Valid Estimation and Prediction Inference in Analysis of a Computer Model

by

Béla Nagy

B.Math, Eötvös Loránd University, 1995
Honours B.Math, Computer Science, University of Waterloo, 2000

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

The Faculty of Graduate Studies

(Statistics)

The University of British Columbia

(Vancouver)

August 2008

Abstract

Computer models or simulators are becoming increasingly common in many fields in science and engineering, powered by the phenomenal growth in computer hardware over the past decades. Many of these simulators implement a particular mathematical model as a deterministic computer code, meaning that running the simulator again with the same input gives the same output.

Often running the code involves some computationally expensive tasks, such as solving complex systems of partial differential equations numerically. When simulator runs become too long, it may limit their usefulness. In order to overcome time or budget constraints by making the most out of limited computational resources, a statistical methodology has been proposed, known as the "Design and Analysis of Computer Experiments".

The main idea is to run the expensive simulator only at a relatively few, carefully chosen design points in the input space, and based on the outputs construct an emulator (statistical model) that can emulate (predict) the output at new, untried locations at a fraction of the cost. This approach is useful provided that we can measure how much the predictions of the cheap emulator deviate from the real response surface of the original computer model.

One way to quantify emulator error is to construct pointwise prediction bands designed to envelope the response surface and make assertions that the true response (simulator output) is enclosed by these envelopes with a certain probability. Of course, to be able to make such probabilistic statements, one needs to introduce some kind of randomness. A common strategy that we use here is to model the computer code as a random function, also

known as a Gaussian stochastic process. We concern ourselves with smooth response surfaces and use the Gaussian covariance function that is ideal in cases when the response function is infinitely differentiable.

In this thesis, we propose Fast Bayesian Inference (FBI) that is both computationally efficient and can be implemented as a black box. Simulation results show that it can achieve remarkably accurate prediction uncertainty assessments in terms of matching coverage probabilities of the prediction bands and the associated reparameterizations can also help parameter uncertainty assessments.

Table of Contents

List of Tables

List of Figures

Acknowledgments

There was a time when I thought that statistics was boring. Driven by common misconceptions (reinforced by dry Statistics Canada press releases) I imagined it was nothing more than endless tabulating and mindless number crunching...

That view came to an end when I had the good fortune of taking my first *real* statistics course from Professor William J. Welch at the University of Waterloo. How different that was from my previous studies in mathematical statistics! Finally, I could see how we could learn about the *real* world by carefully designed statistical experiments.

Unfortunately, I did not have the courage to go for stats at that point in my life. That came several years later when he was about to teach a course on the Design and Analysis of Experiments at UBC. Then I decided to take the plunge and that was one of the best decisions in my life. He is an exceptionally inspirational teacher and I am grateful for his guidance as my academic advisor. I feel incredibly lucky for the opportunity to get my training in the Design and Analysis of Computer Experiments from one of the founders of this field. I have benefited immensely not only from his expertise in technical matters but also in countless other practical things that are important in academic life. Considering his many responsibilities, how he found the time to do so much for me is still a mystery and I have to admit that I often felt guilty for diverting his time from other important things.

My thanks and gratitude also go to Dr. Jason L. Loeppky, who first started advising me as a postdoctoral fellow and later became my co-supervisor.

He was instrumental in getting me started in my research the summer before I started my Ph.D. program. I will always remember our numerous stimulating conversations and debates that helped me identify my shortcomings and broaden my horizons.

I am especially grateful to both of my advisors for granting me many "degrees of freedom" of independence and for enduring my stubbornness (which led me down blind alleys several times resulting in countless hours of ultimately unproductive programming, e.g. concatenating Gaussian processes, constructing sequential designs, exploring bootstrapping schemes, etc.) Their patience and understanding is a trait I value highly.

Derek Bingham at SFU also supported me in a variety of ways (including financially) and I am still astounded by all his efforts to get me involved in conferences, workshops, summer schools, and internships.

Although it is impossible to mention everyone else by name who contributed to my memorable grad student experience here at UBC, I still want to point out several remarkable individuals. Among the faculty in the Department of Statistics, professors Paul Gustafson, Jiahua Chen, and James V. Zidek stand out for their steadfast support and useful suggestions.

Hardworking grad students in our department were too many to mention, but Jesse Raffa, Yiping Dou, and Kenneth Chi Ho Lo have inspired me on many occasions with their seemingly superhuman efforts. I also want to thank Wei Wang and Hui Shen for providing me with LaTeX templates to create this document. Of course, appreciation goes both ways and I will always remember fondly that my fellow grad students have awarded me twice for my own efforts with the "Energizer Award (keeps working and working and...)" in 2006-2007 and the "If I were allowed, I would live in the Department Award" in 2005-2006. (Unfortunately, I did not attend the award ceremonies because I was, well, working...)

This research was funded by the Natural Sciences and Engineering Research Council of Canada and the National Program on Complex Data

Structures of Canada. Computations were made using WestGrid, which is funded in part by the Canada Foundation for Innovation, Alberta Innovation and Science, BC Advanced Education, and the participating research institutions. WestGrid equipment is provided by IBM, Hewlett Packard, and SGI.

My research would not have been possible without all the technical support staff that was helping me perform enormous amounts of computation, store inordinate data sets, and use obscure software libraries. Martin Siegert at SFU has been incredibly helpful and responsive in installing, compiling, and debugging C++ code on WestGrid (robson). I will also be forever grateful to The Ha in our department and Roman Baranowski at the UBC WestGrid facility for not kicking me out for good after my careless code caused an accidental burnout of the department's workhorse (gosset), and on another occasion, a catastrophic slowdown of the largest WestGrid cluster (glacier), respectively.

The Van de Plas family (Paul, Susan, and Monique) also deserve special credit, because they had a large share in my success by welcoming me to their city and being incredibly generous with their unconditional support and time to help me get started in Vancouver. When I think of Canadian hospitality and willingness to help, their example will always glow bright.

Finally, I also want to say thank you to Dr. Matthias Schonlau for his very useful advice about how to keep focused in grad school and credit his web site for helping me finish in the minimum time allowed for my program (three years), doing my part for improving graduation statistics! However, I should also mention that I started doing research four months before starting my Ph.D. and started taking courses in probability and statistics a year before, pushing the total time invested to four years. (At the time of this writing, typing "how to finish a phd" or "finishing my phd" into Google and hitting the *I'm Feeling Lucky* button leads to his web page at `http://www.schonlau.net/finishphd.html`).

Statement of Co-Authorship

The thesis is finished under the supervision of my advisors, Professor William J. Welch and Assistant Professor Jason L. Loeppky.

Chapter 2 is co-authored with Dr. William J. Welch and Dr. Jason L. Loeppky. My main contributions are designing, running, analyzing the simulations and cross-validations, applying the results of Kass and Slate (1994) to minimize nonnormality, and most of the writing.

Chapter 3 is co-authored with Dr. William J. Welch and Dr. Jason L. Loeppky. My main contributions are designing, running, analyzing the simulations, deriving formulas for optimal transformations based on a nonnormality measure of Sprott (1973), and most of the writing.

Chapter 1

Introduction

This thesis is about how to achieve valid inference by reparameterizing a particular statistical model used in the field of computer experiments. After discussing what we mean by valid inference and its related philosophical implications, we describe the model and the rationale behind its reparameterization. Then we preview how the following chapters address specific aspects of this problem.

When we are talking about estimation or prediction, valid inference includes the ability to quantify uncertainty. Frequentists do that by constructing confidence sets, while Bayesians may prefer credible sets. In this thesis we use the classical frequentist interpretation. For instance, we view a 99% confidence region as a random entity that should cover the true value approximately 99% of the time over the course of many repeated, identical trials.

We evaluate the validity of our methods by extensive simulations, averaging over many data sets to estimate the actual coverage probabilities of the confidence regions. Unless the true coverage is roughly the same as the advertised nominal coverage, an inference method cannot be considered valid.

The methods we study can be classified as likelihood-based or Bayesian. But regardless of the philosophical underpinnings, they all have to go through the same frequentist simulation test, e.g. even when we are dealing with a Bayesian credible interval, we still evaluate it by its frequentist properties in terms of matching coverage probabilities.

Hence, one could argue that our approach is a mix of frequentist, likeli-

hoodist, and Bayesian ideas. However, that characterization would not do justice to the spirit of this work, since we are above all pragmatists, driven by practical, real-world applications, not just academic curiosities. The image of the practicing engineer, or research scientist, or some other professional experimenting with a computer model is paramount in our minds. As users, most of them could not care less about philosophical debates in statistics. What matters to them most is whether a given method works or does not work in the *real* situation they are facing. That is also a kind of philosophy we can relate to and hope that practitioners will find our contributions useful and will implement our proposals.

1.1 Computer Model

First, we need to make a distinction between the computer model or simulator and the statistical model or emulator. The computer model is not a statistical model. Instead, it is usually a complex mathematical model of ordinary and partial differential equations, implemented as a computer code, used to simulate a complex real-world phenomenon. Examples include weather modeling, chemical and biochemical reactions, particle physics, cosmology, semiconductor design, aircraft design, automotive crash simulations, etc.

Rapidly growing computing power has enabled scientists and engineers to build sophisticated computer models that can simulate a complex process to sufficient granularity, so that in some cases, it is sufficient to study the virtual world created by the simulator instead of the original physical process in the real world. This may have several advantages, since physical experimentation can be time-consuming, expensive, or not possible at all because of a variety of reasons (physical, legal, ethical, etc.) In contrast, computer experimentation is usually only limited by the available computing resources.

As cutting edge science and engineering is always pushing the boundary

of what is possible, many of these simulators tend to be computationally expensive. Furthermore, the number of input variables may be so large that a systematic exploration of all possible input combinations of interest may not be possible because of a combinatorial explosion. This necessitates a faster approximation: an emulator that emulates the simulator.

1.2 Statistical Model

The emulator is a statistical model that can predict the output of the simulator based on a relatively small number of simulation runs. Since prediction may be many orders of magnitudes faster than running the simulator code itself, the emulator may eventually replace the simulator. (This is why an emulator is sometimes called a meta-model that models the computer model).

Of course, all this hinges on the ability of the emulator to accurately predict the unknown response of the simulator at an untried input combination. This is the subject of a specialized field in statistics that started with the seminal paper (Sacks, Welch, Mitchell, and Wynn, 1989) with the title "Design and Analysis of Computer Experiments".

The design part deals with the question of how to choose the initial input combinations for the simulator runs. Classical design of experiments techniques, such as replication, randomization, or blocking do not apply, since what we are trying to predict is deterministic computer output with no observational error (if we run the code again with the same input, we get the same output). It quickly became apparent that space-filling designs were the most useful, such as the Latin hypercubes of McKay, Beckman, and Conover (1979) that are used in this thesis. Since our work is about the analysis of computer experiments, we are not going to discuss design issues any further. The interested reader is referred to the substantial literature developed over the years about the many possible ways to construct such designs (e.g. Tang (1993); Morris and Mitchell (1995), or Mease and Bingham (2006) for a

recent generalization to Latin hyperrectangles).

Although the output of a simulator may be multivariate, we can assume without loss of generality that it is univariate, since different outputs can be emulated separately. (This approach may be feasible even when the output is functional data, since sometimes we are interested in emulating only a finite number of summaries of the output function, instead of the entire function). Hence, we are emulating a deterministic computer code with a single output y as a function of $d \geq 1$ inputs: $x_1, x_2, \ldots, x_d$. Likewise, for the emulator, we use a statistical model with a single dependent variable Y and d independent variables. Here we are assuming that all variables are real numbers and that the response is a smooth function of the d-dimensional vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_d)^T$, having derivatives of all orders. This is a reasonable assumption for a large class of simulators because of the nature of the underlying system of differential equations.

The main idea is to model the deterministic function specified by the computer code with a random function $Z(\boldsymbol{x})$. This is a counter-intuitive idea, since we are infusing randomness where none exists. Nevertheless, this approach has been proven more successful over the past 20 years or so than any other method for modeling deterministic response surfaces in computer experiments.

Other common names for $Z(\boldsymbol{x})$ are Gaussian process, Gaussian stochastic process, stochastic process, spatial process, or random field, and this construct has been used extensively in spatial statistics starting with geostatistics where it is known as kriging (Cressie, 1993). However, spatial applications are usually in just two or three dimensions, while computer models can have many more input variables. (For example, in Chapter 2, we will work with a model with $d = 41$ inputs). The other major difference is that kriging models usually include a white noise term, but in the deterministic case there is no noise. This is why for example linear models are inappropriate, since the usual independence assumption for a random error

ϵ is not satisfied in a model like

$$Y(\boldsymbol{x}) \;=\; \sum_j \beta_j f_j(\boldsymbol{x}) \;+\; \epsilon,$$

where each $f_j(\boldsymbol{x})$ is a function of $\boldsymbol{x}$ with unknown β_j coefficients. But if we replace the random error term ϵ with a systematic error term $Z(\boldsymbol{x})$, then we obtain the model in Sacks et al. (1989) that is a sum of a regression component or drift and a stochastic process $Z(\boldsymbol{x})$ that captures the systematic departure from the drift:

$$Y(\boldsymbol{x}) \;=\; \sum_j \beta_j f_j(\boldsymbol{x}) \;+\; Z(\boldsymbol{x}),$$

where the assumption for the process $Z(\boldsymbol{x})$ is that it has mean zero, constant variance σ^2 and a parametric correlation function depending on some measure of distance in the input space. To simplify calculations, we assume that there is no significant drift, making the regression part unnecessary and leaving the stochastic part as the only component in our model:

$$Y(\boldsymbol{x}) \;=\; Z(\boldsymbol{x}).$$

We use the Gaussian correlation function that is a common choice for modeling smooth response surfaces:

$$\mathrm{Corr}(Z(\boldsymbol{w}),\, Z(\boldsymbol{x})) \;=\; \prod_{i=1}^{d} \exp\left\{-\theta_i (w_i - x_i)^2\right\}, \tag{1.1}$$

specifying that the correlation between the responses at input sites $\boldsymbol{w}$ and $\boldsymbol{x}$ is a function of the distance between the two points, scaled by positive θ_i range parameters, measuring how active the process is in each of the d dimensions.

We should also mention that such models are often presented in a Bayesian way, saying that what we are doing is essentially using a Gaussian process

prior for the data (Currin, Mitchell, Morris, and Ylvisaker, 1991). From that perspective, this correlation function puts all prior density on smooth, infinitely differentiable functions. However, in this thesis we avoid that kind of terminology because we use the word "Bayesian" in a different way, referring to the joint distribution of the range parameters, as part of the Fast Bayesian Inference method that is the subject of Chapter 2.

1.3 Reparameterization

Statistical models can be reparameterized for many different purposes. Our objective is to make the shape of the likelihood more Gaussian, enabling good normal approximations (i.e. approximating a likelihood function with the density function of a multivariate normal distribution). An excellent reference on this subject is Slate (1991), comparing several different measures for nonnormality. Note that this is about transforming the parameters of the model, as opposed to transforming the response, as popularized by Box and Cox (1964).

We use measures by Sprott (1973) for $d = 1$, and a multivariate extension by Kass and Slate (1994) for $d > 1$ to quantify nonnormality for relatively small sample sizes, when we cannot rely on asymptotics to guarantee a likelihood that is approximately normal. It appears that, in general, small sample normality has not been investigated as thoroughly as asymptotic normality in statistics.

But in practice, small sample results are often more relevant than large sample results. This is especially true in the field of computer experiments, where sample sizes are routinely small relative to the dimensionality of the input space because of the excessive computational cost of obtaining data. Hence, the lack of small sample focus is even more puzzling in this research area (although it is understandable from a historical perspective, since powerful computers required for simulations with finite samples have emerged only gradually over the past decades, while hardware requirements

for asymptotic investigations were rarely more than pen and paper).

Theory is lagging behind current practice, since from the practitioners' point of view the crucial question is how to make the most of a limited number of data points. But theoretical arguments are usually based on asymptotics, providing little guidance for small samples. (See Zhang and Zimmerman (2005) for a recent review of results based on increasing-domain or infill asymptotics, titled "Towards Reconciling Two Asymptotic Frameworks in Spatial Statistics").

The original inspiration for this work was Karuri (2005), who observed that in a Bayesian setting the log transformation of the range parameters improved approximate normality of the posterior for one- and two-dimensional examples and demonstrated its usefulness for integration and prediction. Following up on her original idea, we show that the log transformation can be even more useful in higher dimensions, sometimes enabling surprisingly accurate uncertainty assessments for parameter estimation and prediction.

1.4 Preview of chapters

This is a manuscript-based thesis. Chapters 2 and 3 are separate articles intended for publication. An earlier version of Chapter 2 has already been submitted to a journal and Chapter 3 will follow soon.

Chapters 2 introduces Fast Bayesian Inference (FBI) and compares it to the traditional plug-in method on both simulated and real data sets, demonstrating that the prediction bands of the FBI are more valid than those of the plug-in in terms of their frequentist coverage probabilities. The equivalence of "profiling out" and "integrating out" the process variance σ^2 is established and the resulting profile likelihood function of the range parameters $\theta_1, \ldots, \theta_d$ in (1.1) is approximated with a multivariate normal distribution. The quality of the approximation is evaluated by a nonnormality measure of Kass and Slate (1994). This measure is minimized numerically in an attempt to improve normal approximations. It is found that for large d, within

the family of power transformations, the log transformation is close to being optimal both with respect to minimizing nonnormality and assessing prediction uncertainty.

Using the same model with the same reparameterization, Chapter 3 examines how well the parameters can be estimated by maximum likelihood and how well one can quantify parameter uncertainty by normality-based confidence sets and more exact likelihood-based confidence regions. It is found that although the point estimates slightly underestimate the real parameters, uncertainty can be measured adequately by normality-based (Wald-type) confidence intervals obtained from the standard errors derived from the observed information matrix. However, Wald-type confidence ellipsoids for the joint estimation of the model parameters are not as accurate as the ones obtained from inverting the likelihood-ratio tests (which themselves can become inadequate for small sample sizes). Implications for the FBI are discussed and a Bayes estimator for σ^2 is presented that is less biased than the MLE of σ^2. Likelihood nonnormality (i.e. closeness to normal approximations) is explored graphically, revealing a mismatch in the tails. Another measure by Sprott (1973) for $d = 1$ demonstrates why the log transformation can be far from being optimal in the one-dimensional special case, explaining why results seen for $d = 1$ are in general inferior to results for $d > 1$ in Chapters 2 and 3 and Appendix B.

Chapter 4 concludes the thesis by relating the two manuscripts to each other and to the field of study, reviewing the strengths and weaknesses of the research, and discussing potential directions for future work. Appendix A to Chapter 3 contains the derivations of the formulas necessary to compute a nonnormality measure of Sprott (1973) for random function models in the $d = 1$ case. Appendix B to Chapter 4 illustrates the robustness of the FBI by additional simulation studies from Nagy, Loeppky, and Welch (2007), including a wider range of parameter choices and smaller sample sizes for $d = 1, \ldots, 10$.

Bibliography

Box, G. E. P. and Cox, D. R. (1964), "An Analysis of Transformations," *Journal of the Royal Statistical Society. Series B (Methodological)*, 26, 211–252.

Cressie, N. A. C. (1993), *Statistics for Spatial Data*, John Wiley & Sons.

Currin, C., Mitchell, T., Morris, M., and Ylvisaker, D. (1991), "Bayesian Prediction of Deterministic Functions, with Applications to the Design and Analysis of Computer Experiments," *Journal of the American Statistical Association*, 86, 953–963.

Karuri, S. W. (2005), "Integration in Computer Experiments and Bayesian Analysis," Ph.D. thesis, University of Waterloo.

Kass, R. E. and Slate, E. H. (1994), "Some Diagnostics of Maximum Likelihood and Posterior Nonnormality," *The Annals of Statistics*, 22, 668–695.

McKay, M. D., Beckman, R. J., and Conover, W. J. (1979), "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, 21, 239–245.

Mease, D. and Bingham, D. (2006), "Latin Hyperrectangle Sampling for Computer Experiments," *Technometrics*, 48, 467–477.

Morris, M. D. and Mitchell, T. J. (1995), "Exploratory Designs for Computational Experiments," *Journal of Statistical Planning and Inference*, 43, 381–402.

Nagy, B., Loeppky, J. L., and Welch, W. J. (2007), "Fast Bayesian Inference for Gaussian Process Models," Tech. Rep. 230, Department of Statistics, The University of British Columbia.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989), "Design and Analysis of Computer Experiments (C/R: P423-435)," *Statistical Science*, 4, 409–423.

Slate, E. H. (1991), "Reparameterizations of Statistical Models," Ph.D. thesis, Carnegie Mellon University.

Sprott, D. A. (1973), "Normal Likelihoods and Their Relation to Large Sample Theory of Estimation," *Biometrika*, 60, 457–465.

Tang, B. (1993), "Orthogonal Array-based Latin Hypercubes," *Journal of the American Statistical Association*, 88, 1392–1397.

Zhang, H. and Zimmerman, D. L. (2005), "Towards Reconciling Two Asymptotic Frameworks in Spatial Statistics," *Biometrika*, 92, 921–936.

Chapter 2

Quantifying Prediction Uncertainty in Computer Experiments with Fast Bayesian Inference

2.1 Introduction

Computer models have been used with great success throughout the sciences and engineering disciplines, for example in climate modeling, aviation, semiconductor design, nuclear safety, etc. Implemented as computer programs, deterministic models calculate an output y for a given input vector $\boldsymbol{x}$. Depending on the complexity of the underlying mathematical model, this can be expensive computationally, creating a need for faster approximations. A common approach is to build a statistical model to approximate the output of the computer code. This has become known as the field of computer experiments in statistics, using Gaussian process (GP) models as computationally cheap surrogates (Sacks, Welch, Mitchell, and Wynn, 1989; Currin, Mitchell, Morris, and Ylvisaker, 1991; Welch, Buck, Sacks, Wynn, Mitchell, and Morris, 1992). Trading off accuracy for speed is acceptable as long as

A version of this chapter has been submitted for publication. Authors: Nagy B., Loeppky J.L., Welch W.J.

we can measure how much the surrogate's prediction of the response might deviate from the real one. However, quantifying that uncertainty has been an ongoing challenge.

This paper is about the prediction uncertainty that originates in the GP model itself *and* from estimating the parameters of the model. Of course, there are other sources of uncertainty that can be just as important. For instance, such a simple statistical model may be an oversimplification of the complex original model. But that is outside of the scope of this investigation. Our focus is on prediction within the class of functions defined by the GP model. The rationale is that if we cannot assess prediction uncertainty decently within this class of functions (that satisfy all of our assumptions), then we cannot realistically hope to do so when working with a different class of functions (that may not satisfy the modeling assumptions).

Prediction uncertainty can be quantified by a prediction band providing confidence limits for the response surface. The typical approach in computer experiments is to pretend that the response is a random realization of a GP that can be modeled with a modest number of parameters. Based on that assumption, it is straightforward to compute normality-based prediction limits using the standard error from the prediction variance formula. As long as the model parameters are known, this is a valid practice, resulting in confidence sets that by definition have a perfect match between nominal and actual coverage probabilities at all confidence levels.

