al. (1993)).

5.3 SHP Modeling of Derivatives

As with the GP, the derivatives of the SHP can also be modeled jointly with the response y. For simplicity, we consider the model for y' in 1-dimensional space:

$$y'(x) = \boldsymbol{g}'(x)^T\boldsymbol{\beta} + \sigma \exp(\tau\alpha(x)/2)Z'(x) + \sigma \exp(\tau\alpha(x)/2)\frac{\tau\alpha'(x)Z(x)}{2}. \quad (5.11)$$

The derivative process has mean $\boldsymbol{g}'(x)^T\boldsymbol{\beta}$, variance $\sigma^2 \exp(\tau^2/2)\{\gamma_{z'}(0) + \frac{\tau^2}{4}\gamma_{\alpha'}(0)\}$. By taking the partial derivative of the unconditional SHP covariance function for the response, the unconditional correlation function of the SHP derivative is

$$\begin{aligned}\rho_{y'}(h) = \frac{\exp(-\frac{\tau^2}{4}+\frac{\tau^2}{4}\gamma_\alpha(h))}{\gamma_{z'}(0)+\frac{\tau^2}{4}\gamma_{\alpha'}(0)}\{&\gamma_{z'z'}(h) + \tfrac{\tau^2}{4}\gamma_{\alpha\alpha'}(h)\gamma_{z'z}(h) + \tfrac{\tau^2}{4}\gamma_{\alpha'\alpha}(h)\gamma_{zz'}(h) \\ &+\tfrac{\tau^2}{4}[\gamma_{\alpha'\alpha'}(h) + \tfrac{\tau^2}{4}\gamma_{\alpha'\alpha}(h)\gamma_{\alpha\alpha'}(h)]\gamma_{zz}(h)\}.\end{aligned} \quad (5.12)$$

The effect of varying the correlation parameter ϕ_z on the unconditional correlation function can be seen from Figure 5.2. For a relatively smooth α process (ϕ_α is small), panels (a) shows unconditional correlation function for the SHP derivative y', and panel (b) shows unconditional correlation function between response y and derivative y'. For a relative rough α process (ϕ_α is large), panel (c) shows unconditional correlation function for the SHP derivative y' and panel (d) shows unconditional correlation function between response y and derivative y'. Comparing these with the correlation plots for the Gaussian derivative (Figure 5.1), we can see that for small ϕ_α and τ^2 values, the pattern of ϕ_z effects on the unconditional correlation function for the SHP derivative is similar to the pattern of ϕ effects on the correlation function for the GP derivative. As ϕ_z increases, the correlation for the derivative goes to zero more quickly and the sign of correlation changes more

quickly. The magnitude of the strongest negative correlation does not change with ϕ_z. On the other hand, the magnitude for the strongest negative correlation does change with ϕ_α and τ^2 in SHP model as shown in Figure 5.3 and Figure 5.4. The correlation function decays smoothly for small values of ϕ_α. For large values of ϕ_α, increasing τ^2 increases the variety of correlation functions. The effect of ϕ_α on the unconditional correlation function relates to the value of ϕ_z. As we increase ϕ_z from 10 to 100, as shown in Figure 5.3, the decay of correlation function is dominated by the large value of ϕ_z, and the small range of ϕ_α values has no very obvious effect on the correlation functions.

The unconditional correlation plots for the SHP derivative show unique characteristics, implying that (5.12) can be used as as a new class of isotropic oscillating correlation functions in the GP model. We will illustrate the application of this new function in a traditional GP model with 1-d and 2-d test functions in later sections.

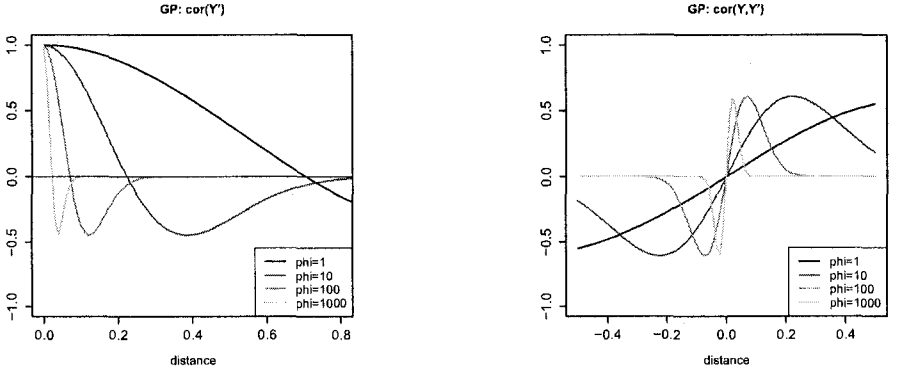


Figure 5.1: Effect of ϕ on correlation function for GP. The left panel shows correlation functions of GP derivative. The right panel shows correlation functions between response and its derivative for GP. Gaussian covariance function are used in all the plots.

5.4 Low-Rank Modeling of SHP Derivatives

For the SHP model, the derivatives add extra information into modeling but the dimensionality of the latent process also increases. The latent process and its

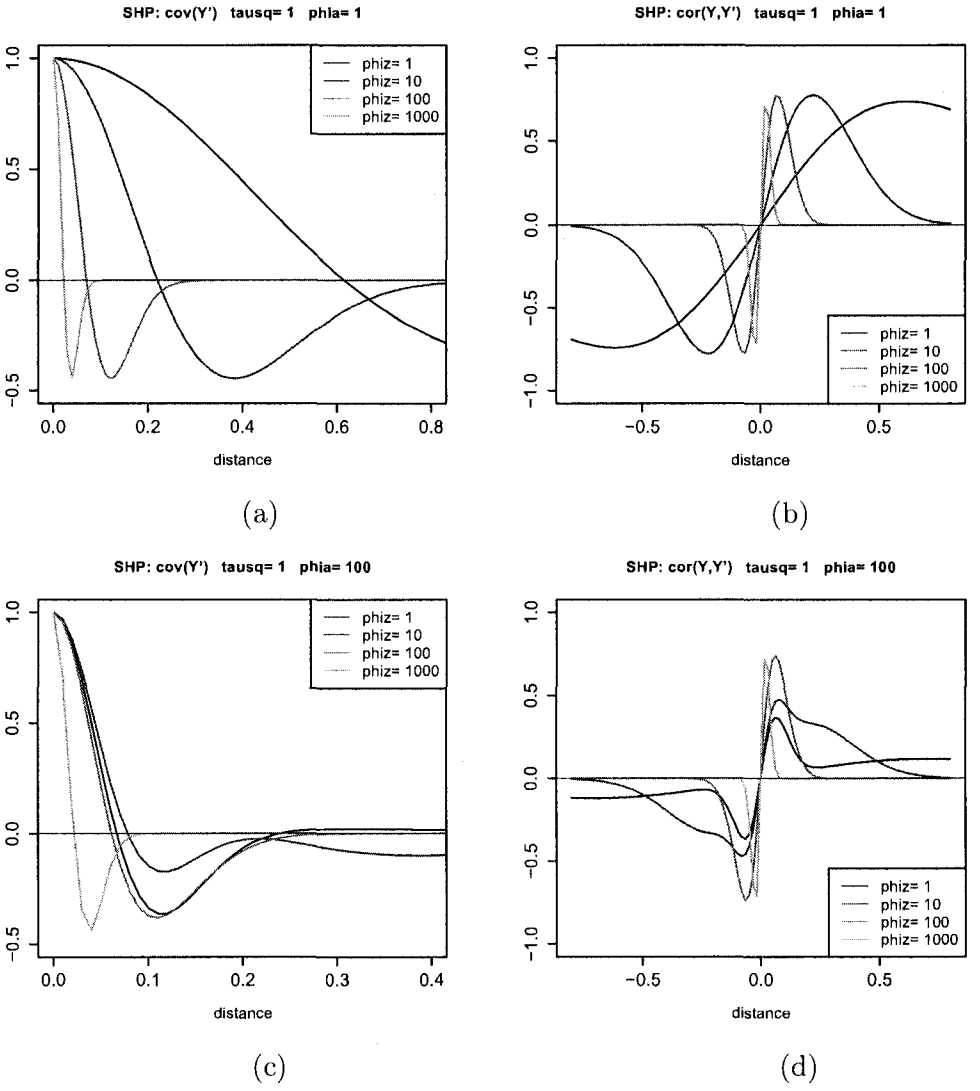


Figure 5.2: Effect of ϕ_z on unconditional correlation function for SHP. (a) Correlation functions of SHP derivative with $\phi_\alpha = 10$. (b) Correlation functions between SHP response and derivative with $\phi_\alpha = 10$. (c) Correlation functions of SHP derivative with $\phi_\alpha = 100$. (d) Correlation functions between SHP response and derivative with $\phi_\alpha = 100$. In all plots, Gaussian covariance function is used for α and Z processes and $\tau^2 = 1$.