However, in practice, most of the time the parameters of the GP model are not known but need to be estimated, usually by maximum likelihood. The resulting estimates are then often used as if they were the true values. But this ignores the uncertainty in estimating the parameters. Plugging in the estimates in place of the true values in the prediction variance formula leads to prediction bands that are narrower than they should be. This has been a long standing problem of the plug-in method (see Abt (1999) for a review).

In this article, we present a Bayesian way to deal with this problem and compare the frequentist properties of the traditional plug-in method with the new method, called Fast Bayesian Inference (FBI). This research was inspired by Karuri (2005), indicating the potential usefulness of the log transformation for the parameters. We concern ourselves with a noise-free GP model using the Gaussian covariance function. Model uncertainty is purposefully ignored by assuming that the response is a realization of such a Gaussian process. The only uncertainty left about the model is the exact values of its parameters. The main finding is that FBI can successfully propagate that parameter uncertainty into assessing prediction uncertainty, leading to improved frequentist properties of the resulting prediction bands.

After defining the GP model in the next section, Section 2.3 outlines the foundations for a computationally fast Bayesian analysis. Then two simulation studies are presented in Section 2.4, followed by two real examples in Section 2.5. The proposed method is described in detail in Sections 2.6 and 2.7. We finish the article with some concluding remarks in Section 2.8.

2.2 The Gaussian Process Model

Sacks et al. (1989) gave the following general model for a deterministic computer code $y(\boldsymbol{x})$:

$$Y(\boldsymbol{x}) \;=\; \sum_j \beta_j f_j(\boldsymbol{x}) \;+\; Z(\boldsymbol{x}),$$

that is the sum of a regression model and a GP model $Z(\boldsymbol{x})$ with mean zero. Note that no white noise term is necessary because of the deterministic nature of the code, i.e. if we rerun the simulator with the same input, we always get the same output. Often the regression component can be omitted, too (e.g. see Chen (1996) and Steinberg and Bursztyn (2004)), because of the flexibility of the stochastic process that can easily take on the features of the underlying function. Following Linkletter, Bingham, Hengartner,

Higdon, and Ye (2006) we assume a standardized response with mean zero (by subtracting the mean of all observations). Thus we model the computer code $y(\boldsymbol{x})$ as if it was a realization of a mean zero Gaussian stochastic process $Z(\boldsymbol{x})$ on the d-dimensional vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_d)^T$:

$$Y(\boldsymbol{x}) \;=\; Z(\boldsymbol{x}).$$

Hence all model parameters are in the covariance function:

$$\mathrm{Cov}(Z(\boldsymbol{w}),\, Z(\boldsymbol{x})) \;=\; \sigma^2\, R(\boldsymbol{w},\, \boldsymbol{x}),$$

where σ^2 is the process variance and $R(\boldsymbol{w}, \boldsymbol{x})$ is the correlation between two configurations of the input vector, $\boldsymbol{w}$ and $\boldsymbol{x}$:

$$R(\boldsymbol{w},\, \boldsymbol{x}) \;=\; \prod_{i=1}^{d} \exp\left\{-\theta_i (w_i - x_i)^2\right\}, \tag{2.1}$$

where the positive θ_i range parameters control how variable the process is in a particular dimension. This is the Squared Exponential or Gaussian correlation function that is frequently used in computer experiments to model smooth response surfaces.

The likelihood is a function of σ^2 and the d-dimensional vector of range parameters $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_d)^T$:

$$L(\sigma^2,\, \boldsymbol{\theta}) \;\propto\; \frac{1}{(\sigma^2)^{\frac{n}{2}}\, |\boldsymbol{R}|^{\frac{1}{2}}} \exp\left\{-\frac{\boldsymbol{y}^T \boldsymbol{R}^{-1} \boldsymbol{y}}{2\sigma^2}\right\}, \tag{2.2}$$

where $\boldsymbol{y}$ is the data vector of length n and $\boldsymbol{R}$ is the $n \times n$ design correlation matrix given by (2.1) for all pairs of input vector configurations in the data set. $\boldsymbol{R}$ is a function of the range parameter vector $\boldsymbol{\theta}$. If $\boldsymbol{\theta}$ is known, then the Best Linear Unbiased Predictor (BLUP) of the response at a new $\boldsymbol{x}_0$ is

$$\hat{y}_0(\boldsymbol{\theta}) \;=\; \boldsymbol{r}(\boldsymbol{x}_0)^T \boldsymbol{R}^{-1} \boldsymbol{y}, \tag{2.3}$$

where $\boldsymbol{r}(\boldsymbol{x}_0)$ is an $n \times 1$ vector of correlations between the new $\boldsymbol{x}_0$ and the n design points (a function of $\boldsymbol{\theta}$), again given by (2.1).

Furthermore, if σ^2 is also known, then the Mean Squared Error of the BLUP is

$$\mathrm{MSE}_{\hat{y}_0}(\sigma^2,\, \boldsymbol{\theta}) \;=\; \sigma^2 \,\left(1 \;-\; \boldsymbol{r}(\boldsymbol{x}_0)^T \boldsymbol{R}^{-1} \boldsymbol{r}(\boldsymbol{x}_0)\right), \tag{2.4}$$

and these two formulas enable one to construct valid normality-based pointwise prediction bands, having a perfect match between nominal and true coverage at all levels under this model. However, that validity is dependent on the assumption that all parameters are known.

But in practice, often none of the parameters are known. Instead, they have to be estimated, usually by maximizing (2.2) to get the estimates $\hat{\sigma}^2$ and $\hat{\boldsymbol{\theta}}$. When we plug in $\hat{\sigma}^2$ in place of σ^2 and $\hat{\boldsymbol{\theta}}$ in place of $\boldsymbol{\theta}$ in (2.3) and (2.4), we lose validity in the sense that the estimator of (2.4) based on $\hat{\sigma}^2$ and $\hat{\boldsymbol{\theta}}$ is biased to be too small relative to the true mean squared error given by (2.4) based on σ^2 and $\boldsymbol{\theta}$. In the computer experiments and the geostatistics literature, this problem is seen as a serious shortcoming of the traditional plug-in method (see the review in Abt (1999) for more details).

The root of this problem is ignoring the uncertainty due to estimating the model parameters. That suggests that a Bayesian approach could potentially help. However, before considering how to deal with parameter uncertainty, it is important to realize that all parameters are not created equal. We can see that $\boldsymbol{\theta}$ exerts its influence on the BLUP (2.3) and its Mean Squared Error (2.4) in a highly nonlinear fashion through the correlation vector $\boldsymbol{r}(\boldsymbol{x}_0)$ and the correlation matrix $\boldsymbol{R}$. In contrast, the dependence on σ^2 is much simpler. It is a factor in the MSE formula (2.4), but the BLUP itself is independent of σ^2. This has important implications when the parameters are unknown. It is easier to deal with uncertainty in σ^2 than in $\boldsymbol{\theta}$ because the predictor is not affected by σ^2 and its MSE is simply proportional to σ^2. In fact, we found that it is best to treat σ^2 as a nuisance parameter and eliminate it from the likelihood because it has a relatively minor role in quantifying prediction

uncertainty. This can be done by either profiling or integrating, as shown later in Section 2.6. Either way, the result is the likelihood function $L(\boldsymbol{\theta})$ that is only a function of the range parameters and this is the likelihood that we use for all subsequent calculations. Having eliminated σ^2, we can call $L(\boldsymbol{\theta})$ the profile likelihood or just simply the likelihood for short, and the log of this function the log-likelihood: $l(\boldsymbol{\theta}) = \log L(\boldsymbol{\theta})$. In practice, one would optimize $l(\boldsymbol{\theta})$ numerically to get the Maximum Likelihood Estimate or MLE (e.g. see Welch et al. (1992)).

2.3 Outline of Fast Bayesian Inference

Bayesian statistics provides a natural way to incorporate parameter uncertainty into a predictive statistical model. However, a Bayesian approach immediately raises two nontrivial questions:

1. How to choose a prior?

2. How to sample from the posterior?

Usually, these are perceived as separate issues. But we choose a prior, together with a reparameterized likelihood, that together lead to a posterior with a multivariate normal shape. This makes sampling from the posterior trivial. The basic idea is to use a parameterization that makes the likelihood approximately normal (Gaussian shape) and then to use a prior that makes the posterior normal. This way the problem is reduced to two questions:

1. How to get a nearly normal likelihood?

2. How to get a normal posterior?

The answer to the first question lies in reparameterization. We look at a family of transformations and pick one that is optimal (or nearly optimal) with respect to a criterion measuring the quality of a second-order approximation to the log-likelihood, and hence a Gaussian shape for the likelihood

as a function of the transformed parameters. We use the family of power transformations from Tukey (1957) (extended by Box and Cox (1964)), indexed by a real λ that includes no transformation (for $\lambda = 1$) and the log transformation (for $\lambda = 0$) for the positive θ_i range parameters:

$$\gamma_i \;=\; \begin{cases} \theta_i^{\lambda}, & \lambda \neq 0, \\ \log \theta_i, & \lambda = 0. \end{cases}$$

The same λ value is used for all θ_i ($i = 1, \ldots, d$). To find the optimal λ with the least observed nonnormality, we use a third derivative-based nonnormality measure from Kass and Slate (1994):

$$\frac{1}{d^2} \sum_{i_1,i_2,i_3,i_4,i_5,i_6=1}^{d} - \partial_{i_1 i_2} l(\mathrm{MLE})^{-1} \times \partial_{i_4 i_5} l(\mathrm{MLE})^{-1} \times$$

$$\times \partial_{i_3 i_6} l(\mathrm{MLE})^{-1} \times \partial_{i_1 i_2 i_3} l(\mathrm{MLE}) \times \partial_{i_4 i_5 i_6} l(\mathrm{MLE}),$$

where $\partial_{ij} l(\mathrm{MLE})$ denotes second and $\partial_{ijk} l(\mathrm{MLE})$ third partial derivatives of the log-likelihood function l, evaluated at the MLE. By minimizing this measure, one can find a log-likelihood that has relatively small third derivatives compared to second derivatives at the mode. This means that third-order nonnormality is minimized, making the shape of the likelihood approximately Gaussian in the neighborhood of the MLE. Simulations show that the optimal λ values tend to cluster around zero (except in low dimensions). Thus, the log transformation for the range parameters empirically leads to a likelihood with approximately normal shape.

Having a nearly normal likelihood, we choose the multivariate normal posterior $N(\mathrm{MLE}, -H_{\mathrm{MLE}}^{-1})$, where H_{MLE} is the Hessian matrix of second derivatives of the log-likelihood at the MLE. Note that this is equivalent to choosing a prior. (But we never need to compute the prior, all we need is the posterior). We can see that this prior is fairly non-informative because it leads to a posterior centered at the MLE, matching the curvature of the

likelihood at the MLE up to the second order. This seems sensible as this choice will not interfere much with the information coming from the data, which is contained in the likelihood function $L(\boldsymbol{\theta})$.

Although we are not interested in the prior per se and never compute it, we should point out connections with earlier work. A nearly normal likelihood implies a nearly uniform prior on the log scale. By a change of variables, we can verify that uniform priors on the log scale are equivalent to inverse priors on the original scale, which are known to approximate the Jeffreys prior in this case (Berger, De Oliveira, and Sansó, 2001). Using the results of Chen (1996), Karuri (2005) verified this approximation for $d = 1$ and $d = 2$ and suggested that similar results may hold in higher dimensions, too. Another way to justify using inverse priors is that they give prior weights inversely proportional to the parameter values, preventing overly large parameter estimates. For example, in our model, excessively large range parameter estimates could potentially underestimate the spatial correlations in the input space, undermining our spatial model.

We should also mention that Nagy, Loeppky, and Welch (2007) presents the FBI from a slightly different viewpoint, namely as an approximation to the Bayesian method that uses uniform priors on the log scale for the range parameters. In this interpretation the FBI is only approximately Bayesian, since it is taking samples from the normal approximation of the posterior, where the posterior is proportional to the likelihood because of the uniform priors. But no matter which semantics we choose, the computations are always the same. More details are provided in Sections 2.6 and 2.7, but first we demonstrate through simulated and real examples that this method works remarkably well.

2.4 Simulations

The simulations were designed to mirror situations that one would expect to encounter in practice. That means balancing two main concerns. The

first one is that the design sample size n is often less than ideal because of the computational cost of obtaining data (slow simulators). The second is that n should still be large enough to enable meaningful prediction. One option is to tie it to the number of dimensions d. According to Sacks (Chapman, Welch, Bowman, Sacks, and Walsh, 1994; Loeppky, Sacks, and Welch, 2008), $n = 10\,d$ could often serve as a rough initial estimate for an adequate sample size. Hence, the first set of simulations used this rule for $d = 1, \ldots, 10$. To ensure adequate prediction accuracy (with median prediction errors within 5% of the range of the data), the range parameters were set to $\theta = 25/(d+1)^2$ in all dimensions. (Of course, the model fitting procedure did not make the assumption that all range parameters were the same and the resulting estimates were dispersed over a wide range). The second set of simulations halved the sample size ($n = 5\,d$) while increasing correlations between the design sites (using $\theta = 5/(d+1)^2$) to maintain comparable prediction accuracy to the first study.

To obtain 1,000 replicates for a given combination of the sample size n and the common range parameter θ, the following steps were repeated 1,000 times:

1. Select a random n point design by Latin hypercube sampling in $[0, 1]^d$ (McKay, Beckman, and Conover, 1979).

2. Sample 15 more points uniformly in $[0, 1]^d$ for prediction.

3. Generate a realization of the mean zero GP over the $n + 15$ points by setting the process variance to one and θ_i to θ for $i = 1, \ldots, d$.

4. Use the data for the n design points to fit the GP model.

5. Compute predictors with mean squared errors for the 15 additional points by the plug-in and FBI methods and then for each $\alpha = 0.01, 0.02, \ldots, 0.99$, construct $100(1-\alpha)\%$ pointwise prediction bands: predictor $\pm\, t_n^{\alpha/2}\, \sqrt{\text{MSE(predictor)}}$, where $t_n^{\alpha/2}$ is the upper $\alpha/2$ critical point of the t_n distribution.

6. Calculate coverage probabilities by counting how many of the 15 points were covered by the prediction bands of the plug-in and FBI for $\alpha = 0.01, 0.02, \ldots, 0.99$.

Finally, the resulting actual coverage probabilities for both methods were averaged over the 1,000 replicates and plotted against the nominal levels for $\alpha = 0.01, 0.02, \ldots, 0.99$. Note that although it is common to use normality-based prediction bands (especially for the plug-in), here we used the t-distribution with n degrees of freedom instead of the normal because it can slightly improve the match between nominal and true coverages, especially for small n. To make the comparison fair, here the t_n distribution was used for the plug-in, too, to match Bayesian prediction bands based on the predictive distribution (O'Hagan, 1994; Santner, Williams, and Notz, 2003).

This simulation sequence was devised to represent a typical real world scenario. Latin hypercubes are the design of choice for GP models for prediction at new, untried inputs anywhere in $[0, 1]^d$. Although there are many improved variants of Latin hypercubes (Tang, 1993; Morris and Mitchell, 1995; Mease and Bingham, 2006), the original random version of McKay et al. (1979) was used here because of the enormous number of realizations generated. 1,000 replicates were used to make sure that both design and random generation effects were averaged out in the final calculation of coverage probabilities. In addition, using 15 random points for prediction (for each realization) gave a total sample size of 15,000 to average out all sampling effects.

For each realization, we tried several different values for λ, including choosing it dynamically by numerically minimizing the nonnormality measure of Kass and Slate (1994) with respect to λ. For the two simulation studies, Figures 2.1 and 2.2, respectively show the distribution of the optimized λ values (having the least nonnormality) over 1,000 simulated data sets each for $d = 1, \ldots, 10$. We can see that unless d is one or two, the

optimal λ is usually close to zero. Since in computer experiments we are primarily interested in high-dimensional applications, we chose $\lambda = 0$ because there is no evidence that any other value is better for high d. That means using the log transformation for θ_i ($i = 1, \ldots, d$).

Figures 2.3 and 2.4 contrast the frequentist performance of the plug-in and FBI methods by plotting nominal coverage levels (from 1% to 99%) vs. actual coverage for $d = 1, 4, 7,$ and 10. The coverage probabilities were calculated by averaging over the 15 new points used for prediction and the 1,000 data sets (using $\lambda = 0$ for the log transformation). In addition to the solid line for the plug-in and the dashed line for FBI, a gray diagonal is also shown in the middle of each plot to help guide the eye: the closer the curves are to the diagonal, the better the match between nominal coverage (horizontally) and true coverage (vertically). Without exception, FBI achieved closer matching coverage than the plug-in at all levels for all $d = 1, \ldots, 10$ in both simulation studies. Results for $d = 2, 3, 5, 6, 8, 9$ were similar (not shown here). Except for $d = 1$, the dashed curves came remarkably close to the diagonal representing perfect matching (from 1% to 99% in Figures 2.3 and 2.4). Hence we can conclude that according to this frequentist criterion, FBI with $\lambda = 0$ provides approximately valid inference about prediction accuracy and is clearly superior to the plug-in method in this respect.

Other λ values around zero yield similar results in terms of coverage probabilities. Also, using the optimal λ for each data set (instead of a fixed value) has no additional benefit. That suggests that the log transformation is nearly optimal in higher dimensions not only with respect to the nonnormality of the likelihood function, but also in terms of matching coverage probabilities of the FBI predictions bands.

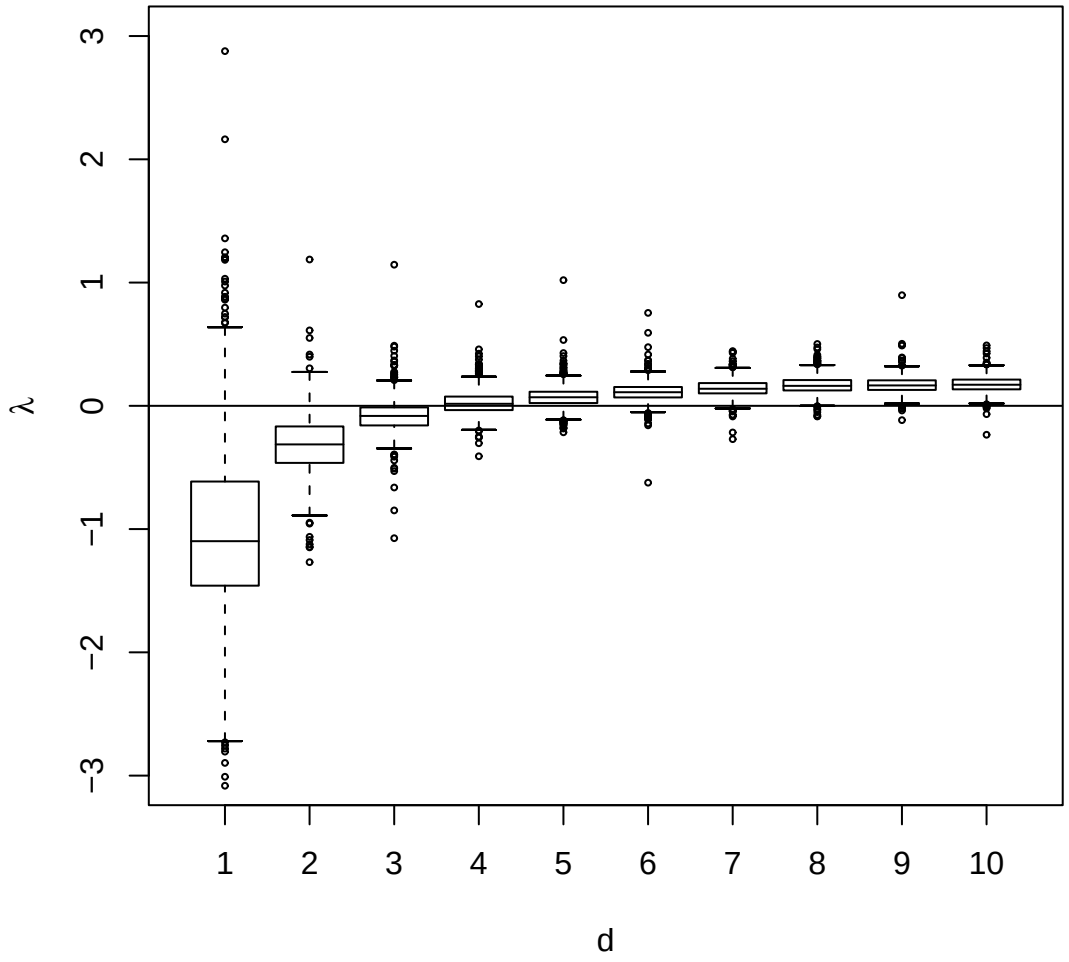


Figure 2.1: Optimal λ values in the first simulation study ($n = 10\,d$) for $d = 1, \ldots, 10$.

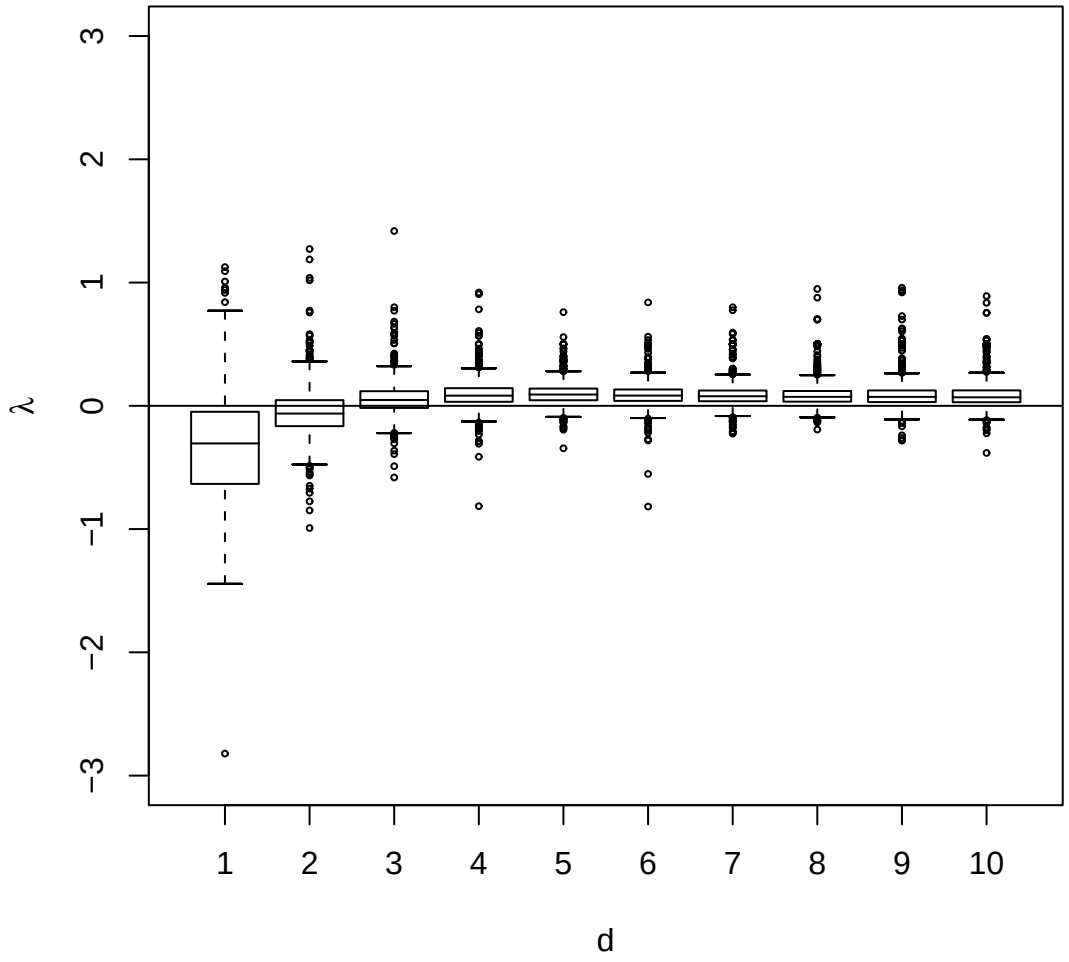


Figure 2.2: Optimal λ values in the second simulation study ($n = 5\,d$) for $d = 1, \ldots, 10$.

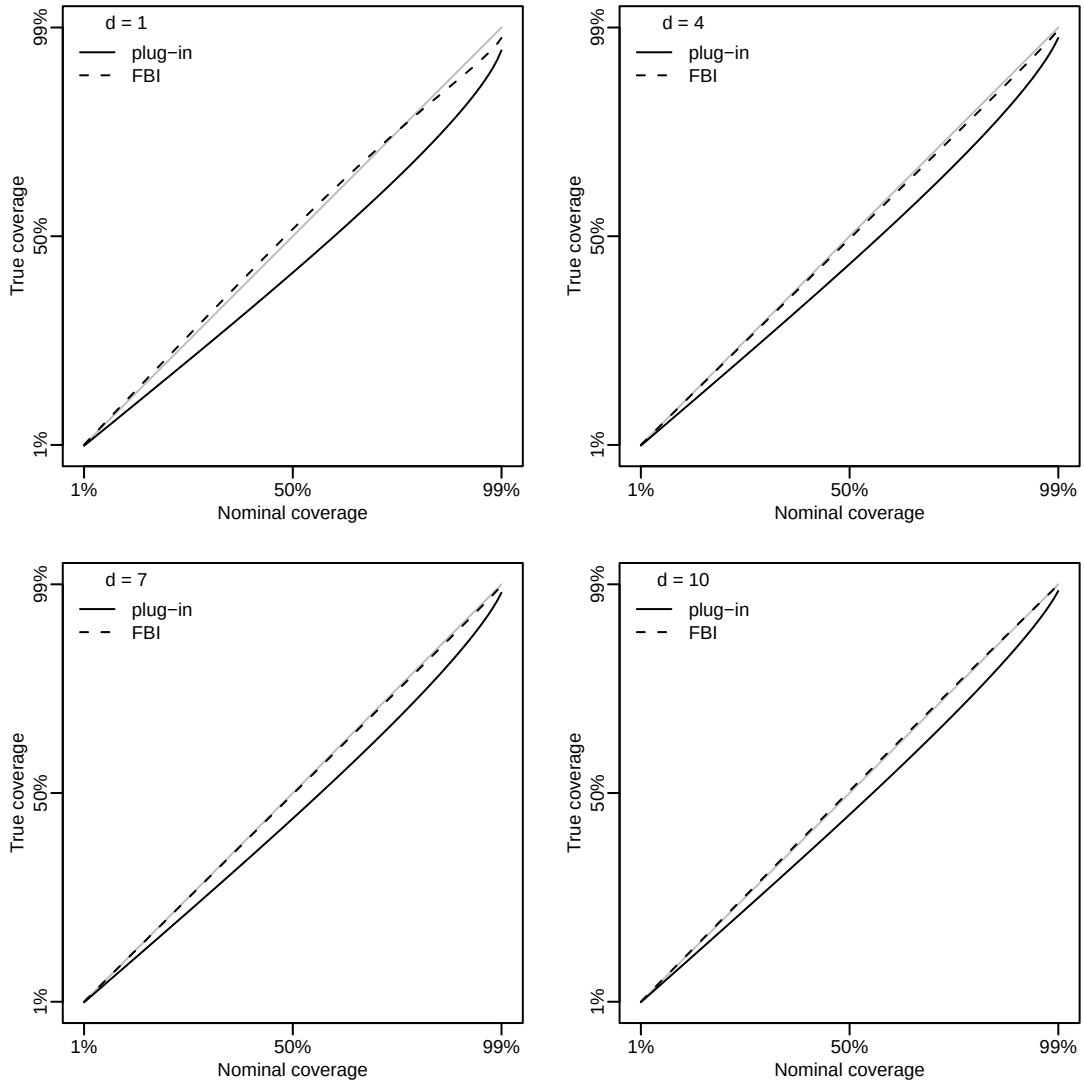


Figure 2.3: Coverage probabilities in the first simulation study ($n = 10\,d$) for $d = 1,\ 4,\ 7,$ and 10.

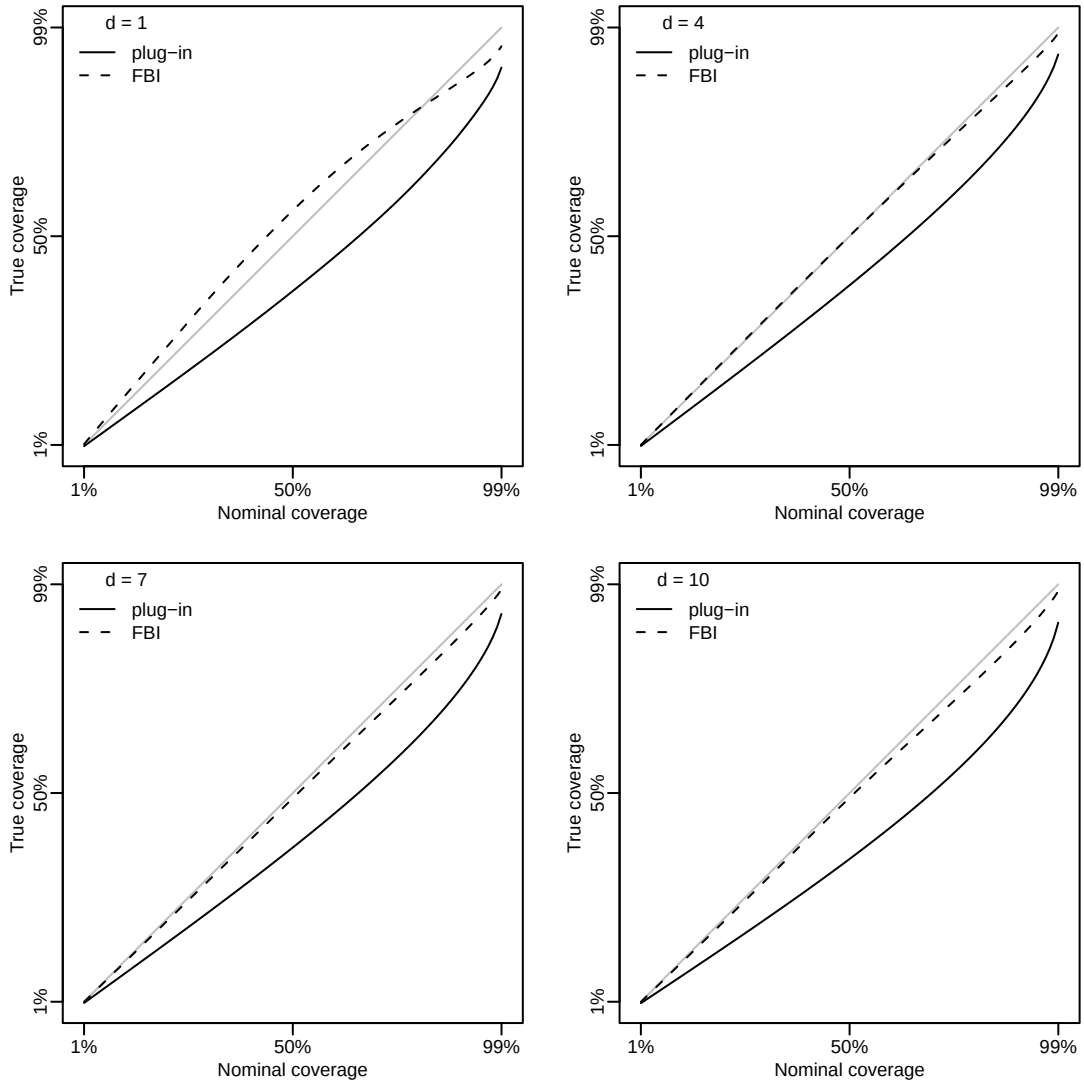


Figure 2.4: Coverage probabilities in the second simulation study ($n = 5\,d$) for $d = 1,\ 4,\ 7,$ and 10.

2.5 Examples

The prediction uncertainty assessments of the two methods were also compared on two real data sets by a computationally intensive version of cross validation. This was done by randomly splitting the data in two (for training and validation) 100 times, and then averaging the resulting coverage probabilities the same way as for the 1,000 replicates for the simulations in Section 2.4. Here 100 replicates were sufficient because they were all subsets of the same data set. Also, for the design size $n < 5d$ was sufficient because of strong correlations between the design sites.

For a fixed design size n and a data set size m, the following steps were repeated 100 times:

1. Select n points randomly (without replacement) from the available m points.

2. Use the data for those n points to fit the GP model, using the log transformation for the range parameters ($\lambda = 0$).

3. For each $\alpha = 0.01,\ 0.02,\ \ldots,\ 0.99$, construct $100(1-\alpha)$ pointwise prediction bands for the remaining $m - n$ points by both methods.

4. Calculate coverage probabilities by counting how many of those $m-n$ points are covered by the prediction bands of the plug-in and FBI for $\alpha = 0.01,\ 0.02,\ \ldots,\ 0.99$.

Finally, the resulting actual coverage probabilities were averaged over the 100 replicates and plotted against the nominal levels to facilitate visual comparison to the simulation results. When doing so, we have to keep in mind that there is an important difference between simulated and real data sets. When one generates data from the GP model repeatedly, then one can expect that over the long-run, any useful inference method should show reasonably valid performance, since the data is from the true model, satisfying all modeling assumptions. But if the data comes from the real world, where

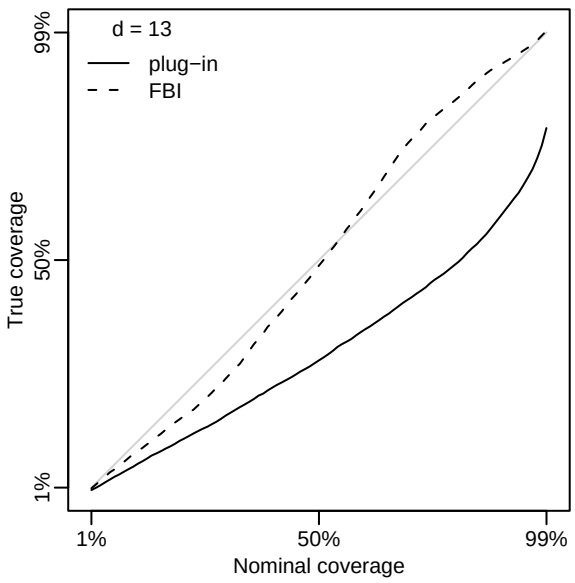


Figure 2.5: Coverage probabilities for the Arctic sea ice example ($n = 50$).

the GP model may or may not be appropriate, that can potentially lead to other inference difficulties.

The first example from Chapman et al. (1994) had $m = 157$ data points in a 13-dimensional input space representing 157 successful runs of a dynamic-thermodynamic Arctic sea ice model with 13 inputs and four outputs. One of the outputs, sea ice velocity, proved especially resistant to prediction uncertainty assessments by the plug-in method, because the standard errors of the predictions were too small and as a result, the prediction bands were always too narrow. To see whether FBI can quantify prediction uncertainty better, random subsets of $n = 50$ were chosen repeatedly (100 times) to fit the model, leaving the remaining 107 points for validation. Figure 2.5 shows the coverage probabilities averaged over all repetitions. By looking at the solid line, it is apparent that the plug-in method indeed underestimated the uncertainty by a large margin. The dashed line for FBI is closer to the diagonal, indicating a better match.

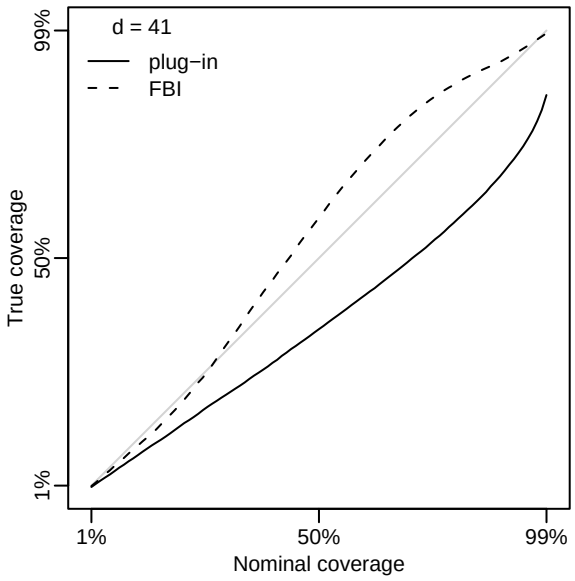


Figure 2.6: Coverage probabilities for the Wonderland example ($n = 100$).

Figure 2.6 is for a more challenging 41-dimensional example with $m = 500$ data points, out of which $n = 100$ were used for fitting, leaving 400 for validation. This is the Wonderland simulator of Milik, Prskawetz, Feichtinger, and Sanderson (1996) for global sustainability with 41 inputs. Here the response is a human development index. Again, the prediction bands of the plug-in are too narrow and the FBI is also far from perfect, often making the opposite mistake by stretching the bands too wide, as indicated by the portion of the dashed line over the diagonal. (Although one could argue that overcoverage is often preferable to undercoverage). But the true coverage of the FBI is still closer to the nominal than that of the plug-in at all confidence levels.

We can see that in both cases, the coverage of the FBI has larger deviations than in the simulations (undercovering at lower levels and overcovering at higher ones). Nevertheless, at the highest confidence levels it is close to the diagonal, indicating a good match. For example, at the 95% nominal

level, the actual coverage of the FBI is 95.7% in both cases. In contrast, the plug-in's true coverage at the 95% level is only 67.8% in Figure 2.5 and 75.7% in Figure 2.6.

2.6 Dealing with the process variance

This section formally defines the likelihood $L(\boldsymbol{\theta})$ that is a function of only the range parameters. Two possible ways are presented for eliminating the process variance σ^2: "maximizing out" to get the profile likelihood and "integrating out" to get the integrated likelihood (see Berger, Liseo, and Wolpert (1999) for a general discussion of these methods). While profiling is common in likelihood-based settings, Bayesians are usually more comfortable with integrating. Although in this case the same $L(\boldsymbol{\theta})$ function is obtained both ways, interpretations can still differ depending on the underlying framework.

2.6.1 Profile likelihood

Given $\boldsymbol{\theta}$, $L(\sigma^2,\, \boldsymbol{\theta})$ in equation (2.2) has a unique maximum at

$$\hat{\sigma}^2(\boldsymbol{\theta}) \;=\; \frac{\boldsymbol{y}^T \boldsymbol{R}^{-1} \boldsymbol{y}}{n}. \tag{2.5}$$

This is easily obtained by differentiating $L(\sigma^2,\, \boldsymbol{\theta})$ with respect to σ^2 or by observing that given $\boldsymbol{\theta}$ and $\boldsymbol{y}$, the likelihood (2.2) is proportional to an Inverse Gamma density function with respect to the variable σ^2:

$$\sigma^2 \,|\, \boldsymbol{\theta},\, \boldsymbol{y} \;\sim\; IG\left(\frac{n}{2} - 1,\, \frac{\boldsymbol{y}^T \boldsymbol{R}^{-1} \boldsymbol{y}}{2}\right)$$

and using the $b/(a+1)$ formula for the mode of an Inverse Gamma distribution $IG(a,\, b)$ with density function

$$f(\, x \,|\, a,\, b\,) \;=\; \frac{b^a \, \exp\left\{-\frac{b}{x}\right\}}{\Gamma(a)\, x^{a+1}}.$$

Plugging in $\hat{\sigma}^2(\boldsymbol{\theta})$ from (2.5) into (2.2) yields the profile likelihood:

$$L(\boldsymbol{\theta}) \propto \frac{1}{(\hat{\sigma}^2(\boldsymbol{\theta}))^{\frac{n}{2}} |\boldsymbol{R}|^{\frac{1}{2}}} \exp\left\{-\frac{\boldsymbol{y}^T\boldsymbol{R}^{-1}\boldsymbol{y}}{2\hat{\sigma}^2(\boldsymbol{\theta})}\right\} \propto (\boldsymbol{y}^T\boldsymbol{R}^{-1}\boldsymbol{y})^{-\frac{n}{2}} |\boldsymbol{R}|^{-\frac{1}{2}}.$$

Now the maximum likelihood estimation can be done using $L(\boldsymbol{\theta})$ instead of the original $L(\sigma^2, \boldsymbol{\theta})$, reducing the dimensionality of the required numerical optimization by one.

2.6.2 Integrated likelihood

Bayesians prefer to put a prior distribution on σ^2 before eliminating it. According to Berger et al. (2001), the most common choice is that of Handcock and Stein (1993), who used the improper prior $1/\sigma^2$ for $\sigma^2 > 0$. This can be interpreted as a relative weight function giving prior weights inversely proportional to the magnitude, encouraging σ^2 to be close to zero. Let $\pi(\boldsymbol{\theta})$ denote the prior for the range parameters, independent of σ^2. Then the joint prior is of the form $\pi(\boldsymbol{\theta})/\sigma^2$ and the posterior is obtained by multiplying with the likelihood (2.2):

$$\frac{\pi(\boldsymbol{\theta})}{\sigma^2} L(\sigma^2, \boldsymbol{\theta}) \propto \frac{\pi(\boldsymbol{\theta})}{(\sigma^2)^{\frac{n}{2}+1} |\boldsymbol{R}|^{\frac{1}{2}}} \exp\left\{-\frac{\boldsymbol{y}^T\boldsymbol{R}^{-1}\boldsymbol{y}}{2\sigma^2}\right\}$$

and notice that

$$\sigma^2 \mid \boldsymbol{\theta}, \boldsymbol{y} \sim IG\left(\frac{n}{2}, \frac{\boldsymbol{y}^T\boldsymbol{R}^{-1}\boldsymbol{y}}{2}\right)$$

which means that σ^2 can be integrated out from the posterior to get the marginal posterior of $\boldsymbol{\theta}$:

$$\int_0^\infty \frac{\pi(\boldsymbol{\theta})}{(\sigma^2)^{\frac{n}{2}+1} |\boldsymbol{R}|^{\frac{1}{2}}} \exp\left\{-\frac{\boldsymbol{y}^T\boldsymbol{R}^{-1}\boldsymbol{y}}{2\sigma^2}\right\} d\sigma^2 = \frac{\pi(\boldsymbol{\theta})\,\Gamma\left(\frac{n}{2}\right)}{\left(\frac{\boldsymbol{y}^T\boldsymbol{R}^{-1}\boldsymbol{y}}{2}\right)^{\frac{n}{2}} |\boldsymbol{R}|^{\frac{1}{2}}} \propto \pi(\boldsymbol{\theta}) L(\boldsymbol{\theta}).$$

Note that after integrating, the posterior for $\boldsymbol{\theta}$ is proportional to the prior for $\boldsymbol{\theta}$ times the same likelihood function $L(\boldsymbol{\theta})$ as above, which means that

in this case profiling and integrating leads to the same likelihood function for the remaining parameters.

2.7 Fast Bayesian Inference in detail

Using the notation $\boldsymbol{\gamma} = (\log \theta_1, \ldots, \log \theta_d)^T = \log \boldsymbol{\theta}$ for the transformed parameter vector and $\boldsymbol{\theta} = (\exp \gamma_1, \ldots, \exp \gamma_d)^T = \exp \boldsymbol{\gamma}$ for the inverse transformation, the transformed likelihood function $L(\exp \boldsymbol{\gamma})$ tends to have a shape that is closer to a normal distribution with respect to $\boldsymbol{\gamma}$ than the shape of the original $L(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$. Working with log-likelihoods, the equivalent statement is that $l(\exp \boldsymbol{\gamma})$ is usually more quadratic than $l(\boldsymbol{\theta})$, which incidentally can also help the Maximum Likelihood Estimation that needs to be done numerically. Another advantage of the log transformation is that it makes the numerical optimization of the log-likelihood unconstrained: $\boldsymbol{\gamma} \in \mathbb{R}^d$. This is the first step of Fast Bayesian Inference, that can be summarized as follows:

1. Maximize the log-likelihood $l(\exp \boldsymbol{\gamma})$ to get the MLE of $\boldsymbol{\gamma}$, denoted $\hat{\boldsymbol{\gamma}}$.

2. Compute the Hessian matrix of second derivatives of the log-likelihood at $\hat{\boldsymbol{\gamma}}$, denoted $H_{\hat{\boldsymbol{\gamma}}}$.

3. Sample from the multivariate normal distribution $N(\hat{\boldsymbol{\gamma}}, -H_{\hat{\boldsymbol{\gamma}}}^{-1})$ to obtain $M = 400$ Monte Carlo samples: $\boldsymbol{\gamma}^{(1)}, \ldots, \boldsymbol{\gamma}^{(M)}$.

4. The FBI predictor is obtained by averaging:

$$\frac{1}{M} \sum_{i=1}^{M} \hat{y}_0(\exp \boldsymbol{\gamma}^{(i)}),$$

   and its Mean Squared Error can be computed by the the variance

decomposition formula:

$$\frac{1}{M}\sum_{i=1}^{M}\mathrm{MSE}_{\hat{y}_0}\left(\hat{\sigma}^2(\exp\boldsymbol{\gamma}^{(i)}),\ \exp\boldsymbol{\gamma}^{(i)}\right) +$$

$$+\ \frac{1}{M-1}\sum_{j=1}^{M}\left(\hat{y}_0(\exp\boldsymbol{\gamma}^{(j)}) - \frac{1}{M}\sum_{i=1}^{M}\hat{y}_0(\exp\boldsymbol{\gamma}^{(i)})\right)^2,$$

that is the average MSE of the plug-in predictors plus the sample variance of those predictors. It is instructive to compare this sequence to the plug-in method (as described in Section 2.2). Both start by locating the MLE. After that the plug-in method jumps into the prediction phase right away, assuming that the value found at the mode is the one best estimate of the truth.