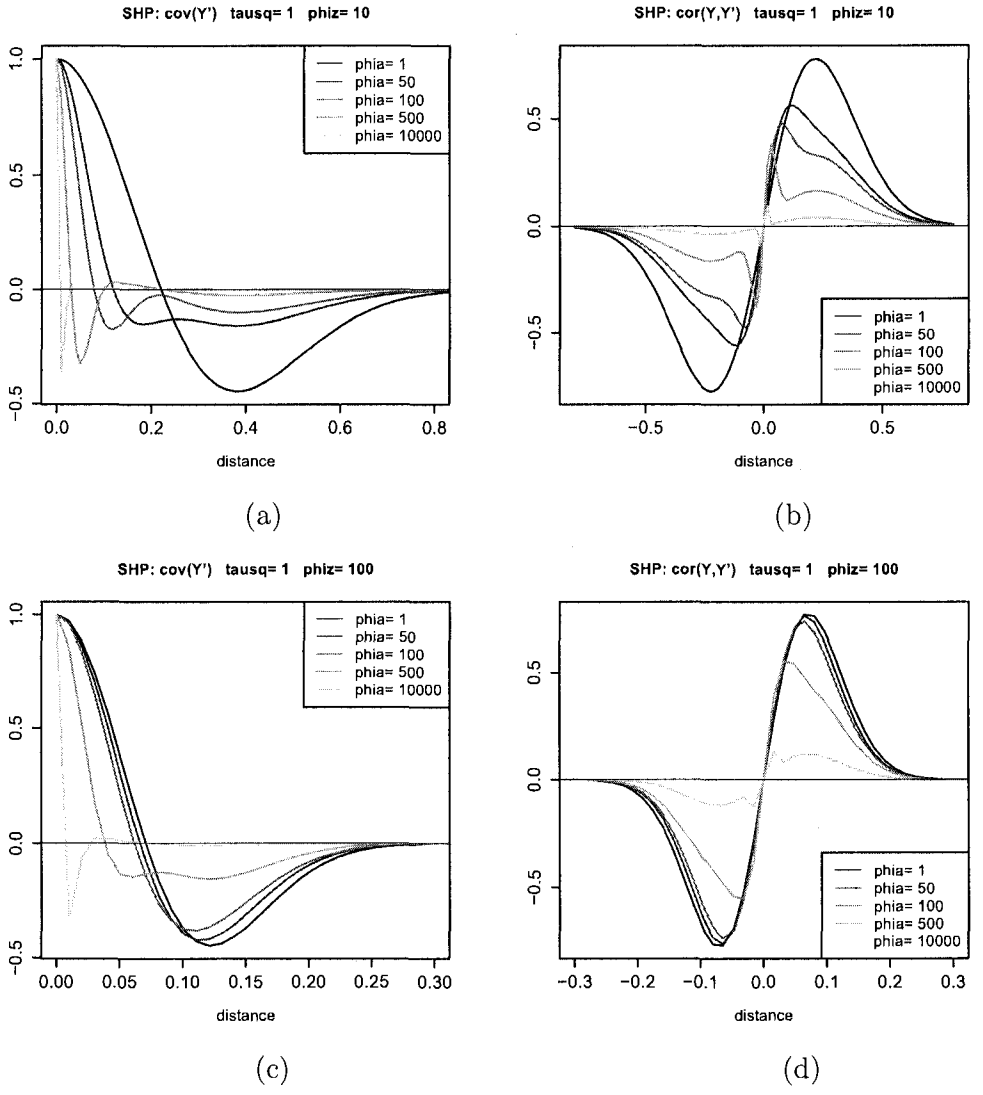


Figure 5.3: Effect of ϕ_α on unconditional correlation function for SHP . (a) Correlation functions of SHP derivative with $\phi_z = 10$. (b) Correlation functions between SHP response and derivative with $\phi_z = 10$. (c) Correlation functions of SHP derivative with $\phi_z = 100$. (d) Correlation functions between SHP response and derivative with $\phi_z = 100$. In all plots, Gaussian covariance function is used for α and Z processes and $\tau^2 = 1$.

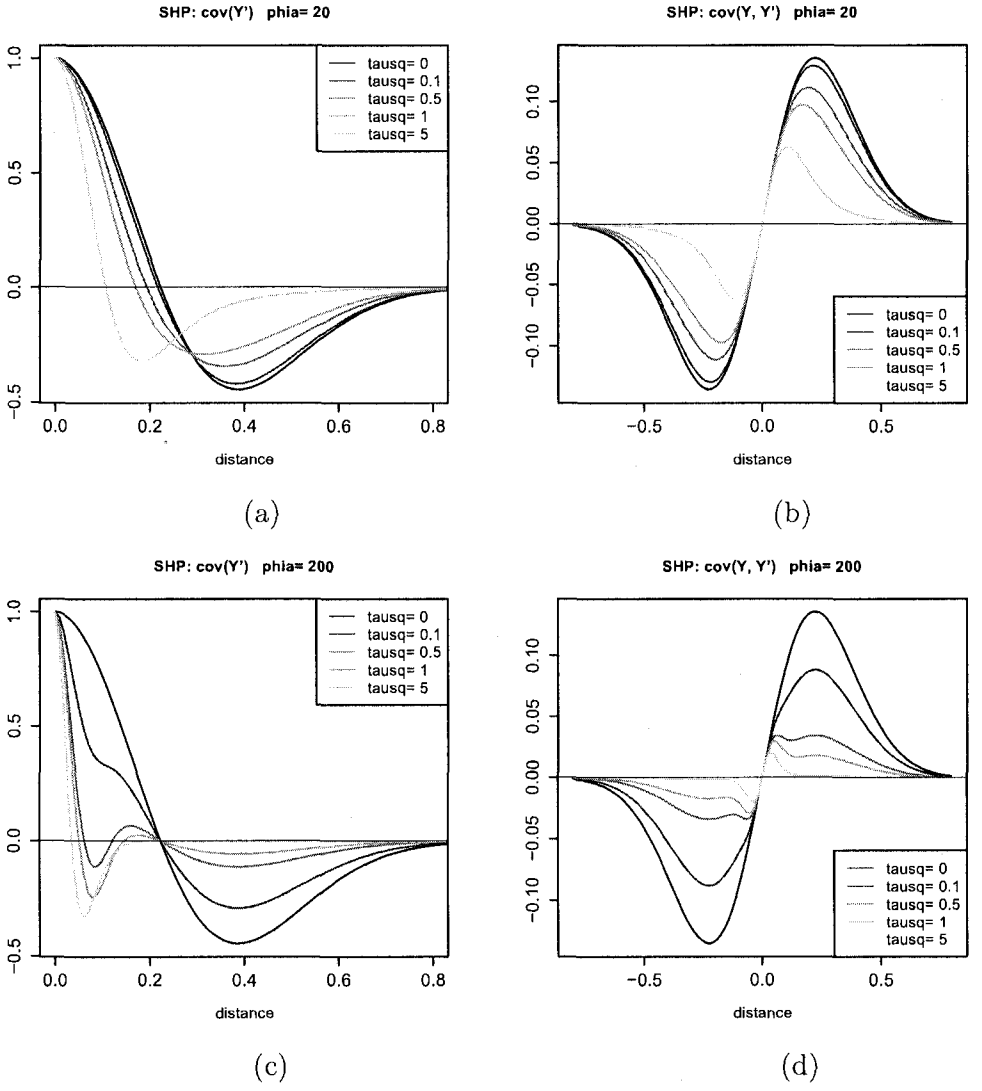


Figure 5.4: Effect of τ^2 on unconditional correlation function for SHP. (a) Correlation functions of SHP derivative with $\phi_\alpha = 20$. (b) Correlation functions between SHP response and derivative with $\phi_\alpha = 20$. (c) Correlation functions of SHP derivative with $\phi_\alpha = 200$. (d) Correlation functions between SHP response and derivative with $\phi_\alpha = 200$. In all plots, Gaussian covariance function is used for α and Z processes and $\phi_z = 10$.

derivative has same dimensionality as the joint vector of response and derivatives. To avoid the high-dimensional integration problem in the likelihood computation, we propose a low-rank SHP modeling of response and derivatives and use it in the procedure of likelihood computation and prediction.

We consider a 1-dimensional SHP model for the response and derivative. The model can be written in matrix form as

$$\begin{aligned} \boldsymbol{y} &= G^T\boldsymbol{\beta} + \sigma \operatorname{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}\boldsymbol{Z} \\ \boldsymbol{y}' &= G'^T\boldsymbol{\beta} + \sigma \operatorname{diag}\left\{\exp\left(\frac{\tau\boldsymbol{\alpha}}{2}\right)\right\}\left\{\frac{\tau}{2}\operatorname{diag}\{\boldsymbol{\alpha}'\}\boldsymbol{Z} + \boldsymbol{Z}'\right\} \end{aligned} \tag{5.13}$$

Since the low-rank kriging approximation for the latent process α is written as $\boldsymbol{\alpha} = B\boldsymbol{\omega}$ and $\boldsymbol{\alpha}' = B'\boldsymbol{\omega}$, the low-rank SHP model of responses and derivatives is given by

$$\begin{aligned} \boldsymbol{y} &= G^T\boldsymbol{\beta} + \sigma \operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}\boldsymbol{Z} \\ \boldsymbol{y}' &= G'^T\boldsymbol{\beta} + \sigma \operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}\left\{\frac{\tau}{2}\operatorname{diag}\{B'\boldsymbol{\omega}\}\boldsymbol{Z} + \boldsymbol{Z}'\right\}, \end{aligned} \tag{5.14}$$

where $\boldsymbol{\omega} \sim N(0, \Omega^{-1})$. If using a Gaussian covariance function in the α process,

$$B(i,j) = \exp(-\phi_\alpha(x_i - \kappa_j)^2),\ i = 1, ..., n,\ j = 1, ..., J$$

and

$$\Omega(i,j) = \exp(-\phi_\alpha(\kappa_i - \kappa_j)^2),\ i = 1, ..., J,\ j = 1, ..., J.$$

The conditional joint distribution of $p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\psi}, \boldsymbol{\omega})$ is multivariate normal, i.e.,

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{y}' \end{bmatrix} |\boldsymbol{\omega}, \boldsymbol{\psi} \sim N\left(\begin{bmatrix} G^T \\ G'^T \end{bmatrix}\boldsymbol{\beta}, \sigma^2 \begin{bmatrix} R_{yy} & R_{yy'} \\ R_{y'y} & R_{y'y'} \end{bmatrix}\right), \tag{5.15}$$

where

$$
\begin{aligned}
R_{yy} &= \operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\} R_{zz}\operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\},\\
R_{yy'} &= \operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}\left(\frac{\tau}{2}R_{zz}\operatorname{diag}\{B'\boldsymbol{\omega}\}+R_{zz'}\right)\operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\},\\
R_{y'y} &= \operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}\left(\frac{\tau}{2}\operatorname{diag}\{B'\boldsymbol{\omega}\}R_{zz}+R_{z'z}\right)\operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\},\\
R_{y'y'} &= \operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}\left(\frac{\tau^2}{4}\operatorname{diag}\{B'\boldsymbol{\omega}\}R_{zz}\operatorname{diag}\{B'\boldsymbol{\omega}\}+\frac{\tau}{2}\operatorname{diag}\{B'\boldsymbol{\omega}\}R_{zz'}\right.\\
&\quad \left.+\frac{\tau}{2}R_{z'z}\operatorname{diag}\{B'\boldsymbol{\omega}\}+R_{z'z'}\right)\operatorname{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}. \qquad (5.16)
\end{aligned}
$$