The FBI is more careful. In the second step it looks at the curvature of the log-likelihood at the MLE to quantify the uncertainty in the estimation of the *point* estimate. For example, if the surface is flat, that means high uncertainty and the corresponding normal posterior in step 3 will have a high variance reflecting that uncertainty. In the final step, the FBI averages predictions based on the sample from that normal posterior. Again, there is a part that is identical to the plug-in method, since for each sample point, equations (2.3) and (2.4) are used to calculate the predictor and its Mean Squared Error, respectively (also using (2.5) to estimate σ^2 for a given $\boldsymbol{\gamma}^{(i)}$ in the sample). This way the FBI will have many predictions to average (one for each sample point), while the plug-in method will have just one. Hence, the plug-in can be viewed as a special case of the FBI with Monte Carlo sample size $M = 1$.

2.8 Concluding remarks

We have introduced a new method for quantifying prediction uncertainty in computer experiments that is conceptually simple and easy to implement

in practice. We have also shown how much the traditional plug-in method can underestimate prediction uncertainty by ignoring parameter uncertainty. Fast Bayesian Inference can potentially correct this deficiency by incorporating the uncertainty around the MLE. This is accomplished by utilizing a non-interfering prior that leaves the mode of the likelihood where it is and also leaves the curvature at the mode unchanged by a normal posterior that matches that curvature (up to the second order). We have also found that the log transformation for the range parameters was effective in limiting (third order) nonnormality. The main advantage of a normal posterior is that it allows one to draw independent samples from it directly, facilitating fast and easy Bayesian analysis. Although we are not dealing explicitly with the uncertainty in estimating the parameter σ^2, we have seen that incorporating only the uncertainty in estimating $\boldsymbol{\theta}$ (and plugging in the MLE of σ^2 conditional on $\boldsymbol{\theta}$) can propagate sufficient parameter uncertainty through the model for potentially valid prediction uncertainty assessments.

The implementation of the FBI method is straightforward, since it is a simple add-on to the plug-in. It can also be included in a commercial or open source software package as black box computer code, since the user does not need to know anything about its inner workings. Runtimes are comparable to that of the plug-in, since computations are dominated by the numerical optimization required to find the MLE. Hence, the word fast in the name of the method is applicable to both implementation or coding time and execution or run time.

Finally, it is important to point out that when one expects the FBI to give valid prediction uncertainty assessments, one needs to keep in mind the two fundamental limitations of our study. The first one was mentioned already: the potential validity of the method rests on the assumption of a Gaussian process as the data generating mechanism. However, for real data, this assumption may be inadequate or totally wrong and results will be entirely dependent on the real underlying function.

The second serious limitation is that we studied the frequentist properties of the prediction bands in terms of coverage probabilities. Hence, validity is implied only over a long sequence of identical trials, according to the classical frequentist interpretation. But in practice, most of the time there is just one unique data set. However, the use of this criterion is not limited to frequentists. It is not uncommon for Bayesians to use it as a sanity check for their Bayesian credible regions. For example, Bayarri and Berger (2004) argue that "there is a sense in which essentially everyone should ascribe to frequentism" and provide the following version of the frequentist principle: "In repeated practical use of a statistical procedure, the long-run average actual accuracy should not be less than (and ideally should equal) the long-run average reported accuracy".

Bibliography

Abt, M. (1999), "Estimating the Prediction Mean Squared Error in Gaussian Stochastic Processes with Exponential Correlation Structure," *Scandinavian Journal of Statistics*, 26, 563–578.

Bayarri, M. J. and Berger, J. O. (2004), "The Interplay of Bayesian and Frequentist Analysis," *Statistical Science*, 19, 58–80.

Berger, J. O., De Oliveira, V., and Sansó, B. (2001), "Objective Bayesian Analysis of Spatially Correlated Data," *Journal of the American Statistical Association*, 96, 1361–1374.

Berger, J. O., Liseo, B., and Wolpert, R. L. (1999), "Integrated Likelihood Methods for Eliminating Nuisance Parameters," *Statistical Science*, 14, 1–28.

Box, G. E. P. and Cox, D. R. (1964), "An Analysis of Transformations," *Journal of the Royal Statistical Society. Series B (Methodological)*, 26, 211–252.

Chapman, W. L., Welch, W. J., Bowman, K. P., Sacks, J., and Walsh, J. E. (1994), "Artic sea ice variability: Model sensitivities and a multidecadal simulation," *Journal of Geophysical Research*, 99, 919–936.

Chen, X. (1996), "Properties of Models for Computer Experiments," Ph.D. thesis, University of Waterloo.

Currin, C., Mitchell, T., Morris, M., and Ylvisaker, D. (1991), "Bayesian Prediction of Deterministic Functions, with Applications to the Design and

Analysis of Computer Experiments," *Journal of the American Statistical Association*, 86, 953–963.

Handcock, M. S. and Stein, M. L. (1993), "A Bayesian Analysis of Kriging," *Technometrics*, 35, 403–410.

Karuri, S. W. (2005), "Integration in Computer Experiments and Bayesian Analysis," Ph.D. thesis, University of Waterloo.

Kass, R. E. and Slate, E. H. (1994), "Some Diagnostics of Maximum Likelihood and Posterior Nonnormality," *The Annals of Statistics*, 22, 668–695.

Linkletter, C., Bingham, D., Hengartner, N., Higdon, D., and Ye, K. Q. (2006), "Variable Selection for Gaussian Process Models in Computer Experiments," *Technometrics*, 48, 478–490.

Loeppky, J. L., Sacks, J., and Welch, W. J. (2008), "Choosing the Sample Size of a Computer Experiment: A Practical Guide," Tech. Rep. 238, Department of Statistics, The University of British Columbia.

McKay, M. D., Beckman, R. J., and Conover, W. J. (1979), "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, 21, 239–245.

Mease, D. and Bingham, D. (2006), "Latin Hyperrectangle Sampling for Computer Experiments," *Technometrics*, 48, 467–477.

Milik, A., Prskawetz, A., Feichtinger, G., and Sanderson, W. C. (1996), "Slow-fast dynamics in Wonderland," *Environmental Modeling and Assessment*, 1, 3–17.

Morris, M. D. and Mitchell, T. J. (1995), "Exploratory Designs for Computational Experiments," *Journal of Statistical Planning and Inference*, 43, 381–402.

Nagy, B., Loeppky, J. L., and Welch, W. J. (2007), "Fast Bayesian Inference for Gaussian Process Models," Tech. Rep. 230, Department of Statistics, The University of British Columbia.

O'Hagan, A. (1994), *Kendall's Advanced Theory of Statistics. Volume 2B: Bayesian Inference*, Cambridge University Press.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989), "Design and Analysis of Computer Experiments (C/R: P423-435)," *Statistical Science*, 4, 409–423.

Santner, T. J., Williams, B. J., and Notz, W. (2003), *The Design and Analysis of Computer Experiments*, Springer Verlag, New York.

Steinberg, D. M. and Bursztyn, D. (2004), "Data Analytic Tools for Understanding Random Field Regression Models," *Technometrics*, 46, 411–420.

Tang, B. (1993), "Orthogonal Array-based Latin Hypercubes," *Journal of the American Statistical Association*, 88, 1392–1397.

Tukey, J. W. (1957), "On the Comparative Anatomy of Transformations," *The Annals of Mathematical Statistics*, 28, 602–632.

Welch, W. J., Buck, R. J., Sacks, J., Wynn, H. P., Mitchell, T. J., and Morris, M. D. (1992), "Screening, Predicting, and Computer Experiments," *Technometrics*, 34, 15–25.

Chapter 3

Inference for covariance parameters of a random function by likelihood-based approximations

3.1 Introduction

Random function models, also known as Gaussian process models or kriging models, have a long history in spatial statistics (Cressie, 1993). Other important application areas include the design and analysis of computer experiments dating back to Sacks, Welch, Mitchell, and Wynn (1989) and more recently machine learning (Rasmussen and Williams, 2006).

Although sometimes the model parameters themselves can be of interest (Mardia and Marshall, 1984; Abt and Welch, 1998; Wang and Zhang, 2003), usually one is more interested in prediction than parameter estimation. Likewise, the main interest is quantifying prediction uncertainty instead of parameter uncertainty. However, ignoring the uncertainty in estimating the parameters leads to underestimating the uncertainty in predictions (Abt, 1999).

A version of this chapter will be submitted for publication. Authors: Nagy B., Loeppky J.L., Welch W.J.

Our interest in this problem arose because of Fast Bayesian Inference (FBI) for deterministic computer codes, as described in Chapter 2, suggesting that quantifying parameter uncertainty was the key to good prediction uncertainty assessments. Our primary goal in this chapter is to investigate how well the covariance parameters can be estimated using the FBI framework. A secondary goal is to evaluate how well the likelihood can be approximated by a normal density function, which is another important aspect of the FBI method and its ability to accurately and efficiently assess the uncertainty in predictions.

In computer experiments, a random function model is used as a computationally cheap statistical surrogate for a complex mathematical model, implemented as a computer code. Often it takes a considerable amount of time to run the code because of the large amounts of computation involved. In general, it is not possible to run them at each input combination of interest because that would lead to a combinatorial explosion for models with several input variables. In these cases the surrogate can be used to approximate the output of the code, based on the outputs from a relatively small sample from the input space.

Hence, from the practitioners' point of view, small sample results are more relevant in computer experiments than large sample results. We evaluate small sample properties by extensive simulations. Existing theory is not very helpful in this context, since it is built mostly on asymptotic arguments (see Stein (1999); Zhang and Zimmerman (2005); Furrer (2005) for the current state-of-the-art of theoretical development).

After reviewing the statistical model used by FBI in the next section together with the related issue of reparameterizations, Section 3.3 describes two sets of simulations using the same model as in Chapter 2, with the same reparameterization (log transformation). Section 3.4 presents the simulation results for the estimation of the parameters, including an assessment of the uncertainty in the estimation by individual and joint confidence sets.

Likelihood-based and Bayesian methods used to obtain those results are discussed in Section 3.5. We finish the chapter with some concluding remarks in Section 3.6.

3.2 Statistical Model

We consider a deterministic computer code with a single output that is a smooth function of $d \geq 1$ input variables. Here we reuse the model in Section 2.2, Chapter 2, that is a version of the statistical formulation in Sacks et al. (1989), Currin, Mitchell, Morris, and Ylvisaker (1991), or Welch, Buck, Sacks, Wynn, Mitchell, and Morris (1992), treating the response (code output) as if it was a realization of a real-valued, zero-mean Gaussian stochastic process $Z(\boldsymbol{x})$ on the d-dimensional real vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_d)^T$:

$$Y(\boldsymbol{x}) \;=\; Z(\boldsymbol{x}).$$

$Z(\boldsymbol{x})$ is parameterized by the process variance σ^2 and the θ_i range parameters in the Gaussian correlation function:

$$\mathrm{Corr}(Z(\boldsymbol{w}),\, Z(\boldsymbol{x})) \;=\; R(\boldsymbol{w},\, \boldsymbol{x}) \;=\; \prod_{i=1}^{d} \exp\left\{-\theta_i (w_i - x_i)^2\right\}, \tag{3.1}$$

specifying that the correlation is a function of the squared distance between the coordinates of the input vectors $\boldsymbol{w}$ and $\boldsymbol{x}$, scaled by the θ_i parameters along the d dimensions ($i = 1, \ldots, d$).

3.2.1 Likelihood

The likelihood is a function of $d + 1$ variables: the range parameters in the d-dimensional vector $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_d)^T$ and the process variance σ^2:

$$L(\sigma^2,\, \boldsymbol{\theta}) \;\propto\; \frac{1}{(\sigma^2)^{\frac{n}{2}}\, |\boldsymbol{R}|^{\frac{1}{2}}} \exp\left\{-\frac{\boldsymbol{y}^T \boldsymbol{R}^{-1} \boldsymbol{y}}{2\sigma^2}\right\}, \tag{3.2}$$

where the $n \times 1$ vector $\boldsymbol{y}$ contains the n outputs of the computer code for the n design points in the input space, and $\boldsymbol{R}$ is the $n \times n$ design correlation matrix (a function of $\boldsymbol{\theta}$), as specified by (3.1).

3.2.2 Profile likelihood

By differentiating (3.2) with respect to σ^2, we get that given $\boldsymbol{\theta}$, the likelihood $L(\sigma^2,\, \boldsymbol{\theta})$ reaches its maximum at

$$\hat{\sigma}^2(\boldsymbol{\theta}) \;=\; \frac{\boldsymbol{y}^T \boldsymbol{R}^{-1} \boldsymbol{y}}{n}. \tag{3.3}$$

Now if we plug in $\hat{\sigma}^2(\boldsymbol{\theta})$ in (3.3) in place of σ^2 into (3.2), we get the profile likelihood $L(\boldsymbol{\theta})$ that is only a function of the d range parameters:

$$L(\boldsymbol{\theta}) \;\propto\; \frac{1}{(\hat{\sigma}^2(\boldsymbol{\theta}))^{\frac{n}{2}}\, |\boldsymbol{R}|^{\frac{1}{2}}} \exp\left\{ -\frac{\boldsymbol{y}^T \boldsymbol{R}^{-1} \boldsymbol{y}}{2\hat{\sigma}^2(\boldsymbol{\theta})} \right\} \;\propto\; (\boldsymbol{y}^T \boldsymbol{R}^{-1} \boldsymbol{y})^{-\frac{n}{2}}\, |\boldsymbol{R}|^{-\frac{1}{2}}. \tag{3.4}$$

Note that one can get the Maximum Likelihood Estimate (MLE) of $\boldsymbol{\theta}$, denoted $\hat{\boldsymbol{\theta}}$, by maximizing $L(\boldsymbol{\theta})$, and then get the MLE of σ^2 by plugging in $\hat{\boldsymbol{\theta}}$ into (3.3).

3.2.3 Log transformation

For parameters that can take only positive values, the log transformation is commonly employed in statistics for various reasons. One such objective is to improve normality of the likelihood for small sample sizes, as argued by Sprott (1973). A thorough investigation of this subject was provided by Slate (1991), showing how reparameterizations of statistical models can make the shape of the likelihood or posterior more Gaussian, enabling good normal approximations.

Karuri (2005) observed that in a Bayesian setting, the log transformation of the range parameters in a random function model improved approximate normality of the posterior for one- and two-dimensional examples and

demonstrated its usefulness for integration and prediction.

Nagy, Loeppky, and Welch (2007a) found that in the one-dimensional ($d = 1$) case this was a general trend for this model, too: the log transformation tends to reduce nonnormality of the profile likelihood, as quantified by two nonnormality measures in Sprott (1973). (In subsection 3.2.5, we revisit one of those measures to illustrate which transformations one could expect to be optimal for reducing nonnormality in the $d = 1$ case).

In Chapter 2 and earlier in Nagy, Loeppky, and Welch (2007b) we demonstrated the usefulness of working on the log scale for $d = 1, \ldots, 10$ for the prediction uncertainty problem across a wide range of parameter settings. Using a multivariate nonnormality measure of Kass and Slate (1994), in Section 2.4, Chapter 2, we also showed that the log transformation was nearly optimal for large d in the class of power transformations.

3.2.4 Example

To give some intuition about the relationship between the likelihood, the profile likelihood, and the log transformation, we present a one-dimensional ($d = 1$) toy example. Although the log transformation is rarely ideal for $d = 1$ (as we will show in the next subsection), it can still illustrate the general principles using the simplest possible case (and leave it up to the readers' imagination to extrapolate from that to higher-dimensional cases).

This example was created the following way: after simulating $n = 3$ data points from a one-dimensional random function repeatedly, using $\theta = 0.2$, $\sigma^2 = 1$, and an equispaced design $\{0, 0.5, 1\}$, we chose a realization where the log transformation was particularly successful in improving the approximate normality of the profile likelihood (for other realizations the approximation was also substantially helped by the log transformation, but in general not as much as for the one chosen here for illustration; see Nagy et al. (2007a) for simulations and quantitative arguments based on two nonnormality measures of Sprott (1973) about the effect of the log transformation for $d = 1$

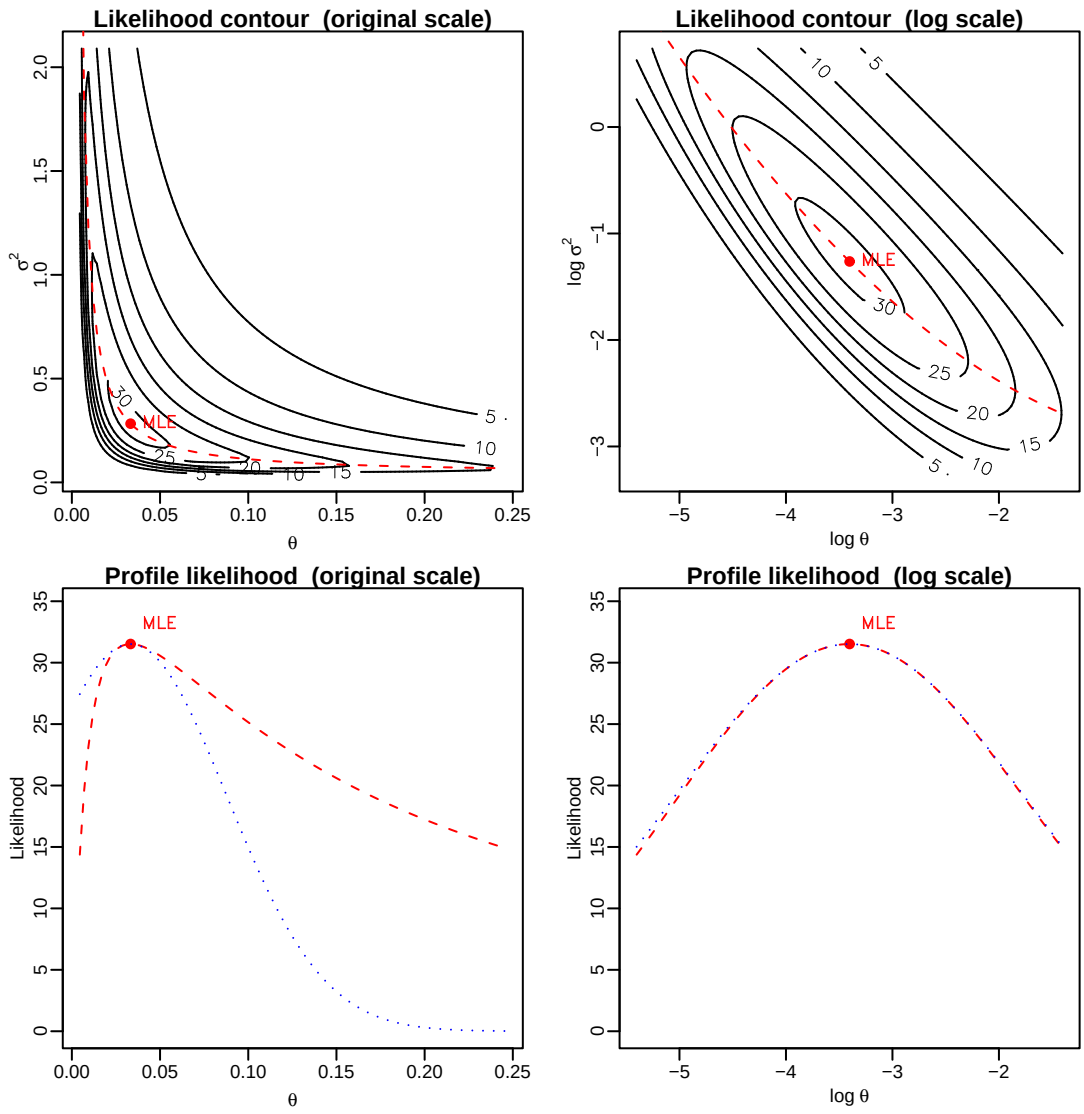


Figure 3.1: The log transformation improved approximate normality of the profile likelihood for this one-dimensional ($d = 1$) example. The top two plots are for the two-parameter likelihood and the bottom two for the one-parameter profile likelihood. The ridges of the contours are marked by the dashed lines, reaching their apex at the MLE. Below the contour plots, these dashed lines are plotted as functions of the range parameter, representing the profile likelihood function. In addition to the profile likelihoods (dashed curves), their normal approximation is also shown for comparison (dotted curves).

and $n = 3, 6, 9, 12$).

Likelihood functions for this example are plotted in Figure 3.1. On the original scale (left), the contour plot of the two-parameter likelihood (3.2) is highly nonnormal, having a banana-shaped peak around the Maximum Likelihood Estimate and a sharp ridge along the axes, marked by the dashed line. Below the contour plot, the one-parameter version of this dashed line is also highly nonnormal. This is the profile likelihood (3.4) that can be obtained by maximizing (3.2) over all σ^2 given θ. The dotted line is an unnormalized normal density function centered on the MLE of the range parameter with variance set to the negative inverse of the second derivative of the log profile likelihood at the MLE. We can see that this normal approximation of $L(\theta)$ is a poor approximation of the profile likelihood.

In contrast, on the log scale (right), the contours are more ellipsoidal, suggesting less nonnormality. Below that, the difference is even more striking for the profile likelihood (dashed) that is virtually indistinguishable from its normal approximation (dotted) over the domain of $\log \theta$ shown (corresponding to the domain of θ on the left). At first look it may not be apparent that there are two separate lines in this plot (one dashed and one dotted) that overlap almost perfectly.

In Chapter 2, we used the log transformation to quantify prediction uncertainty for $d = 1, \ldots, 10$. We follow this reparameterization in this chapter for both the process variance and the range parameters. All of these parameters are positive-valued and here we work with all of them on the log scale for estimation purposes.

The nonnormality measure of Kass and Slate (1994) used in Chapter 2 indicated that although the log transformation was nearly optimal in higher dimensions, this was not necessarily the case in low dimensions. This was especially apparent for $d = 1$ and we decided to double-check that finding using a different measure that is the subject of the next subsection.

3.2.5 One-dimensional special case

For a scalar θ, let $l(\theta) = \log L(\theta)$ denote the logarithm of the profile likelihood, $\hat{\theta}$ the Maximum Likelihood Estimate (MLE) of θ, and $l''(\hat{\theta})$ and $l'''(\hat{\theta})$ the second- and third-derivatives of $l(\theta)$ at the MLE, respectively. Following Sprott (1973), define the Expected nonnormality (ENN) measure for θ:

$$\text{ENN for } \theta \;=\; |\, El'''(\hat{\theta})\,(-El''(\hat{\theta}))^{-\frac{3}{2}}\,|.$$

The intuition is that the expectation of the third derivative standardized by the expectation of the second derivative measures the deviation from normality (see Appendix A for taking expectations). This measure is appropriate when one wishes to consider a family of possible likelihoods without conditioning on any particular data set. Sprott (1973) also provided a formula that quantifies the effect of a transformation ϕ on nonnormality, where ϕ is a twice differentiable function of θ. After the ϕ transformation,

$$\text{ENN for } \phi(\theta) \;=\; \left| El'''(\hat{\theta})\,(-El''(\hat{\theta}))^{-\frac{3}{2}} \;+\; \frac{3\,\phi''(\hat{\theta})}{\phi'(\hat{\theta})\,(-El''(\hat{\theta}))^{\frac{1}{2}}} \right|,$$

where the first term inside the absolute value is the same as before in the definition of the ENN for θ and the second term is the effect of the transformation ϕ. As in Chapter 2, we use the family of power transformations originally explored by Tukey (1957) and later extended by Box and Cox (1964), indexed by a real λ that includes no transformation (for $\lambda = 1$) and the log transformation (for $\lambda = 0$) for the positive θ range parameter:

$$\phi(\theta) \;=\; \begin{cases} \theta^{\lambda}, & \lambda \neq 0, \\ \log\theta, & \lambda = 0. \end{cases}$$

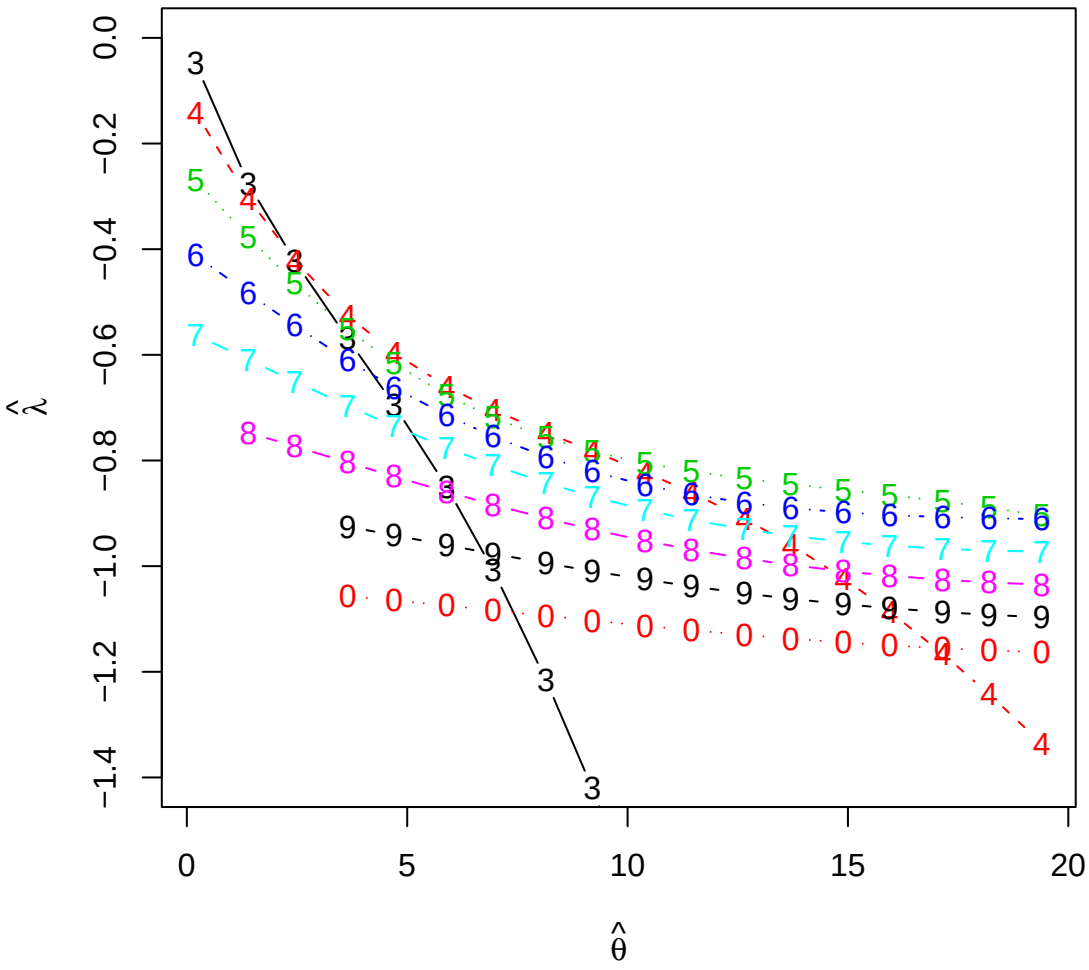


Figure 3.2: Optimal $\hat{\lambda}$ values for $d = 1$ and $n = 3, \ldots, 10$ as a function of $\hat{\theta}$. The digits $3, \ldots, 9$ in the plot represent the design sample size n and the digit 0 represents $n = 10$. The lines for $n = 8$, $n = 9$, and $n = 10$ do not start on the left side of the plot because of numerical difficulties for small $\hat{\theta}$.

The equation ENN for $\phi(\theta) = 0$ has the following solution for λ:

$$\hat{\lambda} = 1 + \frac{\hat{\theta}\, El'''(\hat{\theta})}{3\, El''(\hat{\theta})}.$$

The optimal $\hat{\lambda}$ values for $d = 1$, $n = 3, \ldots, 10$, and $\hat{\theta}$ between 0.2 and 20 are plotted in Figure 3.2. Unlike the simulations where we use random Latin hypercubes (McKay, Beckman, and Conover, 1979), the design here is fixed and equally spaced: $\{\, i/(n-1) : i = 0, \ldots, n-1 \,\}$. Note that some values are missing for $n = 8$, 9, and 10 because of ill-conditioned correlation matrices that could not be inverted for small $\hat{\theta}$ (see Appendix A for more details).

Now we can see that it is no coincidence that the normal approximation of the profile likelihood for the one-dimensional example in Figure 3.1 is so good on the log scale, since $\hat{\lambda}$ is close to zero for small $\hat{\theta}$ when $n = 3$. The values in this plot are also consistent with the box-plots for $d = 1$ in Figures 2.1 and 2.2 in Chapter 2, indicating that the optimal $\hat{\lambda}$ can be substantially less than zero for larger $\hat{\theta}$ or larger n. This also suggests an explanation to the anomaly why the results of the FBI in the $d = 1$ case are often less satisfactory than the results for $d > 1$ when trying to quantify prediction uncertainty in Chapter 2 and Nagy et al. (2007b). We will see that the $d = 1$ case is also quite special with respect to parameter estimation when we present our results in Section 3.4.

3.3 Simulations

To be able to compare prediction uncertainty assessments in Chapter 2 with parameter uncertainty assessments in this chapter, we replicated the simulations in Chapter 2, setting the common range parameter $\theta = 25/(d+1)^2$ and the sample size $n = 10\,d$ for the first set, and $\theta = 5/(d+1)^2$, $n = 5\,d$ for the second set ($d = 1, \ldots, 10$). The process variance σ^2 was fixed at constant 1 for all 20 simulations. To obtain 1,000 replicates for a given

combination of the sample size n and the range parameter θ, the following steps were repeated 1,000 times:

1. Select a random n point design by Latin hypercube sampling in $[0,\, 1]^d$ (McKay et al., 1979).

2. Generate a realization of the Gaussian process over the n points by setting σ^2 to 1 and θ_i to θ for $i = 1,\, \ldots,\, d$.

3. Find the MLE of the range parameters by numerically optimizing the log profile likelihood, and then apply formula (3.3) to get the MLE of σ^2.

4. Estimate the parameters together with standard errors based on the MLE and the observed information, i.e. standard errors were obtained by taking square roots of the diagonal elements of the negative inverse of the Hessian matrix of second derivatives evaluated at the MLE.

Using the notation $\xi = \log \sigma^2$ for the log transformed process variance and $\boldsymbol{\gamma} = (\log \theta_1,\, \ldots,\, \log \theta_d)^T = \log \boldsymbol{\theta}$ for the transformed $\boldsymbol{\theta}$ vector and $\sigma^2 = \exp \xi,\ \boldsymbol{\theta} = (\exp \gamma_1,\, \ldots,\, \exp \gamma_d)^T = \exp \boldsymbol{\gamma}$ for the inverse transformations, the Hessian at the MLE is $\nabla^2 \log L(\exp \hat{\xi},\, \exp \hat{\boldsymbol{\gamma}})$, where $\hat{\xi}$ and $\hat{\boldsymbol{\gamma}}$ are the maximum likelihood estimates of ξ and the vector $\boldsymbol{\gamma}$, respectively, and $\nabla^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})$ is defined as

$$\begin{pmatrix} \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \xi^2} & \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \xi\, \partial \gamma_1} & \cdots & \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \xi\, \partial \gamma_d} \\ \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \gamma_1\, \partial \xi} & \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \gamma_1^2} & \cdots & \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \gamma_1\, \partial \gamma_d} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \gamma_d\, \partial \xi} & \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \gamma_d\, \partial \gamma_1} & \cdots & \frac{\partial^2 \log L(\exp \xi,\, \exp \boldsymbol{\gamma})}{\partial \gamma_d^2} \end{pmatrix}.$$

The observed information matrix is the negative Hessian matrix of second derivatives evaluated at the MLE: $-\nabla^2 \log L(\exp \hat{\xi},\, \exp \hat{\boldsymbol{\gamma}})$. In Section 3.5 we describe well-known asymptotic methods using the inverse of this matrix to quantify the uncertainty in the estimation of the parameters ξ and γ_i

($i = 1, \ldots, d$). But before that, in the next section, first we present a summary of the results, showing that on the log scale these methods work quite well for finite sample sizes.

Summary statistics for the 1,000 replicates were obtained for the log transformed model parameters. Here we outline the quantities calculated for the estimator of the log transformed process variance $\xi = \log \sigma^2$. (Similar summaries are presented for the other estimators in Section 3.4).

1. The average estimate for ξ over the 1,000 replicates is given by

$$\bar{\xi} = \frac{1}{1000} \sum_{i=1}^{1000} \hat{\xi}^{(i)},$$

   where $\hat{\xi}^{(i)}$ is estimated from the ith data set ($i = 1, \ldots, 1000$).

2. The bias of the estimator $\hat{\xi}$ is estimated by subtracting the real value from the mean estimate:

$$\text{Bias}_{\hat{\xi}} = \bar{\xi} - \xi.$$

3. A p-value is attached to this bias by doing a two-sided, one-sample t-test on $\{ \hat{\xi}^{(1)} - \xi, \ldots, \hat{\xi}^{(1000)} - \xi \}$, to test if it is significantly different from zero.

4. The sample variance of the estimator $\hat{\xi}$ is:

$$\text{Variance}_{\hat{\xi}} = \frac{1}{999} \sum_{i=1}^{1000} \left(\hat{\xi}^{(i)} - \bar{\xi} \right)^2.$$

5. The Mean Squared Error (MSE) of $\hat{\xi}$ is:

$$\text{MSE}_{\hat{\xi}} = \frac{1}{1000} \sum_{i=1}^{1000} \left(\hat{\xi}^{(i)} - \xi \right)^2.$$

6. The estimated Mean Squared Error ($\widehat{\text{MSE}}$) of $\hat{\xi}$ is the average of the first diagonal elements of the inverse observed information matrix:

$$\widehat{\text{MSE}}_{\hat{\xi}} \;=\; \frac{1}{1000}\sum_{i=1}^{1000}\Big[\,-\nabla^2 \log L\left(\exp\hat{\xi}^{(i)},\, \exp\hat{\boldsymbol{\gamma}}^{(i)}\right)\Big]^{-1}(1,1),$$

where $\hat{\xi}^{(i)}$ and $\hat{\boldsymbol{\gamma}}^{(i)}$ denote the MLE of ξ and $\boldsymbol{\gamma}$, respectively, estimated from the ith data set ($i = 1, \ldots, 1000$).

Coverage probabilities for confidence intervals, credible intervals, and multi-dimensional confidence regions were also calculated by counting how many of the 1,000 replicates were covered by the $100(1-\alpha)\%$ confidence or credible sets for $\alpha = 0.01,\ 0.02,\ \ldots,\ 0.99$. Section 3.5 provides more details about these procedures after presenting the results in the next section.

3.4 Results

3.4.1 Point estimation

The range parameters are estimated by maximum likelihood. Tables 3.1 and 3.2 summarize the results of estimating the first parameter of the log transformed $\boldsymbol{\theta}$ vector: $\log\theta_1$. All numbers are on the log scale. The first feature that jumps out from both tables is the negative bias that, judging by the p-values, seems significant in all 20 cases, except for $d = 2$ in Table 3.1. This means that correlations between the responses have a tendency to appear stronger than they really are. However, considering the magnitude of the variance, this bias is relatively unimportant (i.e. statistical significance does not necessarily imply practical significance).

In Table 3.1, the $\widehat{\text{MSE}}$ column slightly underestimates the MSE column, but overall it is fairly close, meaning that it can measure well the uncertainty in the point estimation when $n = 10\,d$. But when $n = 5\,d$ (Table 3.2), the $\widehat{\text{MSE}}$ measure can become unstable numerically, as evidenced by the inflated numbers for $d = 5, 6, 8, 9, 10$ in Table 3.2. This is caused by

d	Real	Bias	p-value	Variance	MSE	$\widehat{\text{MSE}}$
1	1.833	-0.019	7.51e-05	0.023	0.023	0.018
2	1.022	-0.012	0.126	0.059	0.059	0.054
3	0.446	-0.051	6.73e-07	0.104	0.106	0.088
4	0.000	-0.052	5.57e-06	0.132	0.135	0.114
5	-0.365	-0.057	9.19e-06	0.161	0.164	0.136
6	-0.673	-0.048	0.000204	0.165	0.168	0.151
7	-0.940	-0.066	2.21e-06	0.193	0.198	0.170
8	-1.176	-0.051	0.000566	0.222	0.224	0.178
9	-1.386	-0.086	2.45e-08	0.235	0.242	0.199
10	-1.577	-0.092	4.59e-10	0.214	0.222	0.203

Table 3.1: MLE of $\log \theta_1$ in the first simulation study ($n = 10\,d$).

d	Real	Bias	p-value	Variance	MSE	$\widehat{\text{MSE}}$
1	0.223	-0.070	6.71e-05	0.304	0.309	0.161
2	-0.588	-0.081	1.77e-05	0.349	0.356	0.243
3	-1.163	-0.103	9.49e-07	0.440	0.450	0.312
4	-1.609	-0.142	1.88e-10	0.489	0.509	0.375
5	-1.974	-0.169	2.35e-08	0.906	0.934	57.134
6	-2.282	-0.172	8.96e-07	1.217	1.246	98.377
7	-2.549	-0.183	4.22e-12	0.680	0.713	0.481
8	-2.785	-0.244	8.49e-09	1.758	1.815	40.899
9	-2.996	-0.206	3.75e-07	1.621	1.662	28.889
10	-3.186	-0.259	5.78e-09	1.950	2.016	20.442

Table 3.2: MLE of $\log \theta_1$ in the second simulation study ($n = 5\,d$).

d	Real	Bias	p-value	Variance	MSE	$\widehat{\text{MSE}}$
1	0.000	-0.127	4.2e-08	0.525	0.540	0.435
2	0.000	-0.108	2.87e-10	0.286	0.297	0.290
3	0.000	-0.052	0.00055	0.227	0.230	0.243
4	0.000	-0.047	0.00257	0.238	0.240	0.220
5	0.000	-0.036	0.013	0.209	0.211	0.207
6	0.000	-0.030	0.0316	0.201	0.201	0.200
7	0.000	-0.024	0.0956	0.205	0.205	0.190
8	0.000	-0.025	0.0585	0.176	0.176	0.182
9	0.000	-0.027	0.0469	0.184	0.184	0.177
10	0.000	-0.017	0.184	0.166	0.166	0.172

Table 3.3: MLE of $\log \sigma^2$ in the first simulation study ($n = 10\,d$).

d	Real	Bias	p-value	Variance	MSE	$\widehat{\text{MSE}}$
1	0.000	-0.286	1.3e-13	1.450	1.530	0.895
2	0.000	-0.201	6.47e-12	0.835	0.875	0.692
3	0.000	-0.164	2.28e-09	0.740	0.766	0.612
4	0.000	-0.119	1.05e-06	0.588	0.601	0.572
5	0.000	-0.097	5.07e-05	0.562	0.571	0.551
6	0.000	-0.106	8.1e-06	0.558	0.568	0.543
7	0.000	-0.096	8.15e-05	0.595	0.604	0.540
8	0.000	-0.133	2.32e-07	0.649	0.666	0.532
9	0.000	-0.120	5.12e-07	0.563	0.577	0.526
10	0.000	-0.114	1.42e-06	0.552	0.564	0.511

Table 3.4: MLE of $\log \sigma^2$ in the second simulation study ($n = 5\,d$).

d	Real	Bias	p-value	Variance	MSE	$\widehat{\text{MSE}}$
1	0.000	-0.082	0.000537	0.561	0.567	0.322
2	0.000	-0.017	0.314	0.292	0.292	0.222
3	0.000	0.051	0.000888	0.233	0.235	0.189
4	0.000	0.056	0.000332	0.244	0.247	0.174
5	0.000	0.062	2.4e-05	0.215	0.218	0.164
6	0.000	0.065	6.33e-06	0.206	0.210	0.159
7	0.000	0.061	2.45e-05	0.209	0.213	0.153
8	0.000	0.055	3.31e-05	0.175	0.178	0.145
9	0.000	0.051	0.000195	0.187	0.190	0.143
10	0.000	0.058	9.9e-06	0.170	0.174	0.140

Table 3.5: Bayes estimate of $\log \sigma^2$ in the first simulation study ($n = 10\,d$).

d	Real	Bias	p-value	Variance	MSE	$\widehat{\text{MSE}}$
1	0.000	-0.224	2.75e-08	1.600	1.648	0.798
2	0.000	-0.035	0.234	0.852	0.852	0.588
3	0.000	0.026	0.352	0.756	0.756	0.527
4	0.000	0.073	0.00309	0.601	0.605	0.479
5	0.000	0.099	4.25e-05	0.584	0.593	0.466
6	0.000	0.062	0.011	0.599	0.602	0.459
7	0.000	0.063	0.0126	0.641	0.644	0.461
8	0.000	0.008	0.767	0.736	0.735	0.488
9	0.000	-0.016	0.524	0.657	0.657	0.503
10	0.000	-0.034	0.182	0.658	0.658	0.517

Table 3.6: Bayes estimate of $\log \sigma^2$ in the second simulation study ($n = 5\,d$).

extreme uncertainty in some cases, when the likelihood surface at the mode is essentially flat in certain directions (i.e. the MLE is on a high-dimensional ridge), and near-zero second derivatives can lead to inflated inverses, rendering the $\widehat{\text{MSE}}$ measure effectively useless. One way to remedy this situation is to detect outliers and eliminate them from the $\widehat{\text{MSE}}$ statistic. However, judging what is an outlier caused by numerical issues and what is not is inherently subjective. In the next subsection we present a better way to assess parameter uncertainty graphically, instead of just relying on a single number.

The process variance can also be estimated by maximum likelihood. Tables 3.3 and 3.4 contain the results for the MLE of $\log \sigma^2$ for the first set of simulations with adequate sample size ($n = 10\,d$), and the second set with limited sample size ($n = 5\,d$), respectively. Again, the numbers in the bias columns are all negative without exception; however, the evidence is not as strong for the first set: there are quite a few relatively large p-values in Table 3.3 for large d (which also implies large n, since $n = 10\,d$ in this case).

Hence, we can conclude that the negative bias is significant for all but the largest sample sizes. This finding is consistent with the simulation study in Mardia and Marshall (1984), but appears to contradict the simulations in Abt and Welch (1998), where no negative bias was reported for the MLE of σ^2 in one or two dimensions (this may be because of the much larger sample sizes: $n = 14$ for $d = 1$ and $n = 64$ for $d = 2$).

We also developed a Bayes estimator for the process variance based on FBI, taking into account the uncertainty in the estimation of the parameters, as described in Section 3.5. Its performance is given in Tables 3.5 and 3.6 that enable direct comparison with the MLE. The most important difference is that with the exception of the $d = 1$ case, there is either no evidence that the Bayes estimator is biased, or when there is, the bias is positive. Even in the one-dimensional case, the estimated negative bias is less severe than that of the MLE.

This may also provide an insight into how the FBI corrects the deficiency of the traditional plug-in method: by refusing to accept the too small MLE of σ^2, it constructs its own estimates that, on average, do not severely underestimate the real σ^2 for large d. As usual, the $d = 1$ case is again an exception, retaining a significant negative bias in Tables 3.5 and 3.6.

3.4.2 Parameter uncertainty

As we already mentioned in the previous section, one way of quantifying the uncertainty in the estimation of the parameters is by comparing the MSE and $\widehat{\text{MSE}}$ columns. (This seems feasible for all six tables presented so far, except Table 3.2 that has inflated $\widehat{\text{MSE}}$ numbers in higher dimensions). If the two numbers are close, we would expect that normality-based (Wald-type) confidence intervals using the standard errors would have good frequentist properties. In other words, validity would be demonstrated by confidence intervals whose actual coverage is approximately equal to the nominal coverage.

But why not make that match (or the lack of it) more explicit? To visualize how good that match is, in Figures 3.3 and 3.4 we plotted nominal coverage levels (from 1% to 99%) vs. the true coverage for the three kinds of estimators for $d = 1$, 4, 7, and 10. These (frequentist) coverage probabilities were calculated by counting how many times the real values were covered out of 1,000 realizations (replicates). In addition, a gray diagonal is also shown in the middle of each plot to help guide the eye: the closer the curves are to the diagonal, the better the match between nominal coverage (horizontally) and true coverage (vertically). This way of plotting is robust with respect to outliers, since a few inflated standard errors (out of 1,000) will have only a negligible effect on the estimated true coverage probabilities.