5.4.1 Likelihood calculation for low-rank model

Let $\boldsymbol{y} := (y(\boldsymbol{x}_1), ..., y(\boldsymbol{x}_n), \boldsymbol{y}'(\boldsymbol{x}_1), ..., \boldsymbol{y}'(\boldsymbol{x}_n))$ denote the vector of observations and derivatives and $\boldsymbol{\psi} := (\boldsymbol{\theta}, \phi_\alpha)$ the model parameters. Here $\boldsymbol{\theta} := (\sigma^2, \tau^2, \phi_z, \boldsymbol{\beta})$. The joint density of $(\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\omega})$ of the low-rank model (5.14) is given by

$$p(\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\omega}|\boldsymbol{\psi}) = p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha). \qquad (5.17)$$

It follows that the likelihood of the observed data is given by the n-fold integral

$$L(\boldsymbol{\psi}; \boldsymbol{y}, \boldsymbol{y}') = \int p(\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\omega}|\boldsymbol{\psi})d\boldsymbol{\omega} = \int p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)d\boldsymbol{\omega}. \qquad (5.18)$$

The likelihood (5.18) cannot be computed explicitly. Low-rank importance sampling $p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}'\boldsymbol{\psi})$ needs to be introduced to increase computational efficiency and improve the accuracy of the approximation. The integral in (5.18) can be rewritten as

$$
\begin{aligned}
L(\boldsymbol{\psi}; \boldsymbol{y}, \boldsymbol{y}') &= \int \frac{p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)}{p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})}p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})d\boldsymbol{\omega}\\
&= \mathrm{E}_a\left[\frac{p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)}{p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})}\right]. \qquad (5.19)
\end{aligned}
$$

Suppose $\boldsymbol{\omega}^{(1)}, ..., \boldsymbol{\omega}^{(N)}$ are drawn from an importance density $p_a(\omega|\boldsymbol{y}, \boldsymbol{\psi})$. Then (5.19) can be approximated by

$$L(\boldsymbol{\psi}; \boldsymbol{y}, \boldsymbol{y}') \approx \frac{1}{N}\sum_{i=1}^{N}\left[\frac{p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)}{p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})}\right]. \qquad (5.20)$$

Maximizing (5.20) with respect to $\boldsymbol{\psi}$, we can get maximum likelihood estimates $\hat{\boldsymbol{\psi}}$.

5.4.2 Importance density for low-rank modeling of derivatives

The importance density should be chosen to be close to the posterior density $p(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})$. If $p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})$ is exactly equal to $p(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{\psi})$, a sample of only $N = 1$ is required for accurate likelihood calculation.

The importance density $p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})$ for low-rank SHP model has the form

$$p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi}) = N(\boldsymbol{\omega}^*, V_\omega^*), \tag{5.21}$$

where the mean $\boldsymbol{\omega}^*$ is the mode of the log-density of $p(\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\omega}|\boldsymbol{\psi})$

$$\begin{aligned} \log p(\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\omega}|\boldsymbol{\psi}) &= -\frac{n}{2}\log(2\pi) + \frac{1}{2}\log|R_{y^*}| - \frac{1}{2}(\boldsymbol{y}^* - \boldsymbol{\mu}^*)^T R_{y^*}^{-1}(\boldsymbol{y}^* - \boldsymbol{\mu}^*) \\ &\quad -\frac{n}{2}\log(2\pi) + \frac{1}{2}\log|\Omega^{-1}| - \frac{1}{2}\boldsymbol{\omega}^T\Omega\,\boldsymbol{\omega}, \end{aligned} \tag{5.22}$$

$\boldsymbol{y}^* = (\boldsymbol{y}, \boldsymbol{y}')$, and $\boldsymbol{\mu}^* = \begin{bmatrix} G \\ G' \end{bmatrix}\boldsymbol{\beta}$. The covariance matrix for the importance density is $V_\omega^* = (-H)^{-1}$, where H is the Hessian matrix

$$H = \frac{\partial^2}{\partial\boldsymbol{\omega}\partial\boldsymbol{\omega}^T}\log p(\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\omega}|\boldsymbol{\psi})|_{\boldsymbol{\omega}=\boldsymbol{\omega}^*}.$$

For the low-rank SHP model, there is no analytical form for the Hessian matrix. Instead, we use a numerical solution of the Hessian matrix in the `optim` function in R.

5.4.3 Estimation of function of volatility

If $\boldsymbol{\psi}$ were known, a function $f(\cdot)$ of $\boldsymbol{\omega}$ at observed locations can be estimated as the conditional expectation $\mathrm{E}[f(\boldsymbol{\omega})|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi}]$, given by

$$\begin{aligned} \mathrm{E}[f(\boldsymbol{\omega})|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi}] &= \int f(\boldsymbol{\omega})p(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})d\boldsymbol{\omega} \\ &= \int f(\boldsymbol{\omega})\frac{p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)}{p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\psi})}d\boldsymbol{\omega} \\ &= \frac{\int f(\boldsymbol{\omega})p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)d\boldsymbol{\omega}}{\int p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)d\boldsymbol{\omega}} \\ &= \frac{\mathrm{E}_a[f(\boldsymbol{\omega})p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)/p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})]}{\mathrm{E}_a[p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\theta})p(\boldsymbol{\omega}|\phi_\alpha)/p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\psi})]}. \end{aligned} \tag{5.23}$$

If f is the identity function, we are interested in estimating $\boldsymbol{\alpha}$. Once the estimates of the parameters $\hat{\boldsymbol{\psi}}$ are obtained, we can sample $\boldsymbol{\omega}^{(1)}, ..., \boldsymbol{\omega}^{(N)}$ from $p_a(\boldsymbol{\omega}|\boldsymbol{y}, \boldsymbol{y}', \hat{\boldsymbol{\psi}})$ and get estimate $\hat{\boldsymbol{\omega}}$ through (5.23) using a Monte Carlo approximation. Then $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}'$ can be obtained by

$$\begin{aligned} \hat{\boldsymbol{\alpha}} &= \hat{B}\hat{\boldsymbol{\omega}} \\ \hat{\boldsymbol{\alpha}}' &= \hat{B}'\hat{\boldsymbol{\omega}}. \end{aligned} \tag{5.24}$$

5.4.4 Empirical best predictor (EBP) for y

The prediction of a response at an unobserved location $\boldsymbol{x}_0$ is obtained based on the low-rank SHP model. Given the latent process $\boldsymbol{\omega}$, the joint distribution of $p(\boldsymbol{y}, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\psi})$ is multivariate normal. Therefore, the conditional distribution of y_0 at unobserved location $\boldsymbol{x}_0$ given the observation vector $(\boldsymbol{y}, \boldsymbol{y}')$ is heteroscedastic Gaussian with mean

$$E(y_0|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\omega}, \boldsymbol{\psi}) = \boldsymbol{g}(\boldsymbol{x}_0)^T\boldsymbol{\beta} + \boldsymbol{r}_y(\boldsymbol{x}_0, \boldsymbol{x})R_{y^*}^{-1}(\boldsymbol{y}^* - \boldsymbol{\mu}^*), \tag{5.25}$$

and variance

$$\mathrm{Var}(y_0|\boldsymbol{y}, \boldsymbol{y}', \boldsymbol{\omega}, \boldsymbol{\psi}) = \sigma^2(1 - \boldsymbol{r}_y(\boldsymbol{x}_0, \boldsymbol{x})R_{y^*}^{-1}\boldsymbol{r}_y(\boldsymbol{x}, \boldsymbol{x}_0)), \tag{5.26}$$

where $r_y(\boldsymbol{x}_0, \boldsymbol{x})$ is the conditional covariance vector between y_0 and $(\boldsymbol{y}, \boldsymbol{y}')$ and

$$\begin{aligned} \mathrm{Cov}(y_0, \boldsymbol{y}|\boldsymbol{\omega}, \boldsymbol{\psi}) &= \sigma^2 \exp(\tau B_0\boldsymbol{\omega}/2) r_z(\boldsymbol{x}_0, \boldsymbol{x}) \mathrm{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}, \\ \mathrm{Cov}(\boldsymbol{y}_0, \boldsymbol{y}'|\boldsymbol{\omega}, \boldsymbol{\psi}) &= \sigma^2 \exp(\tau B_0\boldsymbol{\omega}/2)(r_z(\boldsymbol{x}_0, \boldsymbol{x})\mathrm{diag}\{B\boldsymbol{\omega}'\} \\ &\quad + r_{zz'}(\boldsymbol{x}_0, \boldsymbol{x}))\mathrm{diag}\left\{\exp\left(\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}. \end{aligned}$$

To account for the uncertainty in estimating $\boldsymbol{\omega}$, we seek the best predictor $E(y_0|\boldsymbol{y}, \boldsymbol{\psi})$ and its predictive variance $\mathrm{Var}(y_0|\boldsymbol{y}, \boldsymbol{\psi})$. If we write $y^* =$

$(\boldsymbol{y}, \boldsymbol{y}')^T$, $R_{y^*} = \text{diag}\{\exp(\tau B\boldsymbol{\omega}/2)\} R^*_{y^*} \text{diag}\{\exp(\tau B\boldsymbol{\omega}/2)\}$ and $r_y(\boldsymbol{x}_0, \boldsymbol{x}) = \sigma^2 \exp(\tau B_0\boldsymbol{\omega}/2) r^*_y(\boldsymbol{x}_0, \boldsymbol{x}) \text{diag}\{\exp(\tau B\boldsymbol{\omega}/2)\}$, the best predictor can be written as

$$\begin{aligned} & E(y_0|\boldsymbol{y}^*, \boldsymbol{\psi}) \\ = \; & E\left[E(y_0|\boldsymbol{y}^*, \boldsymbol{\omega}, \boldsymbol{\psi})|\boldsymbol{y}^*, \boldsymbol{\psi}\right] \\ = \; & E\left[g(\boldsymbol{x}_0)^T\boldsymbol{\beta} + \exp(\tau B_0\boldsymbol{\omega}/2)\boldsymbol{r}^*_y(\boldsymbol{x}_0, \boldsymbol{x}) R^{*\,-1}_{y^*} \text{diag}\left\{\exp\left(-\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}\right. \\ & \left. \times(\boldsymbol{y}^* - \boldsymbol{\mu}^*)|\boldsymbol{y}^*, \boldsymbol{\psi}\right], \end{aligned} \tag{5.27}$$

where $r^*_y(\boldsymbol{x}_0, \boldsymbol{x}) = (r_z(\boldsymbol{x}_0, \boldsymbol{x}), r_z(\boldsymbol{x}_0, \boldsymbol{x})\text{diag}\{B\boldsymbol{\omega}'\} + r_{zz'})^T$ and

$$R^*_{y^*} = \left(\begin{bmatrix} G^T \\ G'^T \end{bmatrix}\boldsymbol{\beta}, \sigma^2 \begin{bmatrix} R^*_{yy} & R^*_{yy'} \\ R^*_{y'y} & R^*_{y'y'} \end{bmatrix}\right),$$

with

$$\begin{aligned} R^*_{yy} &= R_{zz} \\ R^*_{yy'} &= \left(\frac{\tau}{2} R_{zz}\text{diag}\{B'\boldsymbol{\omega}\} + R_{zz'}\right) \\ R^*_{y'y} &= \left(\frac{\tau}{2}\text{diag}\{B'\boldsymbol{\omega}\} R_{zz} + R_{z'z}\right) \\ R^*_{y'y'} &= \left(\frac{\tau^2}{4}\text{diag}\{B'\boldsymbol{\omega}\} R_{zz} \text{diag}\{B'\boldsymbol{\omega}\} + \frac{\tau}{2}\text{diag}\{B'\boldsymbol{\omega}\} R_{zz'}\right. \\ & \quad \left. + \frac{\tau}{2} R_{z'z}\text{diag}\{B'\boldsymbol{\omega}\} + R_{z'z'}\right). \end{aligned} \tag{5.28}$$