On a technical note, we should also mention that although we used the abbreviation CI in these plots for both frequentist confidence intervals (based on the likelihood) and for Bayesian credible intervals (based on the poste-

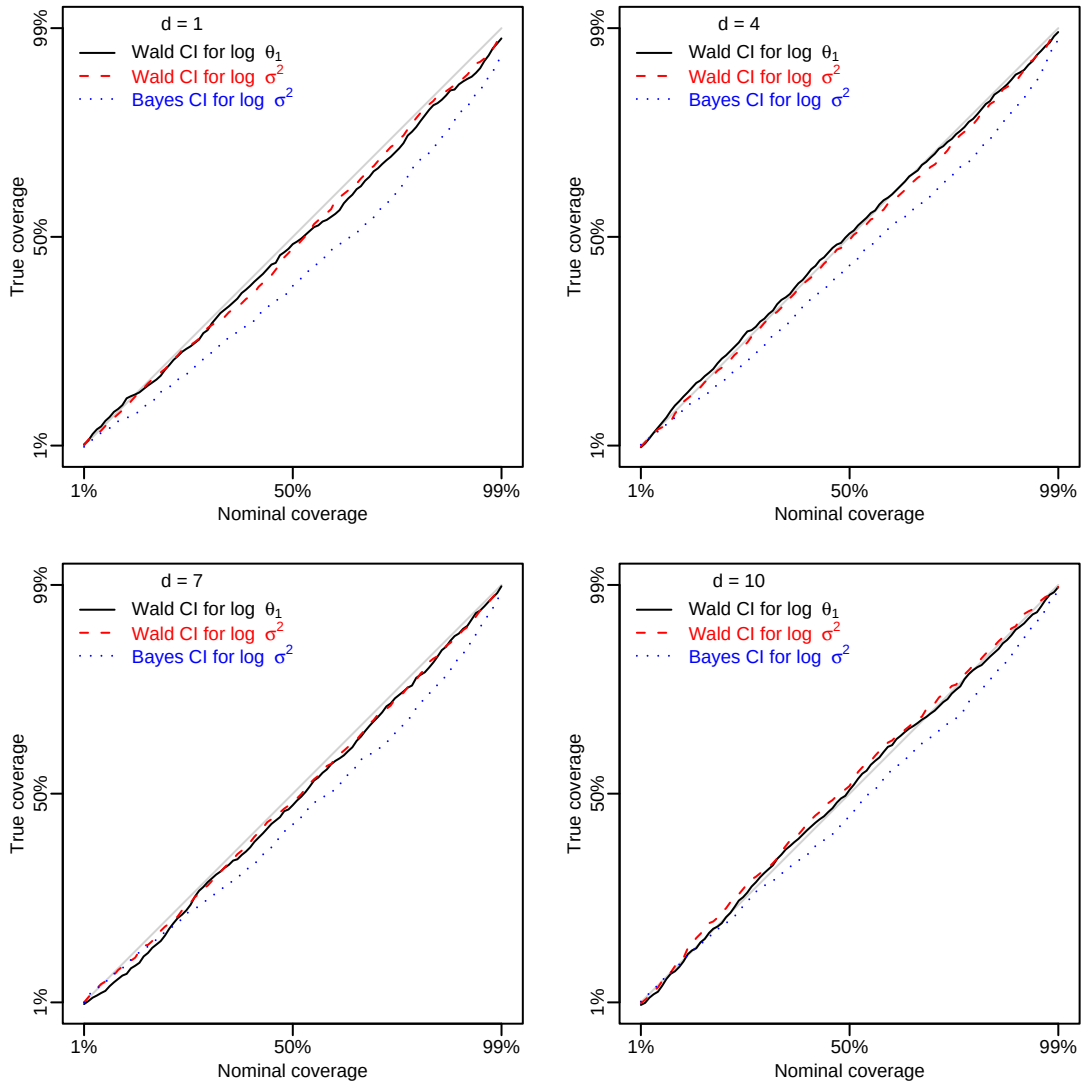


Figure 3.3: Coverage probabilities of Wald confidence intervals and Bayes credible intervals in the first simulation study ($n = 10\,d$) for $d = 1$, 4, 7, and 10.

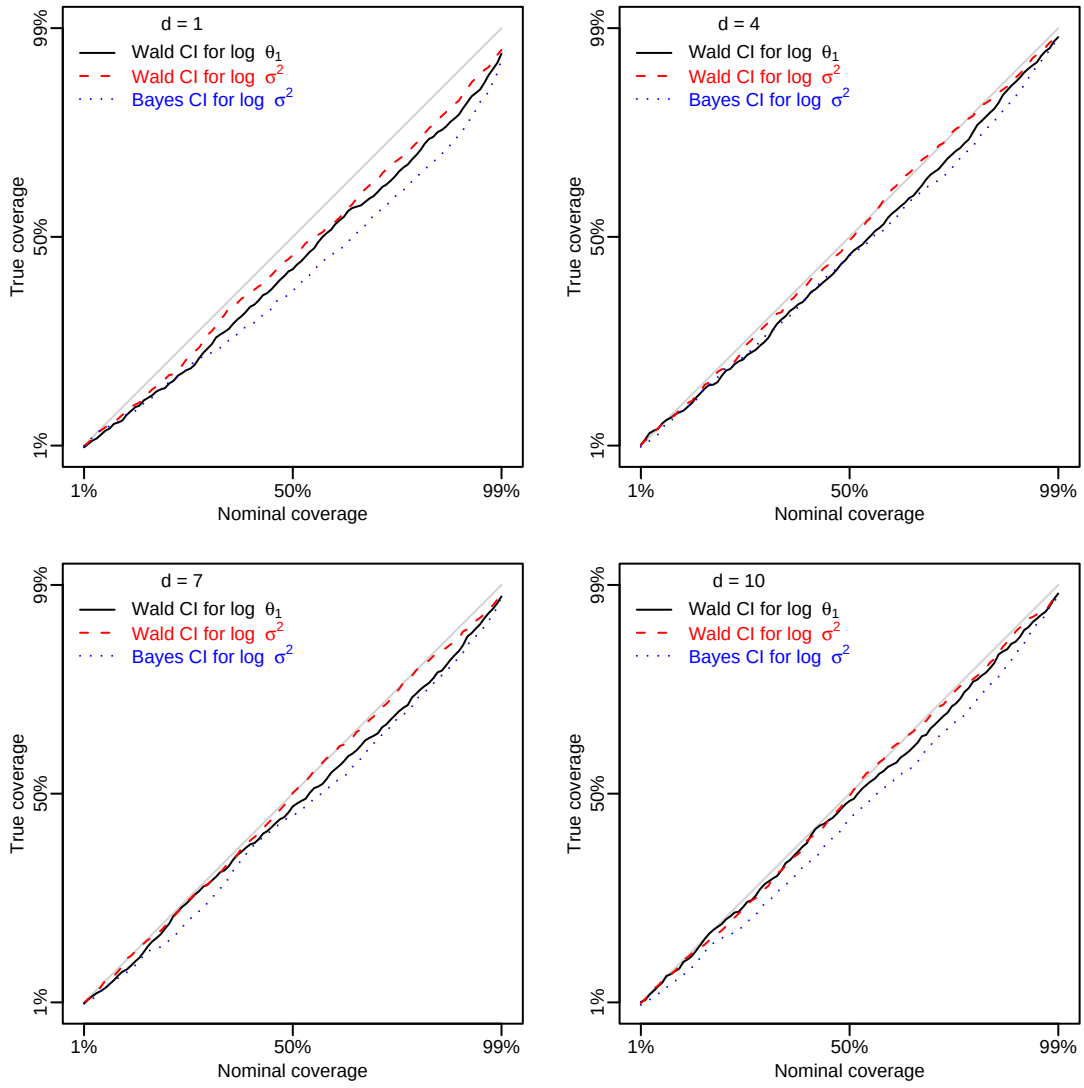


Figure 3.4: Coverage probabilities of Wald confidence intervals and Bayes credible intervals in the second simulation study ($n = 5\,d$) for $d = 1$, 4, 7, and 10.

rior), they have very different interpretations. But we can still evaluate the frequentist properties of these intervals, regardless of what assumptions we made when we derived them. It is not uncommon that a credible interval provides similar matching probabilities to its frequentist counterpart. Indeed, that is what we can see in this case, too, although the coverage of the credible intervals centered on the Bayes estimate are clearly less than that of the Wald confidence intervals centered on the MLE of $\log \sigma^2$ that are very close to the diagonal.

Overall what we can see in Figures 3.3 and 3.4 is that Wald confidence intervals had a good match in the second simulation study and almost perfect match (following the diagonal) in the first study. Also, note that this is indeed a robust way of visualizing uncertainty assessments, not as vulnerable to a few excessive outliers as the $\widehat{\text{MSE}}$ measure in the tables.

Results for $d = 2, 3, 5, 6, 8, 9$ were similar (not shown here). That suggests that the log transformation is nearly optimal not only with respect to assessing prediction uncertainty by FBI, but also with respect to quantifying parameter uncertainty by normality-based confidence intervals. But when we attempt to derive joint confidence regions for all the parameters, a less rosy picture emerges.

Figures 3.5 and 3.6 depict the results for the same four cases ($d = 1$, 4, 7, and 10) that we have seen before in Figures 3.3 and 3.4. But this time the coverage probabilities are for likelihood-based confidence regions for the joint likelihood $L(\sigma^2, \boldsymbol{\theta})$ with $d + 1$ parameters, the profile likelihood $L(\boldsymbol{\theta})$ with d parameters, and their normal approximations, respectively (the exact procedures for these calculations are given in the next section where we will also show that the confidence regions based on the normal approximations are equivalent to Wald-type confidence sets that are easier to compute than the original likelihood-based ones).

Although in Figure 3.5 we can only see moderate mismatch between the nominal and true coverages, that gap grows larger in Figure 3.6, especially

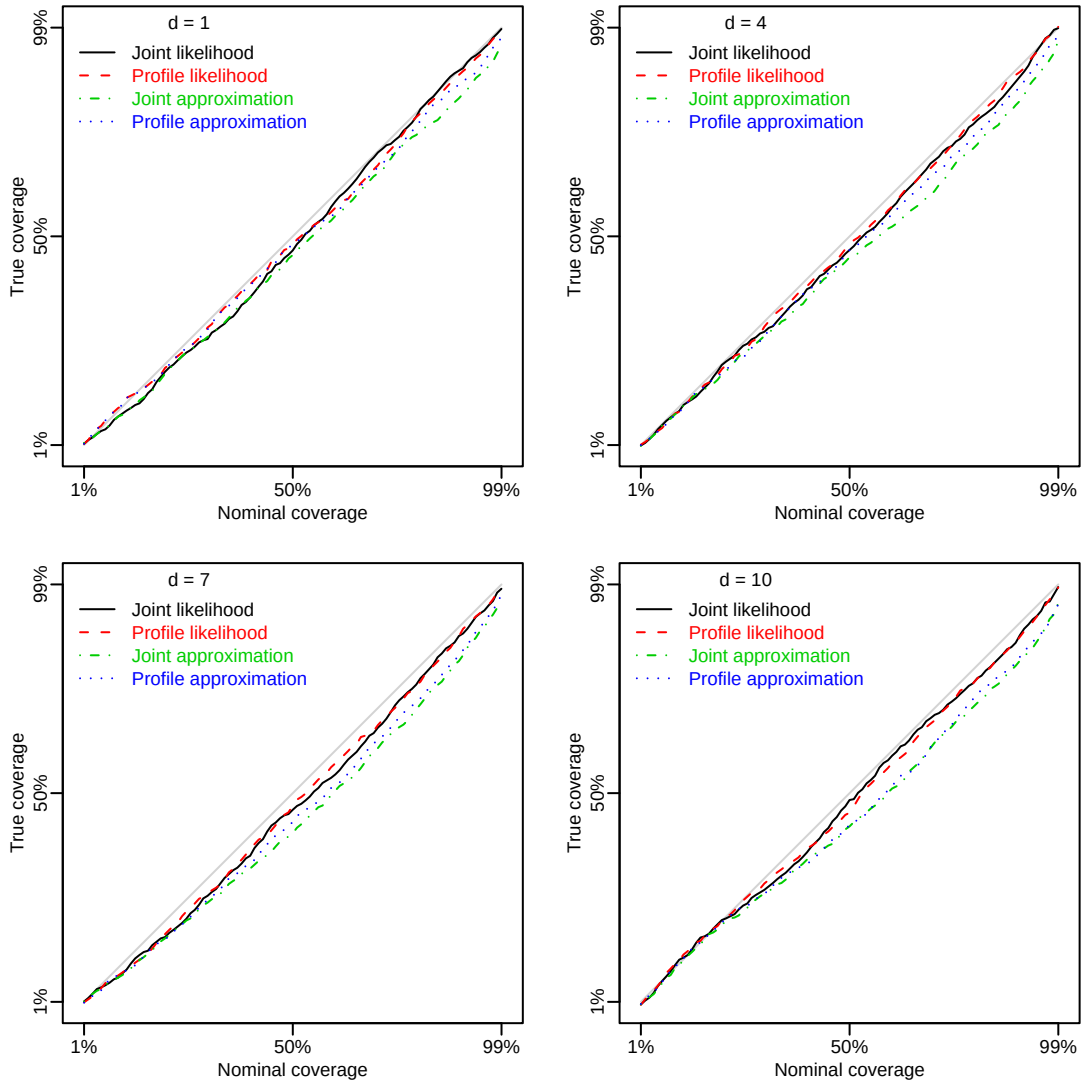


Figure 3.5: Coverage probabilities of confidence regions based on the two likelihood functions and their normal approximations in the first simulation study ($n = 10\,d$) for $d = 1,\ 4,\ 7,$ and 10.

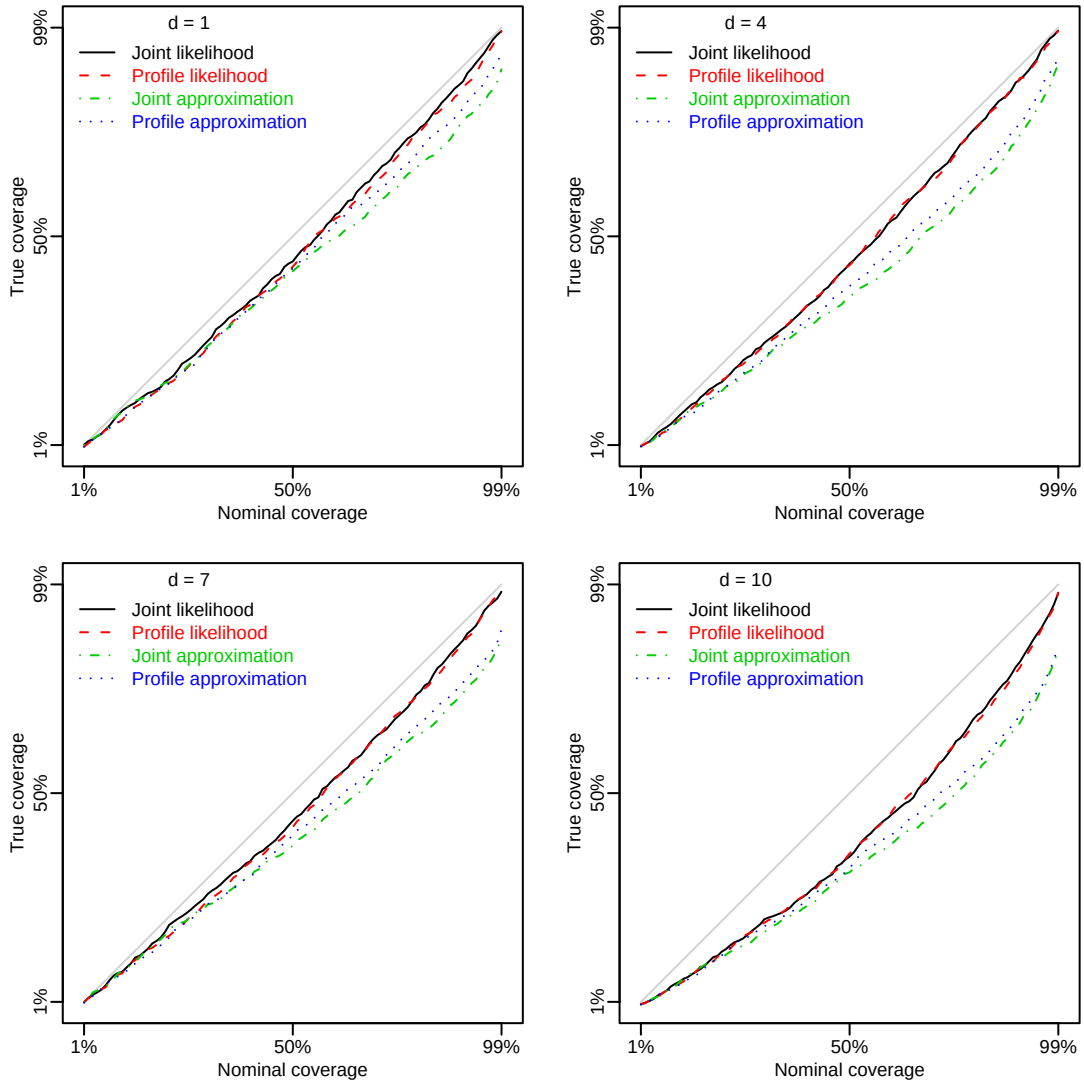


Figure 3.6: Coverage probabilities of confidence regions based on the two likelihood functions and their normal approximations in the second simulation study ($n = 5\,d$) for $d = 1,\ 4,\ 7,$ and 10.

in higher dimensions (e.g. $d = 10$). Not only do the curves for the normal approximations fall short of the diagonal (indicating serious undercoverage), but the ones for the likelihoods do as well, which means that there is insufficient information in the data to satisfactorily quantify the uncertainty in the maximum likelihood estimation.

These two figures can also serve to visualize the nonnormality of the two likelihood functions in a way that is superior to measures that compress all information about the shape of the function into a single real number. For instance, the nonnormality measures of Sprott (1973) and their multivariate extensions by Kass and Slate (1994) can only provide limited information about tail behavior (Slate, 1991) because they are based solely on the curvature at the mode.

In contrast, the figures show not only what happens in the neighborhood of the mode, say, at the 1% nominal confidence level, but also what happens in the tails, say, at the 99% level. We can see that the true coverage for the normal approximations start out very close to the original, indicating a nearly normal shape in the neighborhood of the MLE. However, as the nominal level increases on the x-axis, the gap between the approximation and its original progressively gets larger on the y-axis, meaning that the approximation is less accurate in the tails.

Figure 3.6 also indicates that the FBI does not propagate through all parameter uncertainty (as specified by the likelihood) into prediction uncertainty. The gap between the profile likelihood (dashed line) and its normal approximation (dotted line) is substantial between the 50% and 99% confidence levels, meaning that the normal approximation (that is used as the posterior distribution by the FBI) not only diminishes the tails, but overall it is much more concentrated around the mode than the original profile likelihood. Comparison of Figures 3.5 and 3.6 show, however, that the inference from the normal approximation improves substantially as the sample size increases from $n = 5d$ to $n = 10d$.

We can also observe that the approximation for the profile likelihood (dotted line) lies closer to the diagonal than the approximation for the joint likelihood (dashed-and-dotted line), illustrating an additional benefit of profiling. The difference may not appear to be large; however, from the prediction perspective, this seems justified considering that profiling also allows one to disentangle the uncertainty in the estimation of the relatively unimportant process variance parameter from the uncertainty in the estimation of the more important range parameters.

In summary, it appears that using normal approximations for the two likelihood functions on the log scale to quantify parameter uncertainty may be acceptable with adequate sample size (like $n = 10\,d$ in our first simulation study), but may result in confidence regions with serious undercoverage for smaller samples (like $n = 5\,d$ in the second simulation study). On the other hand, confidence intervals for the maximum likelihood estimates of individual parameters are much more robust in terms of coverage probabilities, retaining surprisingly good matching coverage even for smaller samples (except for $d = 1$). Although the coverage of the Bayes credible intervals for $\log \sigma^2$ is less accurate, it seems robust to smaller sample sizes.

3.5 Methods

3.5.1 Likelihood-based estimators

We estimate the process variance σ^2 and the d range parameters in the $\boldsymbol{\theta}$ vector by maximum likelihood (see Mardia and Marshall (1984) for regularity conditions for the consistency and asymptotic normality of these estimators). Since these are not available in a closed form, the optimization must be done numerically, e.g. by maximizing the logarithm of the joint likelihood function $L(\sigma^2, \boldsymbol{\theta})$. Alternatively, one can optimize the log profile likelihood $\log L(\boldsymbol{\theta})$ to find the MLE of $\boldsymbol{\theta}$ and then use equation (3.3) to get the MLE of σ^2, as we did (see Welch et al. (1992) for optimization-related issues). Differ-

ent parameterizations might affect the optimization process differently, but if done correctly, the end result should be the same because of the invariance of the MLE. (We consistently used the log transformation for everything in this chapter, including maximizing the log profile likelihood).

Based on asymptotic theory, the joint likelihood $L(\sigma^2, \boldsymbol{\theta})$ can also be used to derive confidence sets for all the $d+1$ parameters jointly by inverting the likelihood ratio test. For example, following Meeker and Escobar (1995), an approximate $100(1-\alpha)\%$ likelihood-based confidence region for $(\sigma^2,\ \boldsymbol{\theta})$ is the set of all values of $(\sigma^2,\ \boldsymbol{\theta})$ such that

$$-2\log\left(\frac{L(\sigma^2,\ \boldsymbol{\theta})}{L(\hat{\sigma}^2,\ \hat{\boldsymbol{\theta}})}\right) \ < \ \chi^2_{(1-\alpha;\ d+1)}, \tag{3.5}$$

where $\hat{\sigma}^2$ and $\hat{\boldsymbol{\theta}}$ denote the MLE of the parameter σ^2 and the parameter vector $\boldsymbol{\theta}$, respectively, and $\chi^2_{(1-\alpha;\ d+1)}$ is the $1-\alpha$ quantile of the chi-square distribution with $d+1$ degrees of freedom.

Similarly, the profile likelihood function $L(\boldsymbol{\theta})$ can yield confidence sets for the d range parameters jointly. An approximate $100(1-\alpha)\%$ likelihood-based confidence region for $\boldsymbol{\theta}$ is the set of all values of $\boldsymbol{\theta}$ such that

$$-2\log\left(\frac{L(\boldsymbol{\theta})}{L(\hat{\boldsymbol{\theta}})}\right) \ < \ \chi^2_{(1-\alpha;\ d)}. \tag{3.6}$$

These confidence regions are also invariant to parameter transformations. However, in general, they are not guaranteed to have a nice shape and can be cumbersome to calculate numerically, especially in higher dimensions. Fortunately, in our case we did not have to calculate the boundaries of these regions explicitly, since we were only interested whether the true values were covered by them, and for that one needs to evaluate the likelihood at only two points: at the real value and at the MLE.

3.5.2 Normal approximations

For our normal approximations, we continue to use the MLEs as point estimates. But what happens to the confidence sets when we replace the likelihoods with their (multivariate) normal approximations? We get nicely shaped, symmetric confidence ellipsoids centered on the MLE (in $d+1$ dimensions for all parameters jointly or in d dimensions for the $\boldsymbol{\theta}$ vector).

In this case the boundaries can be calculated analytically, but there is a trade-off for computational simplicity. We lose invariance to transformations and we also lose accuracy in terms of coverage probabilities. This is especially evident in Figures 3.5 and 3.6 if we compare the actual coverage for the two original likelihood functions vs. their approximations.

Here we describe the approximation procedure only for the d-dimensional log transformed $\boldsymbol{\theta}$ vector, since the $(d+1)$-dimensional case is completely analogous with an extra log transformed σ^2 parameter. Using the same notation as in Section 3.3, namely $\boldsymbol{\gamma} = (\log\theta_1, \ldots, \log\theta_d)^T = \log\boldsymbol{\theta}$ for the transformed $\boldsymbol{\theta}$ vector and $\boldsymbol{\theta} = (\exp\gamma_1, \ldots, \exp\gamma_d)^T = \exp\boldsymbol{\gamma}$ for the inverse transformation, we expect the transformed profile likelihood function $L(\exp\boldsymbol{\gamma})$ to have a shape that is more Gaussian with respect to $\boldsymbol{\gamma}$ than the shape of the original $L(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$.

First we maximize $\log L(\exp\boldsymbol{\gamma})$ to get the MLE of $\boldsymbol{\gamma}$, denoted $\hat{\boldsymbol{\gamma}}$. Then we compute the Hessian matrix of second derivatives of $\log L(\exp\boldsymbol{\gamma})$ at $\hat{\boldsymbol{\gamma}}$, denoted $H_{\hat{\boldsymbol{\gamma}}} = \nabla^2 \log L(\exp\hat{\boldsymbol{\gamma}})$, where

$$\nabla^2 \log L(\exp\boldsymbol{\gamma}) = \begin{pmatrix} \frac{\partial^2 \log L(\exp\boldsymbol{\gamma})}{\partial\gamma_1^2} & \cdots & \frac{\partial^2 \log L(\exp\boldsymbol{\gamma})}{\partial\gamma_1\,\partial\gamma_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 \log L(\exp\boldsymbol{\gamma})}{\partial\gamma_d\,\partial\gamma_1} & \cdots & \frac{\partial^2 \log L(\exp\boldsymbol{\gamma})}{\partial\gamma_d^2} \end{pmatrix}.$$

Then we can approximate $L(\exp\boldsymbol{\gamma})$ with the density function of the multi-

variate normal $N(\hat{\boldsymbol{\gamma}},\ -H_{\hat{\boldsymbol{\gamma}}}^{-1})$ distribution, which is proportional to

$$\left|-H_{\hat{\boldsymbol{\gamma}}}^{-1}\right|^{-\frac{1}{2}}\ \exp\left\{-\frac{1}{2}\,(\boldsymbol{\gamma}-\hat{\boldsymbol{\gamma}})^T\left[-H_{\hat{\boldsymbol{\gamma}}}^{-1}\right]^{-1}(\boldsymbol{\gamma}-\hat{\boldsymbol{\gamma}})\right\}.$$

Using this approximation instead of the profile likelihood function in inequality (3.6) leads to the confidence ellipsoid defined by the quadratic form

$$(\boldsymbol{\gamma}-\hat{\boldsymbol{\gamma}})^T\left[-H_{\hat{\boldsymbol{\gamma}}}\right](\boldsymbol{\gamma}-\hat{\boldsymbol{\gamma}})\ <\ \chi^2_{(1-\alpha;\,d)}.$$

This is the same as the quadratic form for the normal-theory Wald subset statistic

$$(\boldsymbol{\gamma}-\hat{\boldsymbol{\gamma}})^T\left[\hat{\Sigma}_{\hat{\boldsymbol{\gamma}}}\right]^{-1}(\boldsymbol{\gamma}-\hat{\boldsymbol{\gamma}}),$$

where $\hat{\Sigma}_{\hat{\boldsymbol{\gamma}}}$ is obtained by leaving out the row and column for the log transformed σ^2 parameter from the inverse of the observed information matrix (see Meeker and Escobar (1995) for a proof and also for a general discussion on the connection between profiling and constructing likelihood-based and Wald-type confidence regions).

With only one parameter, confidence ellipsoids become normality-based Wald confidence intervals, using the quantiles of the standard normal distribution with a standard error to provide confidence bounds symmetric about the MLE. For example, in terms of coverage probabilities, the assumption that $(\gamma_1-\hat{\gamma}_1)/\mathrm{StdErr}_{\hat{\gamma}_1}$ follows a $N(0,1)$ distribution is equivalent to assuming that $(\gamma_1-\hat{\gamma}_1)^T\left[\mathrm{StdErr}^2_{\hat{\gamma}_1}\right]^{-1}(\gamma_1-\hat{\gamma}_1)$ has a χ-squared distribution with one degree of freedom, where the standard error of $\hat{\gamma}_1$, denoted as $\mathrm{StdErr}_{\hat{\gamma}_1}$, is obtained by either taking the square root of $-H_{\hat{\boldsymbol{\gamma}}}^{-1}(1,1)$, or, equivalently, by taking the root on the diagonal for $\hat{\gamma}_1$ of the inverse of the observed information matrix. (The root of the appropriate diagonal element of the inverse of the observed information matrix was also used to obtain a standard error for $\log\hat{\sigma}^2$).

3.5.3 Bayes estimator

Fast Bayesian Inference uses the $N(\hat{\boldsymbol{\gamma}}, -H_{\hat{\boldsymbol{\gamma}}}^{-1})$ distribution as the posterior distribution of $\boldsymbol{\gamma}$. The main advantage is that it is straightforward to obtain independent, identically distributed (iid) Monte Carlo samples from this posterior: $\boldsymbol{\gamma}^{(1)}, \ldots, \boldsymbol{\gamma}^{(M)}$, and since they are iid, a relatively small sample size is sufficient (we used $M = 400$). For each $\boldsymbol{\gamma}^{(i)}$ in this sample, for prediction purposes, internally the FBI estimates σ^2 with $\hat{\sigma}^2(\exp \boldsymbol{\gamma}^{(i)})$, using equation (3.3). Note that the estimator function $\hat{\sigma}^2(\boldsymbol{\theta})$ in (3.3) is a function of the untransformed range parameter vector $\boldsymbol{\theta}$, so we need to use the inverse transformation for the log transformed $\boldsymbol{\gamma}^{(i)}$ vectors in the FBI sample ($i = 1, \ldots, M$). (Also: the notation $\boldsymbol{\gamma}^{(i)}$ used here should not be confused with the hatted $\hat{\boldsymbol{\gamma}}^{(i)}$ used earlier in Section 3.3).

Thus the Bayes estimate of $\log \sigma^2$ is

$$\frac{1}{M} \sum_{i=1}^{M} \log \hat{\sigma}^2(\exp \boldsymbol{\gamma}^{(i)}).$$

To derive a standard error for normality-based credible intervals, we can take the square root of the sample variance:

$$\frac{1}{M-1} \sum_{j=1}^{M} \left(\log \hat{\sigma}^2(\exp \boldsymbol{\gamma}^{(j)}) - \frac{1}{M} \sum_{i=1}^{M} \log \hat{\sigma}^2(\exp \boldsymbol{\gamma}^{(i)}) \right)^2 .$$

3.6 Concluding remarks

Our main finding for point estimation by maximum likelihood is that all parameters tend to be underestimated. We have introduced a Bayes estimator for the process variance that can reduce this negative bias for $d = 1$ and make it insignificant or even turn it positive for $d > 1$. Further work is needed to clarify how this less biased estimator can be exploited (besides quantifying prediction uncertainty in FBI). One possible avenue of investigation could be to fix σ^2 at the Bayes estimate and then see whether the

maximum likelihood estimation leads to improved estimates for the remaining parameters (which in turn could also be used to try to improve the Bayes estimate of σ^2, and so on).

In Chapter 2, the FBI method demonstrated that the log transformation was nearly optimal for assessing prediction uncertainty, since it left almost no room for further improvements in terms of matching coverage probabilities of the prediction bands. Likewise, we have shown that the log transformation is nearly optimal for quantifying parameter uncertainty, since the match between nominal and true coverages of the MLE-centered, normality-based Wald confidence intervals for the individual parameters are almost as good as those seen for the FBI prediction bands.

This also provides some insight into why the FBI is able to compute the uncertainty in the predictions so well. However, this is still not a fully satisfactory explanation in cases when we can get good matching coverage only for the Wald confidence intervals for each parameter separately, but not for the Wald confidence regions jointly.

Wald and likelihood ratio confidence regions are asymptotically equivalent (e.g. see Cox and Hinkley (1974) for a proof). However, for small samples, the Wald approximation is often inferior in terms of matching coverage probabilities. We have shown that for our random function model the difference can be quite substantial. It is an open question how much different reparameterizations could help to close this gap. We have seen that in the one-dimensional case, working on the log scale is not optimal in terms of profile likelihood nonnormality. Transformations that make the shape of the likelihood or the profile likelihood more Gaussian, perhaps adaptively (based on the data), might be worth exploring, since the only special case when Wald confidence regions are equivalent to likelihood-based ones for finite sample sizes is when the likelihood is proportional to a normal density function.

Finally, we should point out that although the log transformation is

rarely optimal in the one-dimensional special case according to the nonnormality measure used in subsection 3.2.5, that does not mean that it is not nearly optimal. On the contrary, results in Nagy et al. (2007a) suggest that overall, the log transformation is quite useful in most cases for reducing the nonnormality of the profile likelihood, and our results seem to support that, since the approximations for $d = 1$ in Figures 3.5 and 3.6 look no worse than the ones for $d > 1$.

Bibliography

Abt, M. (1999), "Estimating the Prediction Mean Squared Error in Gaussian Stochastic Processes with Exponential Correlation Structure," *Scandinavian Journal of Statistics*, 26, 563–578.

Abt, M. and Welch, W. J. (1998), "Fisher Information and Maximum Likelihood Estimation of Covariance Parameters in Gaussian Stochastic Processes," *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 26, 127–137.

Box, G. E. P. and Cox, D. R. (1964), "An Analysis of Transformations," *Journal of the Royal Statistical Society. Series B (Methodological)*, 26, 211–252.

Cox, D. R. and Hinkley, D. V. (1974), *Theoretical Statistics*, London: Chapman and Hall.

Cressie, N. A. C. (1993), *Statistics for Spatial Data*, John Wiley & Sons.

Currin, C., Mitchell, T., Morris, M., and Ylvisaker, D. (1991), "Bayesian Prediction of Deterministic Functions, with Applications to the Design and Analysis of Computer Experiments," *Journal of the American Statistical Association*, 86, 953–963.

Furrer, R. (2005), "Covariance estimation under spatial dependence," *Journal of Multivariate Analysis*, 94, 366–381.

Karuri, S. W. (2005), "Integration in Computer Experiments and Bayesian Analysis," Ph.D. thesis, University of Waterloo.

Kass, R. E. and Slate, E. H. (1994), "Some Diagnostics of Maximum Likelihood and Posterior Nonnormality," *The Annals of Statistics*, 22, 668–695.

Mardia, K. V. and Marshall, R. J. (1984), "Maximum Likelihood Estimation of Models for Residual Covariance in Spatial Regression," *Biometrika*, 71, 135–146.

McKay, M. D., Beckman, R. J., and Conover, W. J. (1979), "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, 21, 239–245.

Meeker, W. Q. and Escobar, L. A. (1995), "Teaching about Approximate Confidence Regions Based on Maximum Likelihood Estimation," *The American Statistician*, 49, 48–53.

Nagy, B., Loeppky, J. L., and Welch, W. J. (2007a), "Correlation parameterization in random function models to improve normal approximation of the likelihood or posterior," Tech. Rep. 229, Department of Statistics, The University of British Columbia.

— (2007b), "Fast Bayesian Inference for Gaussian Process Models," Tech. Rep. 230, Department of Statistics, The University of British Columbia.

Rasmussen, C. E. and Williams, C. K. I. (2006), *Gaussian Processes for Machine Learning*, MIT Press.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989), "Design and Analysis of Computer Experiments (C/R: P423-435)," *Statistical Science*, 4, 409–423.

Slate, E. H. (1991), "Reparameterizations of Statistical Models," Ph.D. thesis, Carnegie Mellon University.

Sprott, D. A. (1973), "Normal Likelihoods and Their Relation to Large Sample Theory of Estimation," *Biometrika*, 60, 457–465.

Stein, M. L. (1999), *Interpolation of Spatial Data: Some Theory for Kriging*, Springer-Verlag Inc.

Tukey, J. W. (1957), "On the Comparative Anatomy of Transformations," *The Annals of Mathematical Statistics*, 28, 602–632.

Wang, H. and Zhang, H. (2003), "On the Possibility of a Private Crop Insurance Market: A Spatial Statistics Approach," *Journal of Risk and Insurance*, 70, 111–124.

Welch, W. J., Buck, R. J., Sacks, J., Wynn, H. P., Mitchell, T. J., and Morris, M. D. (1992), "Screening, Predicting, and Computer Experiments," *Technometrics*, 34, 15–25.

Zhang, H. and Zimmerman, D. L. (2005), "Towards Reconciling Two Asymptotic Frameworks in Spatial Statistics," *Biometrika*, 92, 921–936.

Chapter 4

Discussion

In this thesis, we demonstrated a novel way to achieve approximately valid estimation and prediction inference for a particular statistical model frequently used in computer experiments to model smooth response surfaces. In general, there are two kinds of goals for statistical modeling: prediction and model identification. We followed that separation of concerns when dividing up our work into two separate publications. Although the two manuscripts share the same model and simulation design, Chapter 2 only deals with issues related to prediction and Chapter 3 with identifying (the covariance of) the random function model.

However, we do not just accept the model as it is, but seek nearly optimal reparameterizations to minimize nonnormality of the profile likelihood. Hence, another way to relate the two chapters is by looking at what kind of nonnormality measures they use. Chapter 2 employs a multivariate measure from Kass and Slate (1994) based on the curvature at the mode. Chapter 3 takes a more visual approach to compare likelihoods with their normal approximations and reveal discrepancies not only in the neighborhood of the mode but also in the tails. (The one-dimensional special case is also investigated by a univariate measure of Sprott (1973) in Chapter 3, subsection 3.2.5, that is the same as the "Expected Non-Normality" measure in Nagy, Loeppky, and Welch (2007a). Also, the multivariate measure in Kass and Slate (1994) for $d = 1$ reduces to the univariate "Observed Non-Normality" in Nagy et al. (2007a), also from Sprott (1973)).

Since the main goal in computer experiments is prediction, the contributions in Chapter 2 are arguably more important to this field than the results

in Chapter 3. The main advantage of Fast Bayesian Inference is that it is computationally efficient and can be implemented as a black box. Moreover, it can also relieve the user from the burden and responsibility of selecting a suitable prior or an appropriate MCMC algorithm. In other words, FBI has all the required ingredients that make it suitable for incorporation into a standard statistical package. That holds out the promise that one day it may become a widely used method across many fields in science and engineering.

In contrast, the results in Chapter 3 seem less interesting from the practical standpoint. One could even say that the significance of Chapter 3 lies mostly in providing some insights about why the FBI prediction bands are so accurate in Chapter 2, since precise assessments of parameter uncertainty can certainly help quantify prediction uncertainty, too. However, that is at most a partial explanation of the success of the FBI, since the prediction bands retain much of their accuracy even in extreme situations (such as very small sample sizes or extremely large range parameters) as shown in Nagy, Loeppky, and Welch (2007b) (see results included in Appendix B). Although these extremes are irrelevant in practice (since meaningful prediction is not possible), it is still an interesting theoretical question what makes the FBI so robust across such a wide range of settings.

One practical shortcoming of the thesis is that the estimated true coverage probabilities of the prediction bands are averaged over both the hypercube $[0,\ 1]^d$ and over all the simulated data sets. But it is never explored what happens at just one specific point in the input space or for just one particular realization, both of which could be more relevant in practice than the present blanket measure. And what if the data is not a realization of a Gaussian process? In practical applications, this is almost always the case, yet no attempt is made to assess the robustness of FBI with respect to model misspecifications. (One cannot make generalizations based on just two real examples in Chapter 2).

The simulation design can also be criticized on the grounds that it uses

only one fixed θ value for each dimension (although we should mention that we obtained similar results with Bayesian simulation designs where $\log \theta$ was drawn from either a uniform or normal distribution, but those results are not presented here). On the other hand, the simulations can also be considered the primary strength of this thesis, pushing the limits of both currently available hardware (WestGrid high performance computing facilities) and software (e.g. Intel's Math Kernel Library for matrix operations or ADOL-C for automatic differentiation in C++).

There are also many obvious extensions to our work, some of which seem easier to tackle than others. We briefly discuss some possible research directions and speculate on their perceived feasibility at the time of this writing.

4.1 Alternative correlation functions

Whether the FBI can be adopted for other covariance structures is one of the first questions that comes to mind. For instance, one possible generalization of the Gaussian correlation function is the Power Exponential family:

$$\mathrm{Corr}(Z(\boldsymbol{w}),\ Z(\boldsymbol{x}))\ =\ \prod_{i=1}^{d} \exp\{-\theta_i |w_i - x_i|^{p_i}\},$$

where $0 < p_i \leq 2$ (Sacks, Welch, Mitchell, and Wynn, 1989). In this thesis we only presented results for the $p_i = 2$ ($i = 1, \ldots, d$) special case that is known as the Squared Exponential or Gaussian correlation function. But we also experimented with running the FBI with constant exponents fixed at values other than 2. What we found was that the prediction uncertainty assessments were not as uniformly valid as for $p = 2$. More work is required to understand what causes the difference. Results for $d = 1$ in Nagy et al. (2007a) suggest that transformations may play a large role. (For example, the log transformation tends to reduce nonnormality for $p = 2$, but that is

less often the case for $p < 2$).

Perhaps the most exciting question is if the FBI can be extended to the situation when the parameters p_i are unknown. Unfortunately, it is not clear at this time how one might go about accomplishing that, since these parameters can only take values between 0 and 2 and the boundary case is very special. We have seen that the process can behave very differently (including losing differentiability), even for values of p that are only slightly less than 2, such as $p = 2 - 10^{-6}$.

Brian Williams suggested another way to generalize the Gaussian correlation that can be viewed as a limiting case of the Matérn class (Matérn, 1947) in terms of differentiability. This was also recommended earlier in Stein (1999). The rationale is that this family has a parameter ν for fine-tuning differentiability, i.e. the process is differentiable k times if and only if $k < \nu$, allowing much finer control than the Power Exponential family, for which the process is infinitely differentiable for $p = 2$ and not differentiable at all for $p < 2$.

4.2 Additional terms in the model

A general model could have a regression component and a white noise term in addition to the stochastic process $\mathrm{Z}(\boldsymbol{x})$:

$$Y(\boldsymbol{x}) \;=\; \sum_j \beta_j f_j(\boldsymbol{x}) \;+\; Z(\boldsymbol{x}) \;+\; \epsilon,$$

where each $f_j(\boldsymbol{x})$ is a function of $\boldsymbol{x}$ with known or unknown β_j coefficients and ϵ is iid random error parameterized by a known or unknown variance. In fact, when we began developing FBI, we started out with a model that included both an unknown mean μ and white noise with known or unknown variance

$$Y(\boldsymbol{x}) \;=\; \mu \;+\; Z(\boldsymbol{x}) \;+\; \epsilon.$$

Encouraging preliminary results for $d = 2$ were reported at the Annual Meeting of the Statistical Society of Canada in London, Ontario in May 2006, entitled "Uncertainty in kriging predictions with and without random error". Since it did not become more clear later how to deal with noise, we ended up dropping it from the model and turned our focus to the deterministic case:

$$Y(\boldsymbol{x}) = \mu + Z(\boldsymbol{x}),$$

having an unknown mean μ. FBI results for $d = 1, \ldots, 5$ comparable to the ones in Appendix B were presented at the Joint Statistical Meetings in Seattle, Washington in August 2006 with the title "Validity of Likelihood and Bayesian Inference for Gaussian Process Regression".

Eventually, we dispensed with μ, too, to simplify the algebra and speed up computations to be able to explore higher-dimensional cases and run long Markov chains. Since we found that the FBI prediction bands were remarkably accurate with or without μ in the model, it is not unreasonable to guess that one could get similarly good prediction uncertainty assessments for a model including more regression terms. The only required change in FBI would be that in addition to σ^2, one would also have to eliminate the extra regression coefficients when deriving the profile likelihood function for the remaining range parameters. All this is standard practice, described in detail in Sacks et al. (1989) or Welch, Buck, Sacks, Wynn, Mitchell, and Morris (1992).

4.3 Different reparameterizations

There is no reason to restrict oneself to the family of power transformations in Tukey (1957) as we did in this thesis. Although this class of functions is quite flexible and powerful, as shown by Box and Cox (1964), this choice is still arbitrary and can not be expected to provide a satisfying solution in every situation.

In the one-dimensional case we experimented briefly with the $\rho = e^{-\theta}$ reparameterization for the range parameter (Linkletter, Bingham, Hengartner, Higdon, and Ye, 2006), which was also the inspiration for the logexp transformation for θ, defined as $\log(e^{\theta} - 1)$ in Nagy et al. (2007a), where we compared it to the log transformation. However, those results were inconclusive, raising more questions than answers, and we decided not to include them here.

As we already mentioned in Chapter 3, adaptive transformations based on the data at hand might be worth exploring, too. Although the negative results in Chapter 2 seem to contradict this (i.e. adaptively optimizing λ could not beat the log transformation), one should keep in mind that we were using just one specific model with one particular correlation structure that resulted in FBI prediction bands with almost perfect frequentist properties in terms of matching coverage probabilities, leaving almost no room for improvement. But a different model with another correlation function may need a more sophisticated reparameterization scheme and it seems unwise to rule out adaptive approaches a priori based on a single negative result.

4.4 Numerical optimizations

In theory, reducing the nonnormality of likelihoods or profile likelihoods can help the required numerical optimizations to find the MLE. This is because when the shape of the likelihood functions is more Gaussian, then the shape of the log-likelihoods is more quadratic, and those are exactly the kinds of functions that can be optimized very efficiently with Newton-type algorithms.

However, in practice, this can be tricky, since it is well-known that methods relying heavily on derivatives can easily become unstable (Press, Flannery, Teukolsky, and Vetterling, 2002). This can be caused by either the features of the objective function or by numerical inaccuracies. For example, although in our case the likelihood is log-concave (Paninski, 2004), it

may still appear as possessing several local maxima along the ridge where the MLE is located (Warnes and Ripley, 1987).

Another challenge for derivative-based optimizers is that in high dimensions some partial derivatives can get dangerously close to zero, even in places that are still far away from the MLE. This problem may be possible to alleviate to some extent by dimensionality-reduction techniques, or maybe even by just screening out the input variables with little or no effect before the optimization or in parallel (Welch et al., 1992).

But when it does work, Newton's method can achieve quadratic convergence, which means doubling the number of correct digits at each iteration. That suggests that it may be worth investing some time into carefully designing a statistical experiment to identify the factors that determine efficient convergence for the log-likelihoods in question. If those factors can be controlled in a black box implementation, that can make Fast Bayesian Inference even faster, since its running time is determined by the numerical optimization required for maximum likelihood estimation.

Bibliography

Box, G. E. P. and Cox, D. R. (1964), "An Analysis of Transformations," *Journal of the Royal Statistical Society. Series B (Methodological)*, 26, 211–252.

Kass, R. E. and Slate, E. H. (1994), "Some Diagnostics of Maximum Likelihood and Posterior Nonnormality," *The Annals of Statistics*, 22, 668–695.

Linkletter, C., Bingham, D., Hengartner, N., Higdon, D., and Ye, K. Q. (2006), "Variable Selection for Gaussian Process Models in Computer Experiments," *Technometrics*, 48, 478–490.

Matérn, B. (1947), "Methods of Estimating the Accuracy of Line and Sample Plot Surveys," *Medd. fr. Statens Skagsforsknings Inst.*, 36.