The predictive variance is

$$\text{Var}(y_0|\boldsymbol{y}^*, \boldsymbol{\psi}) = E\left[\text{Var}(y_0|\boldsymbol{y}^*, \boldsymbol{\omega}, \boldsymbol{\psi})|\boldsymbol{y}^*, \boldsymbol{\psi}\right] + \text{Var}\left[E(y_0|\boldsymbol{y}^*, \boldsymbol{\omega}, \boldsymbol{\psi})|\boldsymbol{y}^*, \boldsymbol{\psi}\right]. \tag{5.29}$$

where

$$\begin{aligned} & E\left[\text{Var}(y_0|\boldsymbol{y}^*, \boldsymbol{\omega}, \boldsymbol{\psi})|\boldsymbol{y}^*, \boldsymbol{\psi}\right] \\ = \; & \sigma^2 E\left[\exp(\tau B_0\omega)(1 - r^*_y(\boldsymbol{x}_0, \boldsymbol{x}) R^{*\,-1}_{y^*} r^*_y(\boldsymbol{x}_0, \boldsymbol{x})^T)|\boldsymbol{y}^*, \boldsymbol{\psi}\right] \end{aligned} \tag{5.30}$$

and

$$
\begin{aligned}
& \mathrm{Var}\left[E(y_0|\boldsymbol{y}^*,\boldsymbol{\omega},\boldsymbol{\psi})|\boldsymbol{y}^*,\boldsymbol{\psi}\right] \\
= \; & E\left[\left(E(y_0|\boldsymbol{y}^*,\boldsymbol{\omega},\boldsymbol{\psi}) - E(y_0|\boldsymbol{y}^*,\psi)\right)^2|\boldsymbol{y}^*,\boldsymbol{\psi}\right] \\
= \; & E\left[\exp(\tau B_0\boldsymbol{\omega}) r_y^*(\boldsymbol{x}_0,\boldsymbol{x}) {R_{y^*}^*}^{-1} \mathrm{diag}\left\{\exp\left(-\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\}(\boldsymbol{y}^*-\boldsymbol{\mu}^*)(\boldsymbol{y}^*-\boldsymbol{\mu}^*)^T\right. \\
& \left.\times \mathrm{diag}\left\{\exp\left(-\frac{\tau B\boldsymbol{\omega}}{2}\right)\right\} R_{y^*}^{*-1} r_y^*(\boldsymbol{x}_0,\boldsymbol{x})^T|\boldsymbol{y}^*,\boldsymbol{\psi}\right] \\
& - \left[E(y_0|\boldsymbol{y}^*,\psi) - g(\boldsymbol{x}_0)\boldsymbol{\beta}\right]^2. \qquad (5.31)
\end{aligned}
$$

Since the best predictor of y_0 (5.27) and its predictive variance (5.29) are functions of $\boldsymbol{\omega}$, we can compute them using (5.23) through Monte Carlo integration. In practice, we plug in the maximum likelihood estimates of $\boldsymbol{\psi}$ into (5.23). We refer to this predictor as the *empirical best predictor* (EBP) for y_0.

5.5 Application

In this section, SHP modeling of responses and derivatives is illustrated through two 1-dimensional test functions and two 2-dimensional test functions. The SHP prediction performance is compared with GP model and HOPS approximation. At the end, we revisit the SIR model. The Gaussian covariance function is used in the α and Z processes in all examples.

5.5.1 1-d test functions

We use two 1-d test functions as examples to compare the stochastic modeling of response with and without derivatives. The first function is

$$f = 2\cos(7\pi x/2)e^{-3x}, \quad x \in [0,2],$$

which is rescaled by a factor of 2 from the original function in Santer et al. (2003). The left panel in Figure 5.5 is the true function and the right panel is the corresponding true derivative. The true function is based on 200 equally-spaced data

points on $[0, 2]$. We consider sample sizes of 5, 10, 15 and 20 regularly-spaced points out of 200 as training data. The response and derivative are computed at each observed location. Six different models are fitted with the training data points and out-of-sample RMSE is computed for each model. The six models are categorized into two groups. One group is modeling response without derivative, including the GP model with Gaussian covariance function and the SHP model. The other group is modeling response with derivative, including the GP model with Gaussian covariance function (GP.old), the GP model with SHP unconditional covariance function as a new covariance function (GP.new), the SHP model and HOPS approximation.

Table 5.1 lists the RMSE for different models with different sample sizes. In general, the SHP model outperforms the GP model with or without derivative information. For example, the RMSEs for the SHP model with and without derivatives are 0.151 and 0.0028 when n is 10, which are 35% and 54% smaller than those with the GP model.

When the sample size is small, neither GP and SHP modeling with responses only reconstruct the curve well. The left panel in Figure 5.6 is the fitted curves by the GP and SHP models, based only on 5 observations. Neither of them has a good fitted curve. The right panel plots the fitted curves by modeling with derivative information. Even though the fitted curve is not perfect, the derivative does give local slope information and helps the local prediction of the function shape. This is also the reason why the HOPS approximation can do a comparable job to stochastic modeling for a very small sample size. As the sample size increases, stochastic modeling is more flexible in capturing the curvature of the function than is the HOPS linear approximation.

As we discussed in the previous section, the unconditional covariance function for SHP and its derivative have unique characteristics and can be used as a new class of covariance functions in a GP model. The RMSEs for the GP model with this

new covariance function are the same or better than the RMSEs for the GP model with Gaussian covariance function. This confirms the potential of the unconditional SHP covariance function as a new class of covariance functions.

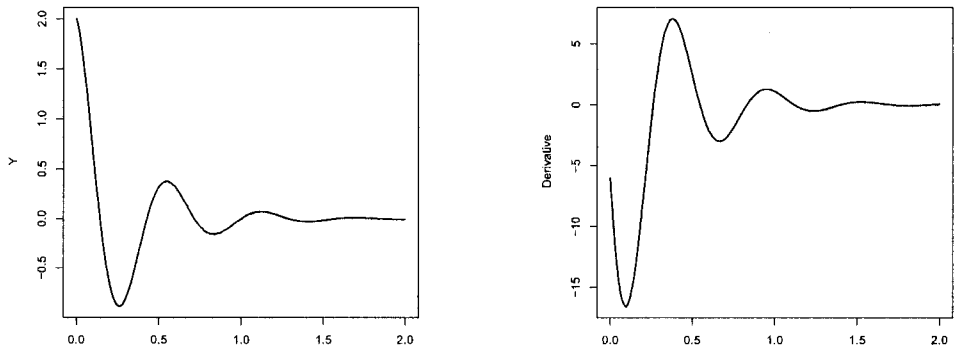


Figure 5.5: First 1-d test function. Left panel: true curve. Right panel: derivative curve.

Table 5.1: RMSE for the first 1-d test function under different models and sample sizes. GP.old is the GP model using Gaussian covariance function. GP.new is the GP model using unconditional covariance function of SHP and its derivative as a new covariance function.

	w/o derivative		with derivative			
	GP	SHP	GP.old	GP.new	SHP	HOPS
$n = 5$	0.670	0.579	0.328	0.328	0.223	0.350
$n = 10$	0.229	0.151	0.0060	0.0060	0.0028	0.145
$n = 15$	0.017	0.0038	0.0016	0.0013	0.00038	0.055
$n = 20$	0.0049	0.0034	0.00050	0.00042	0.00014	0.031

The second test function is from Xiong et al. (2007):

$$f(x) = \sin(30(x - 0.9)^4)\cos(2(x - 0.9)) + (x - 0.9)/2, \quad x \in [0, 1].$$

The left panel in Figure 5.7 is the true function and right panel is the corresponding true derivative curve. The true function is based on 200 equally-spaced data points

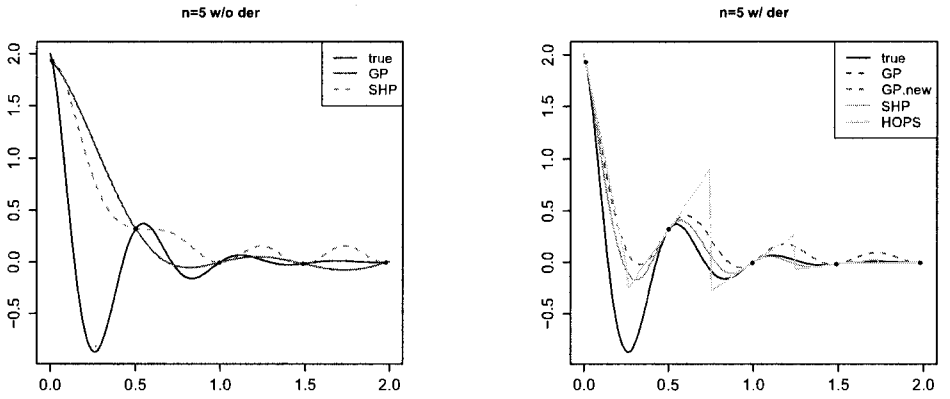


Figure 5.6: Fitted curve and derivative for the first 1-d test function using different models.

on $[0, 1]$. We consider sample sizes of 12, 18, and 24 regularly-spaced points out of 200 as training data. The sample sizes are greater than with the second function since it is rougher and more inhomogeneous. Again, six different models are fitted with the training data points and out-of-sample RMSE is computed for each model.

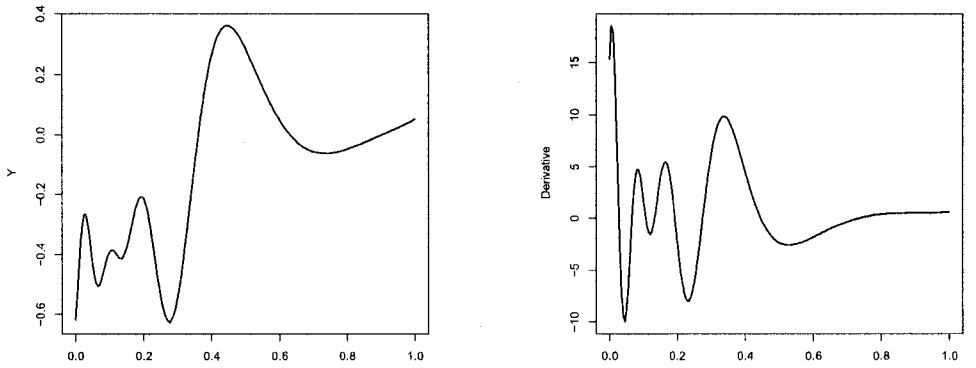


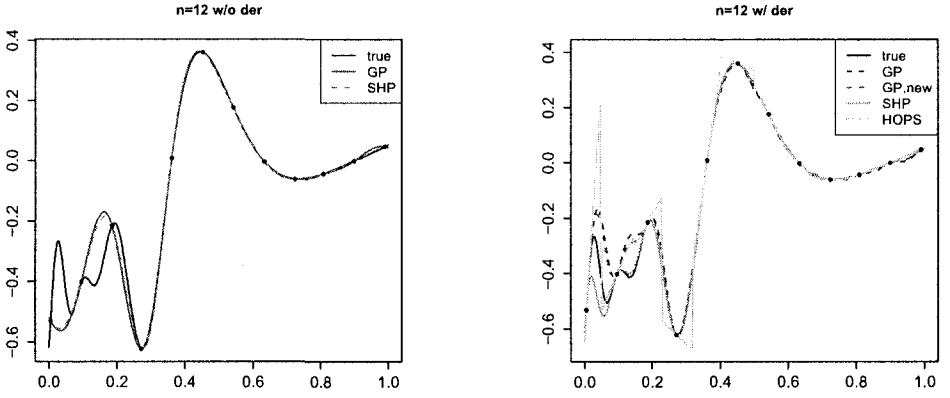
Figure 5.7: Second 1-d test function. Left panel: true curve. Right panel: derivative curve.

Table 5.2 lists the RMSEs for different models with different sample sizes. Similar results are obtained as with the first test function. SHP model is more able to capture the local behavior than GP model. The stochastic models outperform the

HOPS approximation. Figure 5.5.1 plots the fitted curves for modeling with and without derivatives at sample size 12.

Table 5.2: RMSE for the second 1-d test function under different models and sample sizes. GP.old is the GP model using Gaussian covariance function. GP.new is the GP model using unconditional covariance function of SHP and its derivative as a new covariance function.

	w/o derivative		with derivative			
	GP	SHP	GP.old	GP.new	SHP	HOPS
$n = 12$	0.066	0.058	0.050	0.042	0.028	0.078
$n = 18$	0.046	0.045	0.011	0.011	0.0078	0.037
$n = 24$	0.032	0.026	0.0022	0.0022	0.0010	0.013



caption[Fitted curve and derivative for the second 1-d test function.] Fitted curve and derivative for the second 1-d test function using different models.

As we can see from Figure 5.5.1, the second function has varying volatility over $[0, 1]$. The function in region $[0, 0.3]$ is more volatile than it is in region $[0.3, 1]$. Table 5.2 shows that the SHP model outperforms the traditional GP model and HOPs approximation globally. It is also of interest to investigate the performance of predictors locally, for sub-regions in $[0, 0.3]$ and $[0.3, 1]$. For the training data of size 12, 18 and 24, the RMSEs are listed in Table 5.3. In this example, the SHP predictions outperforms GP and HOPS not only globally but also locally.

Table 5.3: RMSE for the second 1-d test function in subregion $[0, 0.3]$ and $[0.3, 1]$ under different models and sample sizes. GP.old is the GP model using Gaussian covariance function. GP.new is the GP model using unconditional covariance function of SHP and its derivative as a new covariance function.

		w/o derivative		with derivative			
		GP	SHP	GP.old	GP.new	SHP	HOPS
$[0, 0.3]$	n=12	0.120	0.121	0.091	0.077	0.051	0.131
	n=18	0.084	0.083	0.019	0.019	0.014	0.065
	n=24	0.059	0.047	0.0039	0.0039	0.0019	0.0232
$[0.3, 1]$	n=12	0.0046	0.0013	0.0075	0.0080	7.94e-05	0.0373
	n=18	0.0035	0.0011	0.0003	0.0003	1.64e-05	0.0095
	n=24	0.0013	0.0003	0.0003	0.0003	7.90e-05	0.0232

One further observation can be made from Table 5.1. The RMSE for SHP modeling with derivative at sample size 5 is comparable to the RMSE of SHP modeling without derivative at sample size 10; the RMSE for SHP modeling with derivative at sample size 10 is comparable to the RMSE of SHP modeling without derivative at sample size 20. Similar observations can be made for Table 5.2: adding derivative information is like doubling the response-only sample size.

A question arises from the above observations: Shall we use more responses without derivatives in modeling or less responses but with derivatives in modeling? There is no universal answer for this question. We need to first compare the cost of producing response and derivatives. Even though the cost for derivatives is in general less than response, obtaining derivatives can also be expensive for an expensive forward model, such as some climate models with a large number of parameters. Modeling with derivative is more complicated and time consuming. Therefore, for a standard expensive computer experiment, modeling with more responses may be preferred. On the other hand, derivative provides useful local information about the curvature, modeling with derivative is preferred if we have some background information about the volatility of the underlying physical process.

The derivative can be used in adaptive sampling. The FAPS method proposed by Estep and Neckels (2006) is an adaptive sampling method using derivative information. One important byproduct of the FAPS method is the sensitivity analysis of input variables. This allows model dimensionality reduction and model evaluation in a reduced input space by sampling more responses only in sensitive directions, with other non-effective inputs fixed at some reference values.

5.5.2 2-d test functions

For a 2-dimensional example, the low-rank SHP model for response and derivatives is

$$\begin{aligned} \boldsymbol{y} &= G^T\boldsymbol{\beta} + \sigma \operatorname{diag}\{\exp(\tau B\boldsymbol{\omega}/2)\}\, \boldsymbol{Z} \\ \boldsymbol{y}_1' &= G_1'^T\boldsymbol{\beta} + \sigma \operatorname{diag}\{\exp(\tau B\boldsymbol{\omega}/2)\} \left\{\frac{\tau}{2}\operatorname{diag}\{B_1'\boldsymbol{\omega}\}\boldsymbol{Z} + \boldsymbol{Z}_1'\right\} \\ \boldsymbol{y}_2' &= G_2'^T\boldsymbol{\beta} + \sigma \operatorname{diag}\{\exp(\tau B\boldsymbol{\omega}/2)\} \left\{\frac{\tau}{2}\operatorname{diag}\{B_2'\boldsymbol{\omega}\}\boldsymbol{Z} + \boldsymbol{Z}_2'\right\} \end{aligned} \tag{5.32}$$

where $B_1' = \partial B/\partial x_1$ and $B_2' = \partial B/\partial x_2$. The parameter estimation and prediction is based on this low-rank model.

The first 2-dimensional function is given by Paciorek (2003):

$$f(x_1, x_2) = 1.9(1.35 + e^{x_1}\sin(13(x_1 - 0.6)^2)e^{-x_2}\sin(7x_2)), \quad x_1, x_2 \in [0, 1].$$

The true surface is based on 21×21 grid points on $[0, 1] \times [0, 1]$ and is plotted in Figure 5.9(a).

To compare the prediction performance of SHP with GP and HOPS approximation, the six models that are used in the 1-dimensional application are again used here to fit the training data. We use maximin LHS design to sample 10 points over the input domain as training locations. The sampling was implemented through the package `lhs` in `R` (Team (2005)). After fitting the responses $\boldsymbol{y}$ with and without derivatives $(\boldsymbol{y}_1', \boldsymbol{y}_2')$, we predict 21×21 grid points and compute the RMSE. The

process of sampling, fitting and predicting was repeated 20 times. We then compute the 20 RMSE ratios of GP/SHP with and without derivatives. The summary statistics of the 20 RMSE ratios are given in Table 5.4. When modeling response without derivative, the SHP model performs similarly to the GP model with such a small sample size. When modeling response with derivatives, the SHP model has smaller RMSE in 18 out of 20 trials compared with the GP model. The boxplots of RMSEs in Figure 5.8 gives a graphical comparison of the performance of GP, SHP and HOPS for prediction with and without derivatives in the model.

We give an example of fitted surfaces in Figure 5.9 (b), (c) and (d). The plots show that the derivative information helps the SHP model to catch peaks and valleys. HOPS approximation does not perform well even compared with stochastic modeling without derivatives. This is because HOPS uses linear approximation and the true function is nonlinear. HOPS method is a very cheap approximation method for computer outputs, but requires more data points to achieve satisfactory prediction accuracy.

Table 5.4: Summary statistics of 20 RMSE ratios for the first 2-d test function with sample size 10 under different models. The last column is the percentage of RMSE ratios being greater than 1 out of 20.

	min	25^{th}	median	mean	75^{th}	max	percentage
GP/SHP	0.664	0.926	1.00	1.00	1.08	1.30	50
GP/SHP (w/der)	0.860	1.059	1.100	1.130	1.206	1.536	90

The second function is the two-dimensional example $f(x_1, x_2) = 10x_1 \exp(-x_1^2 - x_2^2)$ used in Chapter 3. This function shows more inhomogeneous behavior than the first 2-dimensional test function. We already know from previous results that SHP has a better prediction accuracy than a GP when modeling without derivatives. In this section, we want to evaluate how SHP performs when modeling with derivatives.

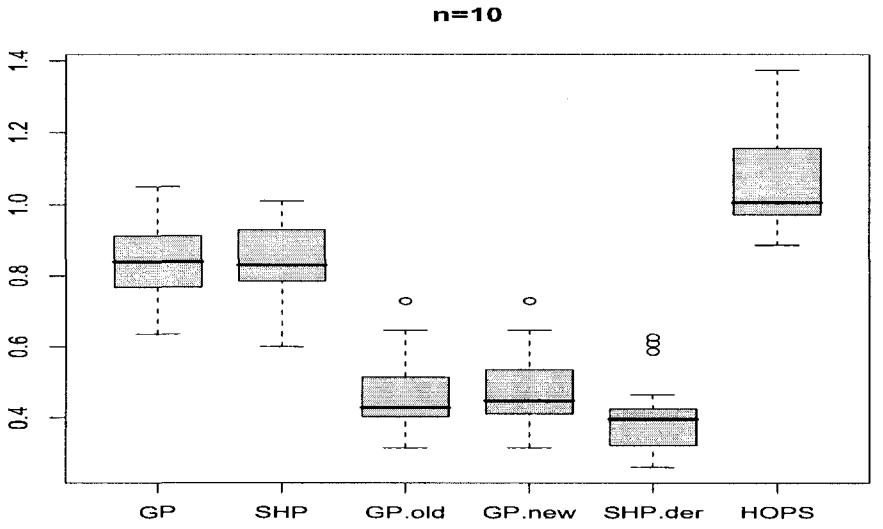


Figure 5.8: RMSE Boxplots over 20 replicates for the first 2-d test function with sample size 10 for modeling with and without derivatives.

The true surface is again based on 21 × 21 grid points on $[-2, 6] \times [-2, 6]$. Responses $\boldsymbol{y}$ and the first partial derivatives $\boldsymbol{y}'_1, \boldsymbol{y}'_2$ at 12 sampled locations are used as training data. We use LHS to place 6 points on the first quadrant $[-2, 2] \times [-2, 2]$ and 6 points on other areas. The sampling was implemented through the `R` package `tgp` (Gramacy (2007)). The six models in previous examples are used again to fit the training data and the RMSE of predicted values at 21 × 21 grid points is computed. The process of sampling, model fitting and prediction was repeated 20 times. The summary statistics of the 20 RMSE ratios are given in Table 5.5. When modeling response without derivative, the SHP model performs similarly to the GP model with such a small sample size. When modeling response with derivative, the SHP model has smaller RMSE in 14 out of 20 trials comparing with the GP model. The boxplots of RMSEs in Figure 5.10 gives a graphical comparison of the performance of GP, SHP and HOPS for prediction with and without derivatives in the model.

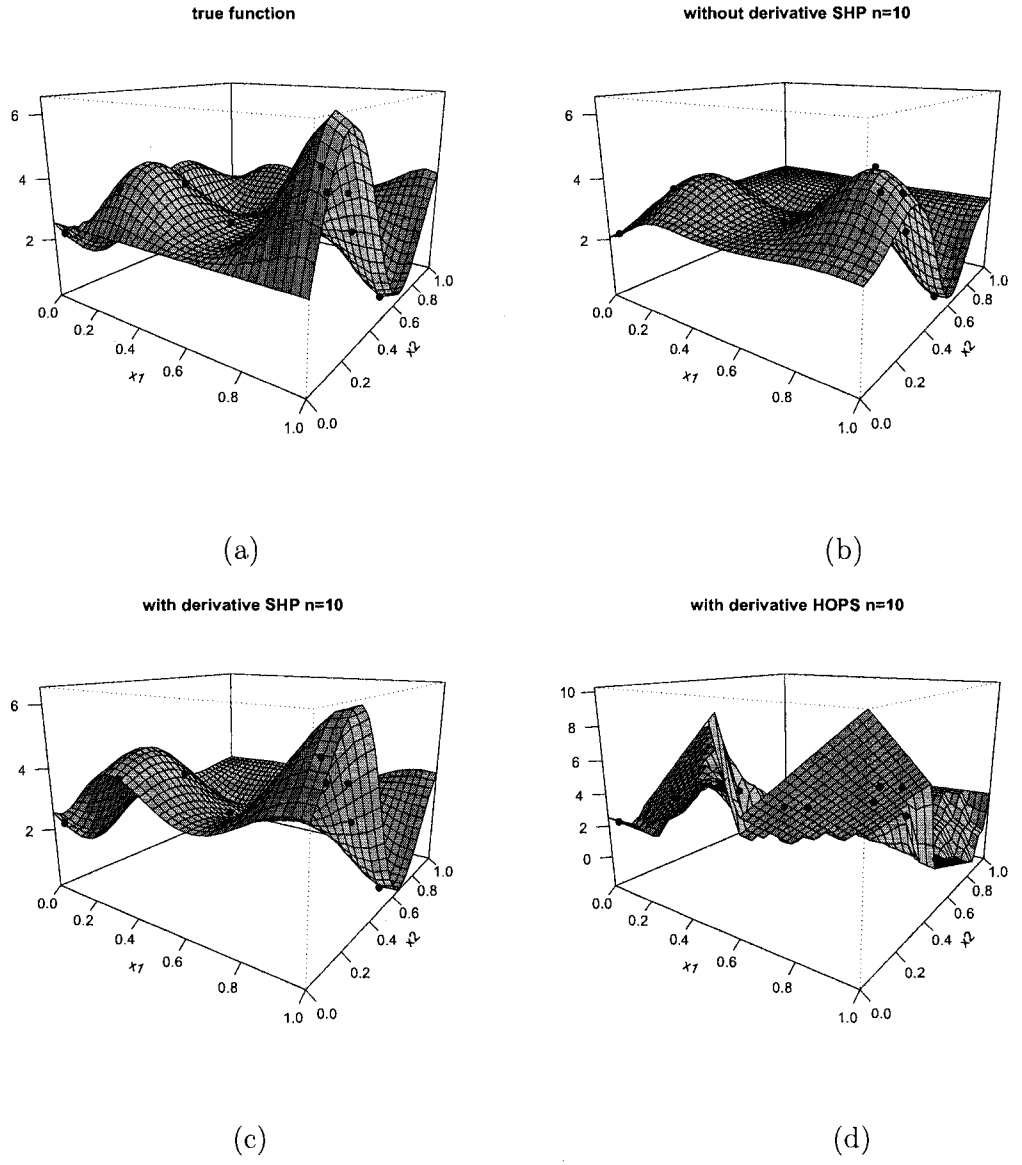


Figure 5.9: First 2-d test function and fitted surfaces. (a) The true function. (b) SHP model fitted surface with responses only. (c) SHP fitted surface with responses and derivatives. (d) HOPS approximated surface.

5.5.3 SIR model revisited

Using the generalized Green's function and a variational analysis, Estep and Neckels (2006) compute not only the quantity of interest but also the derivatives at sampled input points. This derivative information is used in Estep and Neckels

Table 5.5: Summary statistics of 20 RMSE ratios for the second 2-d test function with sample size 12 under different models. The last column is the percentage of RMSE ratios being greater than 1 out of 20.

	min	25^{th}	median	mean	75^{th}	max	percentage
GP/SHP	0.407	0.915	1.029	1.003	1.116	1.429	60
GP/SHP (w/der)	0.548	0.932	1.267	1.418	1.601	3.271	70

(2006) to create what they refer to as FAPS, to adaptively sample the data points over input space and then use HOPS to approximate the quantity of interest at untried locations. We want to compare the prediction performance of GP and SHP to the HOPS method fitting with FAPS sample points.

FAPS in computer experiments

Estep and Neckels (2006) pointed out in their study that FAPS not only gies an adaptive sampling method, placing more points in the region of large error, but also provides an important byproduct that is the sensitivity analysis of input variables.

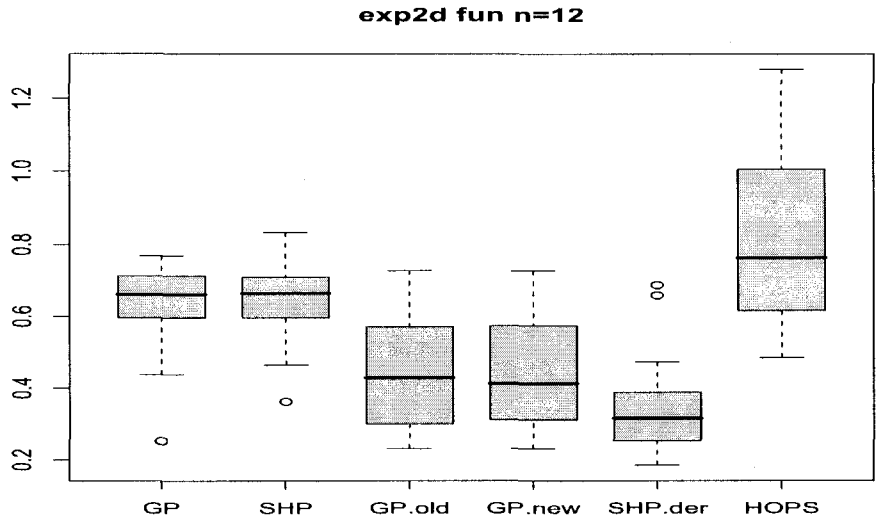


Figure 5.10: Boxplots of RMSE over 20 replicates for the second 2-d test function with sample size 12.

Since most computer models are systems of differential equations, FAPS can be used in design and sensitivity study of computer experiments.

The FAPS method was implemented in the SIR model. The 100 FAPS adaptively sampled points for three quantities of interest in SIR model are plotted in Figure 5.11, Figure 5.12 and Figure 5.13. The splitting direction of these FAPS sample points provides a measure of sensitivity to the input variables for the quantities of interest. For the average susceptible population $q1$, the important factors are contraction rate r_I and death rate from disease d_I. This is clear from Figure 5.11 because nearly all splitting in FAPS occur on these input dimensions. For the average infected population $q2$, the important factors are recovery rate a_R, natural growth rate r_n, natural death rate d_n and death rate from disease d_I. For the average resistant population $q3$, the important factors are recovery rate a_R, natural growth rate r_n, natural death rate d_n and death rate from disease d_I. Though the important factors are different for different quantities of interest, the probability of inheriting resistance p_R and carrying capacity k are not important for any of them: FAPS never splits along these dimensions.

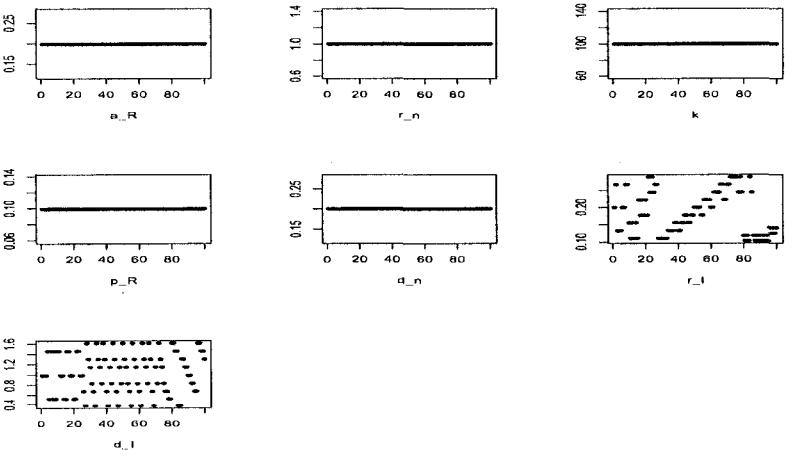


Figure 5.11: Scatter plots of 100 FAPS points on each dimension for $q1$ variable. The x axis is the index for the sequence of sampled points.

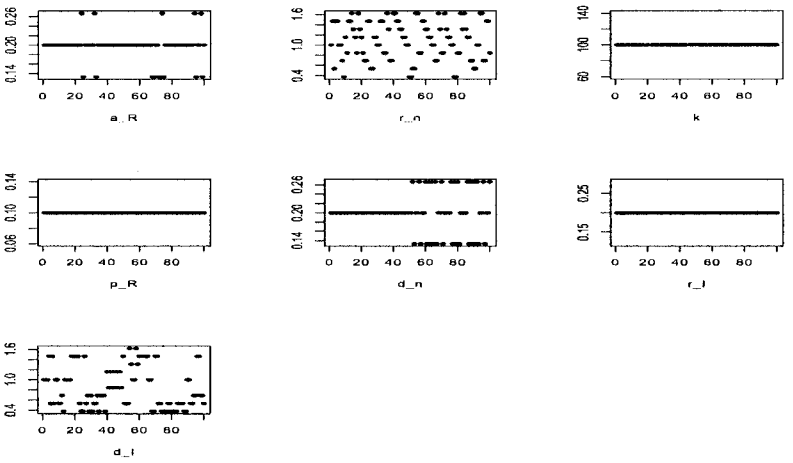


Figure 5.12: Scatter plots of 100 FAPS points on each dimension for $q2$ variable. The x axis is the index for the sequence of sampled points.

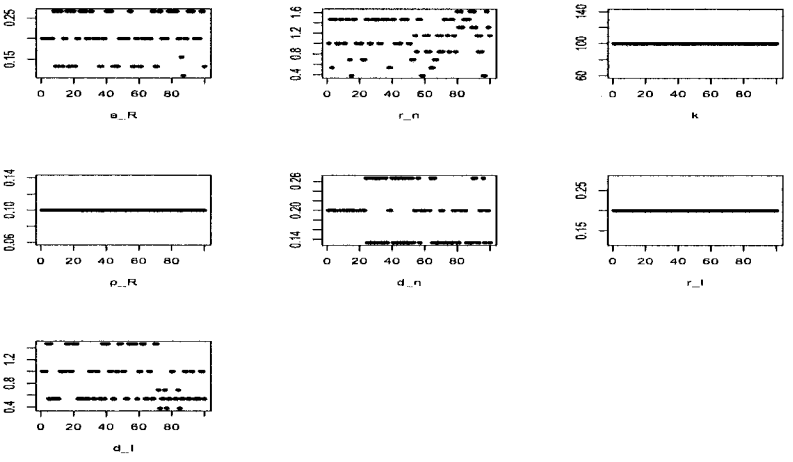


Figure 5.13: Scatter plots of 100 FAPS points on each dimension for $q3$ variable. The x axis is the index for the sequence of sampled points.

The above results confirm the two applications of FAPS in computer experiments. One is that FAPS can be used in computer experiments as an adaptive sampling method since most mathematical models implemented in computer experiments are systems of differential equations. The other application is the sensitivity results offered by FAPS, which can be used in dimension reduction in computer experiments. For example, we can fix carrying capacity and the probability of in-

heriting resistance in SIR model at their mean values and vary the other 5 variables. The disease study can be reduced from 7-dimensional space to 5-dimensional space and SIR model evaluation can be performed in the effective 5-d space. This will reduce the cost and time to help management decision for further disease control.

The quantity of interest in Estep and Neckels (2006) is the average infected population $q2$. In following subsections, we want to use the reduced 5-dimensional SIR model as an example to explore the performance of SHP modeling with derivatives. To compare the performance of SHP and GP with HOPS methods, a test data set of 1000 data points in 5-d space was sampled by maximin distance LHS design. We fit three models with FAPS points and predict $q2$ at test locations. The RMSEs are computed and are used to compare the prediction performance.

Simultaneously modeling with responses and derivatives

We fit responses and derivatives simultaneously at 10 and 20 FAPS sampled locations with HOPS approximation, GP model and SHP model and predict the average infected population at 1000 test locations in 5-dimensional space.

Table 5.6 lists the RMSEs for the $q2$ variable with different models. The SHP model performs better than the GP model in terms of smaller RMSE. The results support the flexibility of SHP and its derivative process. The stochastic modeling with derivatives has smaller RMSE than the HOPS approximation. The RMSEs for the GP and SHP models are 26% and 40% less than the RMSE for the HOPS approximation at sample size 10. The RMSEs for the GP and SHP models are 19% and 41% less than the RMSE for the HOPS approximation at sample size 20. Even though the HOPS is cheaper to implement than stochastic modeling, it requires more training data to achieve good prediction performance. For an expensive computer experiment, the computational cost for stochastic modeling may be negligible.

Table 5.6: RMSE for $q2$ variable at 1000 test locations using 10 and 20 FAPS sampled points as training data sets. The responses and derivatives are simultaneously fitted with HOPS, GP and SHP models.

	HOPS	GP	SHP
$n = 10$	0.821	0.607	0.492
$n = 20$	0.705	0.570	0.414

Two-stage modeling with responses and derivatives

As sample size n increases, the data vector combining response and first-order partial derivatives increases quickly, especially for computer experiments with a high-dimensional (d-dimensional) input space. When using stochastic modeling with derivatives, the joint covariance function for response and derivatives has size $n(d+1) \times n(d+1)$. Numerical challenges, such as computing the determinant of the covariance function, occur in the optimization process of maximum likelihood estimation. Therefore, obtaining parameter estimates for larger sample sizes may be difficult or impossible.

To avoid this problem, we develop a two-stage approach for using response and derivatives in model fitting. At the first stage, parameter estimates are obtained based only on responses, which is the situation we described in Chapter 2. The derivative information is then used in the prediction at the second stage with parameter estimates from the first stage. We illustrate the methodology for $q2$ in the 5-d SIR model. We implement the two-stage approach with GP and SHP models fitted with FAPS sampled data sets of size 30 to 70.

The RMSE for 1000 test locations of different models with different sample sizes are listed in Table 5.7. Both stochastic modeling approaches beat the HOPS approximation. Not shown in the table is the fact that RMSEs for the GP and SHP at sample size 30 are comparable with the HOPS approximation at sample size 100 or even 200. The SHP model performs better than the GP model at different sample

sizes in this 5-d example. The cost of this increase efficiency is that the SHP model takes longer to implement than the GP model.

Table 5.7: RMSE for $q2$ variable using different models and using FAPS sampled points as training data. For GP and SHP model fitting, the two-stage prediction strategy is used to avoid numerical problems in optimization process: response-only data are used for model fitting, and derivatives are included for prediction.

FAPS	30	40	50	60	70
HOPS	0.561	0.504	0.494	0.458	0.461
GP+der	0.377	0.352	0.347	0.313	0.294
SHP+der	0.322	0.290	0.276	0.256	0.249

5.5.4 Summary of modeling with derivative

From the above examples, we can see that the HOPS method in Estep and Neckels (2006) is not comparable to the GP and SHP models in prediction performance for small or moderate sample sizes. But their FAPS method can be introduced into design and analysis of computer experiments since most mathematical models implemented in the computer experiments are systems of differential equations. The procedure of stochastic modeling with FAPS data points is summarized as follows:

- Run FAPS m times to obtain an optimum sample.
- Use the splitting direction of FAPS sample points as sensitivity information for the d input variables. Choose the effective input dimensions $x_{e1}, x_{e2}, ..., x_{el}$ and fix the non-effective inputs at their reference values. The analysis of d-dimensional computer outputs is thus reduced to the analysis in l-dimensional space.
- Fit a stochastic model with FAPS sample in effective l-dimensional space.

To implement FAPS, we need to solve not only the forward system but also the adjoint problem. The adjoint problem is less expensive to execute since it is linear. However it also can be expensive if the forward system is expensive with a high-dimensional input space. So for an expensive model, it will cost much more to get m sampled data points from FAPS than from just solving the forward system m times. For a standard computer experiment, modeling with more responses may be preferred since modeling with derivatives is more complicated and time consuming.

However, the FAPS method offers an important by-product, the sensitivity analysis of input factors from the splitting directions of FAPS-sampled data. Meanwhile, the FAPS method has the potential to provide an initial set of data points in the design of computer experiments. Further sequential or adaptive sampling method of the GP and SHP models can be implemented in the reduced input space. This will save time and cost and improve prediction performance for computer experiments.

Chapter 6

CONCLUSIONS AND FUTURE WORK

The need for a more flexible and efficient metamodel in computer experiments outputs motivates the work in this dissertation. The deterministic computer response is modeled as a realization from a stochastic heteroskedastic process (SHP), a stationary non-Gaussian process with conditionally non-stationary covariance function. Comparing to the traditional Gaussian process models, the SHP model presents more flexibility in capturing the salient features of computer experiments and better quantification for guiding the next sampling point in an adaptive sampling scheme.

The contributions of this dissertation are grouped into Chapters $2-5$. Chapter 2 introduces the SHP model as a new metamodeling approach in modeling deterministic computer experiments. The unique properties of SHP correlation functions and sample paths are studied. By introducing a spatial stochastic latent process into the GP model, the SHP model produces the sample paths with greater variability and hence offers more modeling flexibility than those produced by a traditional GP model. We use maximum likelihood for inference, which is complicated by the high dimensionality of the latent process. Accordingly, we develop an importance sampling method for likelihood computation and use a low-rank kriging approximation to reconstruct the latent process. Responses at unobserved locations can be predicted using empirical best predictors or by empirical best linear unbiased predictors. Prediction error variances are also obtained. In examples with simulated and real computer experiment data in Chapter 3, the SHP model is superior to

traditional GP models. In addition, the SHP model can be used in an active learning context to select new locations that provide improved estimates of the response surface. Chapter 4 implements active learning via the SHP model, which appears to work better than other traditional approaches in several simulated examples.

For the motivating SIR model, we can obtain both responses and the first-order partial derivatives. Chapter 5 develops a low-rank SHP for modeling responses and derivatives. The model efficiency improves a lot by combining derivative into the response. A new importance sampling method was proposed for likelihood computation. In examples with simulated data and in the SIR example, the low-rank SHP model is superior to GP in modeling responses and derivatives.

Even though this study explores SHP model in many aspects, there is always more work that can be done in the future. The SIR model is a simple example of a computer experiment. It would be interesting to use SHP as a surrogate model in more complex computer experiments.

Another area of future research in computer experiments is model validation where data come from both a computer experiment and a "real" experiment. The SHP can be used to not only model computer code outputs, but also the bias term between computer experiment outputs and real experimental data.

The SHP model can be easily extended to a general regression context by adding measurement error terms. Palacios and Steele (2006) proposed a similar model in geostatistical modeling. Bayesian inference is performed in their study. But the prior distribution needs to be chosen carefully to improve the convergence and avoid the identification issues in parameter estimation. In computer experiments, a similar Bayesian approach may be adopted for the SHP model. If we have information on prior distribution, the Baysian approach of SHP model may solve the identification difficulties in parameter estimation.

The SHP model presents more flexibility than the GP model. Not surprisingly, the increased flexibility comes at the price of increased computational cost. There is,

however, a trade-off between computational cost and accuracy. Further exploration on improving the computational efficiency and accuracy is desired.

Bibliography

Banerjee, S., Carlin, B. P., and Gelfand, A. E. (2003). *Hierarchical Modeling and Analysis for Spatial Data.* Chapman and Hall/CRC.

Chen, V., Tsui, K., Barton, R., and Allen, J. (2003). A review of design and modeling in computer experiments. *Handbook of Statistics*, 22:231–261.

Chen, W. and Varadarajan, S. (1997). Integration of design of experiments and artificial neural network for achieving affordable concurrent design. *38th AIAA/ASME/ASCE/AHA/ASC Strctures, Structural Dynamics, and Materials Conference and AIAA/ASME/AHS Adaptive Structures Forum*, 2:1316–1324.

Cohn, D. A. (1996). Neural network exploration using optimal experimental design. *Advances in Neural Information Processing Systems*, pages 679–686. Morgan Kaufmann Publishers.

Cohn, D. A., Ghahramani, Z., and Jordon, M. J. (1996). Active learning with statistical models. *Journal of Artificial Intelligence Research*, 4:129–145.

Danielsson, J. and Richard, J. F. (1993). Accelerated Gaussian importance sampler with applications to dynamic latent variable models. *Journal of Applied Econometrics*, 8:153–173.

Davis, R. A. and Rodriguez-Yam, G. (2005). Estimation for state-space models: an approximate likelihood approach. *Statistica Sinica*, 15:381–406.

Durbin, J. and Koopmans, S. J. (1997). Monte Carlo maximum likelihood estimation for non-Gaussian state space models. *Biometrika*, 84:669–684.

Estep, D. and Neckels, D. (2006). Fast and reliable methods for determing the evolution of uncertain parameters in differential equations. *Journal of Computational Physics*, 213:530–556.

Fang, K. T., Li, R., and Sudjianto, A. (2006). *Design and Modeling for Computer Experiments.* Boca Raton, FL : Chapman and Hall/CRC.

Fuentes, M. and Smith, R. L. (2001). Modeling nonstationary processes as a convolution of local stationary processes. Technical report, North Carolina State University, Dept. of Statistics.

Gelfand, A. E., Kim, H. J., Sirmans, C. F., and Banerjee, S. (2003). Spatial modeling with spatially varying coefficient processes. *Journal of American Statistical Association*, 98:387–396.

Gramacy, R. B. (2005). *Bayesian Treed Gaussian Process Models.* PhD thesis, Univeristy of California, Santa Cruz, U.S.A.

Gramacy, R. B. (2007). tgp: An R package for Bayesian nonstationary, semiparametric nonlinear regression and design by treed Gaussian process models. *Journal of Statistical Software*, 19(9).

Gramacy, R. B., Lee, H. K. H., and Macready, W. G. (2004). Parameter space exploration with Gaussian process trees. *Proceedings of the 21st Internatioinal Conference on Machine Learning*, pages 353–360.

Higdon, D. (2002). Space and space-time modeling using process convolutions. *Quantitative Methods for Current Enviromental Issues*, pages 37–56. London: Springer-Verlag.

Higdon, D., Swall, J., and Kern, J. (1999). Non-stationary spatial modeling. In *Bayesian Statistics 6*, pages 761–768. Oxford: Oxford University Press. Bernardo, J. M., Berger, J. O., Dawid, A. P. and Smith, A. F. M. (Eds.).

Jin, R., Chen, W., and Simpson, T. W. (2000). Comparative studies of metamodeling techniques under multiple modeling criteria. *8th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization.* AIAA, Long Beach, CA, AIAA-2000-4801.

Kennedy, M. C. and O'Hagan, A. (2000). Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, 87:1–13.

Kennedy, M. C. and O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63:425–464.

Koehler, J. R. and Owen, A. B. (1996). Computer experiments. *Handbook of Statistics*, 13 (S.Ghosh and C.R.Rao(eds)):261–308.

Mackay, D. J. C. (1992). Information-based objective functions for active data selection. *Neural Computation*, 4:589–603.

Matérn, B. (1960). *Spatial Variation.* PhD thesis, Meddelanden fran Statens Skogsforskningsinstitut. Vol 49, Num. 5.

Matheron, G. (1963). Principles of geostatistics. *Economic Geology*, 58:1246–1266.

Mckay, M. D., Beckman, R. J., and Conover, W. J. (1979). The comparison of three methods for selecting values of input variables in the anlysis of output from a computer code. *Technometrics*, 21:239–245.

Metropolis, N. and Ulam, S. (1949). The Monte Carlo method. *Journal of the American Statistical Association*, 44:335–341.

Morris, M. D., Michtchell, T. J., and Ylvisaker, D. (1993). Bayesian design and analysis of computer experiments: Use of derivatives in surface prediction. *Technometrics*, 35:243–255.

Myers, R. H. and Montgomery, D. C. (1995). *Response Surface Methodology: Process and Product Optimization Using Designed Experiments.* John Wiley and Sons, Inc., New York, NY.

Paciorek, C. J. (2003). *Nonstationary Gaussian Processes for Regression and Spatial Modelling.* PhD thesis, Carnegie Mellon University, Pittsburgh, PA, U.S.A.

Palacios, M. B. and Steele, M. F. J. (2006). Non-Gaussian Bayesian geostatistical modeling. *Journal of the American Statistical Association*, 101:604–618.

Qian, Z., Seepersad, C., Joseph, R., Allen, J., and Wu, C. F. J. (2006). Building surrogate models with detailed and approximate simulations. *AMSE Journal of Mechanical Design*, 128:668–677.

Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods.* Springer-Verlag, New York, 2nd edition.

Ruppert, D., Wand, M. P., and Carroll, R. J. (2003). *Semiparametric Regression.* New York: Cambridge University Press.

Sacks, J., Schiller, S. B., and Welch, W. J. (1992). Design for computer experiments. *Technometrics*, 31:41–47.

Sacks, J. W., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and analysis of computer experiment. *Statistical Science*, 4:409–423.

Santer, T. J., Williams, B. J., and Notz, W. I. (2003). *The Design and Analysis of Computer Experiments.* Springer.

Satelli, A., Tarantola, S., and Chan, K. (1999). A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics*, 41:39–56.

Seo, S., Wallat, M., Graepel, T., and Obermayer, K. (2000). Gaussian process regression: Active data selection and test point rejection. *Proceedings of the International Conference on Neural Networks*, pages 241–246. IEEE.

Shephard, N. (1996). Statistical aspects of ARCH and stochastic volatility. In *Time Series Models in Econometrics, Finance and Other Fields*, pages 1–67. Chapman and Hall, London. In: Cox, D. R., Hinkley, D. V. and Barndorff-Nielsen, O. E. (Eds.).

Simpson, T. W., Lin, D. K. J., and Chen, W. (2001a). Sampling strategies for computer experiments: Design and analysis. *International Journal of Reliability and Applications*, 2(3):209–240.

Simpson, T. W., Peplinski, J. D., Koch, P. N., and Allen, J. K. (2001b). Metamodels for computer-based engineering design: survey and recommendations. *The Journal of Engineering with Computers, Special Issue Honoring Professor Steven J. Fenves*, 17:129–150.

Taylor, S. J. (1986). *Modelling Financial Time Series.* Chichester: John Wiley.

Team, R. D. C. (2005). R: A language and environment for statistical computing. R foundation for statistical computing, Vienna, Austria. ISBN 3-900051-07-0.

Wu, C. F. J. and Hamada, M. (2000). *Experiments: Planning, Analysis, and Parameter Design Optimization.* John Wiley, New York.

Xiong, Y., Chen, W., Apley, D., and X., D. (2007). A non-stationary covariance-based kriging method for metamodelling in engineering design. *International Journal for Numerical Methods in Engineer*, 71:733–756.

Yan, J. (2007). Spatial stochastic volatility for lattice data. *Journal of Agricultural, Biological, and Environmental Statistics*, 12(1):25–40.