Nagy, B., Loeppky, J. L., and Welch, W. J. (2007a), "Correlation parameterization in random function models to improve normal approximation of the likelihood or posterior," Tech. Rep. 229, Department of Statistics, The University of British Columbia.

— (2007b), "Fast Bayesian Inference for Gaussian Process Models," Tech. Rep. 230, Department of Statistics, The University of British Columbia.

Paninski, L. (2004), "Log-concavity results on Gaussian process methods for supervised and unsupervised learning," *Advances in Neural Information Processing*, 17.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (2002), *Numerical recipes in C++*, Cambridge: Cambridge University Press, includes bibliographical references, 3 appendixes and 2 indexes.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989), "Design and Analysis of Computer Experiments (C/R: P423-435)," *Statistical Science*, 4, 409–423.

Sprott, D. A. (1973), "Normal Likelihoods and Their Relation to Large Sample Theory of Estimation," *Biometrika*, 60, 457–465.

Stein, M. L. (1999), *Interpolation of Spatial Data: Some Theory for Kriging*, Springer-Verlag Inc.

Tukey, J. W. (1957), "On the Comparative Anatomy of Transformations," *The Annals of Mathematical Statistics*, 28, 602–632.

Warnes, J. J. and Ripley, B. D. (1987), "Problems with Likelihood Estimation of Covariance Functions of Spatial Gaussian Processes," *Biometrika*, 74, 640–642.

Welch, W. J., Buck, R. J., Sacks, J., Wynn, H. P., Mitchell, T. J., and Morris, M. D. (1992), "Screening, Predicting, and Computer Experiments," *Technometrics*, 34, 15–25.

Appendix A

Appendix to Chapter 3

In this appendix, we derive formulas for the Expected nonnormality (ENN) measure in Chapter 3, Section 3.2.5.

Let y denote the response vector having length n, mean zero, and covariance matrix $\sigma^2 R$, where σ^2 is the process variance and R is the symmetric, positive definite $n \times n$ design correlation matrix (that is a function of the parameter θ). The MLE of θ is treated as given, denoted by $\hat{\theta}$. Let G denote the inverse matrix of R and define the matrices $F = GR'$, $S = GR''$, and$T = GR'''$, where R', R'', and R''' are the first, second, and third derivatives of R, respectively (with respect to θ). The trace of a matrix is denoted by $tr(\cdot)$. For concise notation, we also define $t(\cdot) = tr(\cdot)/n$.

Taking the log of $L(\theta)$, the log-likelihood is (up to an additive constant):

$$l(\theta) = -\frac{n}{2}\log\frac{y^T R^{-1} y}{n} - \frac{1}{2}\log|R|.$$

The functions g and h are also used to simplify calculations:

$$g(\theta) = \frac{y^T R^{-1} y}{n} \quad \text{and} \quad h(\theta) = -\frac{\log|R|}{n}.$$

Suppressing θ from $l(\theta)$, $g(\theta)$, and $h(\theta)$ gives the following equations for the log-likelihood l and its first three derivatives:

$$l = \frac{n}{2}(h - \log g),$$

$$l' \;=\; \frac{n}{2}\left(h' \;-\; \frac{g'}{g}\right),$$

$$l'' \;=\; \frac{n}{2}\left(h'' \;+\; \left(\frac{g'}{g}\right)^2 \;-\; \frac{g''}{g}\right),$$

$$l''' \;=\; \frac{n}{2}\left(h''' \;-\; 2\left(\frac{g'}{g}\right)^3 \;+\; 3\,\frac{g'g''}{g^2} \;-\; \frac{g'''}{g}\right),$$

where $h' = -t(F), \quad h'' = t(F^2 - S), \quad h''' = -t(2F^3 - 3FS + T),$
and $g' = y^T G' y/n,\; g'' = y^T G'' y/n,\; g''' = y^T G''' y/n\,,$
where $G' = -FG,\; G'' = (2F^2 - S)G,\; G''' = -(6F^3 - 3FS - 3SF + T)G.$

Lemma 1. For a symmetric $n \times n$ matrix Q and $y \;\sim\; N(0,\, \sigma^2 R)$

$$E\, y^T Q y \;=\; \sigma^2\, tr(QR).$$

Proof: Using any standard text on matrix algebra, e.g. Harville (1997), $E\, y^T Q y \;=\; E\, tr(y^T Q y) \;=\; E\, tr(Q y y^T) \;=\; tr(Q\, E\, y y^T) \;=\; tr(Q\, \sigma^2 R) \;=\; \sigma^2\, tr(QR)$, where we used the fact that y has covariance matrix $\sigma^2 R$.

Lemma 2. For a symmetric $n \times n$ matrix Q, $y \;\sim\; N(0,\, \sigma^2 R)$, and $G \;=\; R^{-1}$

$$E\,\frac{y^T Q y}{y^T G y} \;=\; t(QR).$$

Proof: Let $z = C^{-1} y$, where C is the lower-triangular Cholesky-factor of the covariance matrix $\sigma^2 R$. Then $z \;\sim\; N(0,\, I_n)$ and

$$CC^T \;=\; \sigma^2 R \quad\Rightarrow\quad R \;=\; CC^T/\sigma^2 \quad\Rightarrow\quad R^{-1} \;=\; \sigma^2 \big(C^T\big)^{-1} C^{-1}.$$

Substituting $y = Cz$ and $G = \sigma^2 \big(C^T\big)^{-1} C^{-1}$ we get:

$$E\,\frac{y^T Q y}{y^T G y} \;=\; E\,\frac{z^T C^T Q C z}{z^T C^T \sigma^2 (C^T)^{-1} C^{-1} C z} \;=\; \frac{1}{\sigma^2}\, E\,\frac{z^T (C^T Q C) z}{z^T z}.$$

Conniffe and Spencer (2001) state that the expectation of a ratio of this form

is the ratio of the expectations for any quadratic form in the numerator. This is a consequence of the fact that the ratio is independent of its denominator, a result attributed to Geary (1933). Hence we can apply Lemma 1 separately to the numerator and the denominator:

$$E\,\frac{y^TQy}{y^TGy} \;=\; \frac{E\,y^TQy}{E\,y^TGy} \;=\; \frac{\sigma^2\,tr(QR)}{\sigma^2\,tr(GR)} \;=\; \frac{tr(QR)}{tr(I_n)} \;=\; \frac{tr(QR)}{n} \;=\; t(QR).$$

Lemma 3. $El''(\hat{\theta})$ and $El'''(\hat{\theta})$ for the ENN are:

$$El'' \;=\; \frac{n}{2}\,\left(t^2(F) \;-\; t(F^2)\right),$$

$$El''' \;=\; \frac{n}{2}\,\left(2t^3(F) \;-\; 6t(F)t(F^2) \;+\; 3t(F)t(S) \;-\; 3t(FS) \;+\; 4t(F^3)\right).$$

Proof: When $\theta = \hat{\theta}$ (the MLE of θ), then $l' = 0$ and that implies that $g'/g = h'$. Replacing g'/g with h' in the second and third derivative formulas for l leads to the following expressions:

$$l'' \;=\; \frac{n}{2}\,\left(h'' \;+\; \left(h'\right)^2 \;-\; \frac{g''}{g}\right)$$

$$l''' \;=\; \frac{n}{2}\,\left(h''' \;-\; 2\,\left(h'\right)^3 \;+\; 3\,h'\,\frac{g''}{g} \;-\; \frac{g'''}{g}\right).$$

Taking expectations:

$$El'' \;=\; \frac{n}{2}\,\left(h'' \;+\; \left(h'\right)^2 \;-\; E\,\frac{g''}{g}\right)$$

$$El''' \;=\; \frac{n}{2}\,\left(h''' \;-\; 2\,\left(h'\right)^3 \;+\; 3\,h'\,E\,\frac{g''}{g} \;-\; E\,\frac{g'''}{g}\right).$$

Now Lemma 2 can be applied to the expectations of the ratios:

$$E\,\frac{g''}{g} \;=\; E\,\frac{y^TG''y}{y^TGy} \;=\; t(G''R) \qquad \text{and} \qquad E\,\frac{g'''}{g} \;=\; E\,\frac{y^TG'''y}{y^TGy} \;=\; t(G'''R).$$

Substituting the formulas for G'', G''', and h', h'', h''' completes the proof.

Bibliography

Conniffe, D. and Spencer, J. E. (2001), "When Moments of Ratios Are Ratios of Moments," *Journal of the Royal Statistical Society, Series D: The Statistician*, 50, 161–168.

Geary, R. C. (1933), "A General Expression for the Moments of Certain Symmetrical Functions of Normal Samples," *Biometrika*, 25, 184–186.

Harville, D. A. (1997), *Matrix Algebra from a Statistician's Perspective*, Springer-Verlag Inc.

Appendix B

Appendix to Chapter 4

This appendix contains the simulation results in Nagy, Loeppky, and Welch (2007) for three different inference methods. In addition to the plug-in and FBI that are the same as in Chapter 2, it also includes an extra Bayesian method using Markov chain Monte Carlo (MCMC) to sample from another posterior distribution that is different from the one used by the FBI. That means that the two Bayesian methods are not expected to give the same results. (However, as we will see in Section B.3, there is a strong connection: the FBI's posterior is the normal approximation of the posterior sampled by the MCMC).

After warning the reader in the next section why the results of this appendix should be taken with a grain of salt, in Section B.2 we outline the simulation procedure in Nagy et al. (2007) that is similar to the one presented in Chapter 2, but it explores a much wider range of experimental setups, and also has another method using MCMC, which is described in detail in Section B.3. Other differences include an alternative way to calculate the Coverage Probability (CP) of a prediction interval (derived in Section B.4) using the prior information that the data is a realization of a Gaussian process. Finally, the results presented in Section B.5 are based on the normality assumption for the prediction intervals (which can be improved for the smallest sample sizes by using the t-distribution instead, as argued in Chapter 2). The true coverage probabilities obtained for the three different methods for the nominal 90%, 95%, and 99% levels are summarized in a table, followed by 10 figures for the the simulation results in $d = 1, \ldots, 10$, plotting the actual coverages of the $100(1-\alpha)\%$ pointwise prediction bands

against the nominal levels for $\alpha = 0.01,\ 0.02,\ \ldots,\ 0.99$ in the same fashion as Figures 2.3 and 2.4 before in Chapter 2.

B.1 Warning

Unlike the carefully designed 20 simulation studies in Chapters 2 and 3, the 90 simulations in this appendix have several limitations. This is because in addition to sensible choices (that one may expect to encounter in practice), we also wanted to explore more extreme experimental setups, such as overly challenging response surfaces or samples that are much too small relative to the number of input variables. Of course, the drawback of casting such a wide net is that we catch more interesting behaviors than we ask for. For example, unlike in Chapters 2 and 3, many times we could not complete computations for all $1,000$ replicates because of numerical difficulties. Note that here we are no longer talking about ill-conditioning of correlation matrices whose inverses had unrealistically large elements (as in Chapter 3), but outright failures when attempting to take the inverse ended with an error message. Although these failures were excluded from the final analysis, other degenerate cases were not, e.g. when we did get an inverse, we did not check whether it was "realistic" or not. Hence, there is no guarantee that numerical issues did not have a significant influence in some cases, and these limitations should be kept in mind when drawing conclusions from the results in this appendix. But in spite of all the difficulties, it is still interesting to observe the performance of the three methods on the frontiers of their applicability.

B.2 Simulations

The simulation plan can be viewed as a set of 10 statistically designed experiments for $d = 1,\ \ldots,\ 10$. For each experiment, the design was a 3×3 full-factorial with $1,000$ replicates. The two factors were the range parame-

ter θ and the sample size n, both at three levels (equally spaced on the log scale): $\theta = 0.2,\ 2,\ 20$ and $n = 10\,d/4,\ 10\,d/2,\ 10\,d$ (where $10\,d/4$ was rounded up to the nearest integer).

To obtain $1,000$ replicates for a given combination of θ and n, the following four steps were repeated (attempted) $1,000$ times:

1. Select an n point design by Latin hypercube sampling in the d-dimensional unit hypercube $[0,\ 1]^d$ (McKay, Beckman, and Conover, 1979).

2. Generate a realization $\boldsymbol{y}$ of the Gaussian process over the n design points by setting the range parameter to θ in all dimensions and the process variance to one.

3. Sample 10 new points uniformly in the unit hypercube $[0,\ 1]^d$ for prediction.

4. Compute the predictors for the three methods with their mean squared errors for the 10 new points from the data $\boldsymbol{y}$.

Note that step 2 or 4 could fail because of numerical issues, leading to an unsuccessful realization (missing value) for that particular replicate (not included in subsequent analysis). The only case when this had a catastrophic impact on results was the $\theta = 0.2,\ n = 10$ case for $d = 1$, since setting θ to 0.2 pushes the limits of the standard double precision representation in the one input case: numerical difficulties arise because the high correlations in the $n \times n$ correlation matrix (all close to one) make it ill-conditioned (nearly singular). Hence, in Section B.5, the numbers and plots are missing in the $\theta = 0.2,\ n = 10$ case from the first ($d = 1$) row of the table and the first ($d = 1$) figure, respectively.

B.3 MCMC

To compare the FBI with another Bayesian method, we used uniform priors on the log scale for the range parameters and sampled the resulting poste-

rior by MCMC, using the Metropolis random walk algorithm (Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller, 1953) that has been used successfully in many high-dimensional problems. Of course, this comparison is not entirely fair because the FBI is not taking samples directly from this posterior but its normal approximation. But to make it as comparable to the FBI as possible, everything was done on the log scale using the same $\boldsymbol{\gamma}$-parameterization as in Chapter 2. Also, the first two moments of the $N(\hat{\boldsymbol{\gamma}},\, -H^{-1}_{\hat{\boldsymbol{\gamma}}})$ normal approximation were utilized to help the implementation in step 1 and step 3 of the algorithm, respectively:

1. Initialize $\boldsymbol{\gamma}^{(1)}$ at $\hat{\boldsymbol{\gamma}}$.

2. To select a direction for a random walk step, sample an integer j uniformly from $1, \ldots, d$.

3. Given the current $\boldsymbol{\gamma}^{(i)}$, set $\boldsymbol{\gamma}^*$ to $\boldsymbol{\gamma}^{(i)}$ and then add to the jth coordinate of $\boldsymbol{\gamma}^*$ a normal random deviate with mean zero and standard deviation equal to three times the standard error in the jth dimension, estimated from the Hessian: $\sqrt{-H^{-1}_{\hat{\boldsymbol{\gamma}}}(j,j)}$.

4. Compute the acceptance ratio for $\boldsymbol{\gamma}^*$, given $\boldsymbol{\gamma}^{(i)}$:

$$p \;=\; \min\left\{\, 1,\; \frac{L(\exp\boldsymbol{\gamma}^*)}{L(\exp\boldsymbol{\gamma}^{(i)})} \,\right\}.$$

5. Set $\boldsymbol{\gamma}^{(i+1)}$ to $\boldsymbol{\gamma}^*$ with probability p and to $\boldsymbol{\gamma}^{(i)}$ with probability $1-p$.

6. Repeat steps 2–5 until i reaches the desired sample size.

When this algorithm works well, it constructs a Markov chain whose stationary distribution is the posterior distribution. The resulting sample then can be used for prediction exactly the same way as the sample for the FBI (step 4 in Chapter 2, Section 2.7). In other words, once the sampling is done, the treatment of the samples are identical.

But that does not mean that the samples are equivalent or similar. The MCMC algorithm constructs a large, dependent sample from a posterior that is only known up to a scale (proportional to the likelihood because of the uniform priors used for the γ parameters). On the other hand, the FBI can get away with a much smaller independent, identically distributed (iid) sample that is not from the same posterior, but from its normal approximation $N(\hat{\boldsymbol{\gamma}},\ -H_{\hat{\boldsymbol{\gamma}}}^{-1})$, as in Chapter 2. The Monte Carlo sample size for the FBI was only 400, minus those sample points that ran into numerical difficulties caused by the ill-conditioning of the correlation matrix. This happened mostly in lower-dimensional cases, especially in $d = 1$. The MCMC sample size was $N = 100,000$ (after $10,000$ burn-in). Unlike the FBI sample, the MCMC sample did not suffer from numerical problems because problematic points would never be accepted by the algorithm, since the likelihood/posterior was set to zero whenever the Cholesky-decomposition of the correlation matrix failed.

An MCMC run was considered successful if the acceptance rate was at least 15% and the Mean Effective Sample Size (MESS) was at least 50. Both measures were calculated after the burn-in phase. The following formula was used for the MESS:

$$\text{MESS} \;=\; \frac{1}{d}\sum_{i=1}^{d} N\left[1 + 2\sum_{k=1}^{1000}\left(1 - \frac{k}{N}\right)\hat{\rho}_k(i)\right]^{-1},$$

where $\hat{\rho}_k(i)$ is the kth sample autocorrelation in the ith dimension (Carter and Kohn, 1994).

These measures were intended to provide some minimal automatic quality control, since visual examination of various diagnostic plots for all runs were clearly not possible. Of course, there is no guarantee that an MCMC chain that met both of these criteria (and as a result was classified as successful) has actually converged to the stationary distribution or was not deficient in some other way. The original technical report Nagy et al. (2007)

has more details about potential problems and the challenges of this particular MCMC implementation.

B.4 Coverage Probability

Coverage probabilities for prediction bands were calculated by averaging the individual CPs over all new points and all successful realizations. A realization was considered successful if all operations for all three methods completed without error. It is straightforward to compute an individual CP. Suppose that we want to predict the output Y_0 at a new, untried input $\boldsymbol{x}_0$. Since the true model is known during the simulation, we know that conditionally on the realized data, Y_0 is normally distributed with mean μ_0 and variance σ_0^2, where μ_0 and σ_0^2 are given by equations (2.3) and (2.4) in Chapter 2, respectively.

Now suppose that after estimation, the predictor for Y_0 was μ_1 with mean squared error σ_1^2. This amounts to mis-specifying the distribution of the random variable Y_0 as $N(\mu_1, \sigma_1^2)$ instead of the true $N(\mu_0, \sigma_0^2)$.

Then the CP of a normality-based $100(1-\alpha)\%$ prediction interval about μ_1 is

$$P_0\left(\mu_1 - \sigma_1\, z_{\alpha/2} \;<\; Y_0 \;<\; \mu_1 + \sigma_1\, z_{\alpha/2}\right) \;=$$

$$=\; P_0\left(\frac{\mu_1 - \sigma_1\, z_{\alpha/2} - \mu_0}{\sigma_0} \;<\; \frac{Y_0 - \mu_0}{\sigma_0} \;<\; \frac{\mu_1 + \sigma_1\, z_{\alpha/2} - \mu_0}{\sigma_0}\right) \;=$$

$$=\; \Phi\left(\frac{\mu_1 + \sigma_1\, z_{\alpha/2} - \mu_0}{\sigma_0}\right) \;-\; \Phi\left(\frac{\mu_1 - \sigma_1\, z_{\alpha/2} - \mu_0}{\sigma_0}\right),$$

where P_0 denotes the true probability distribution, Φ is the cumulative distribution function of the standard normal $N(0,1)$, and $z_{\alpha/2}$ satisfies $\Phi(-z_{\alpha/2}) = \alpha/2$.

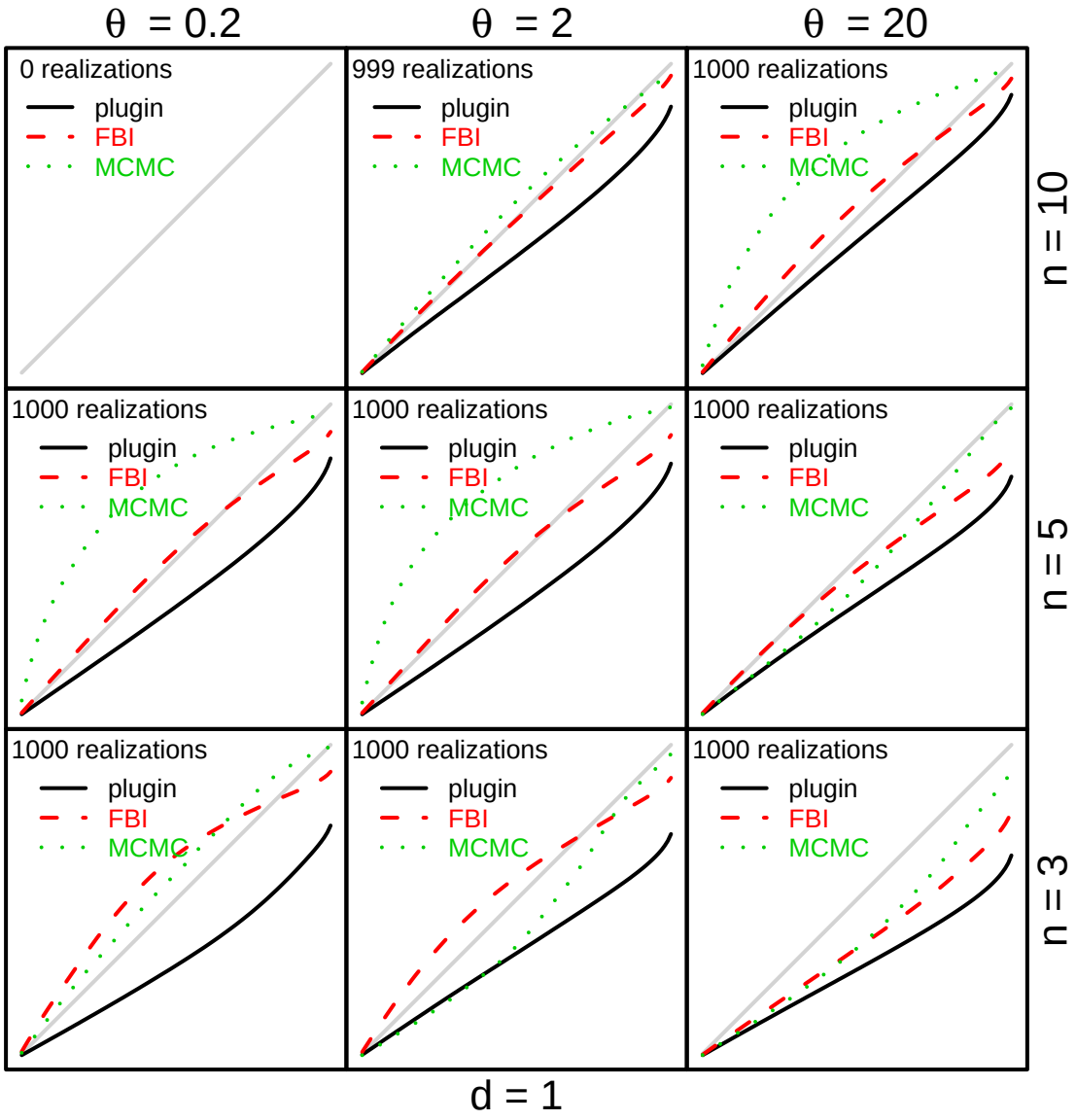
B.5 Results

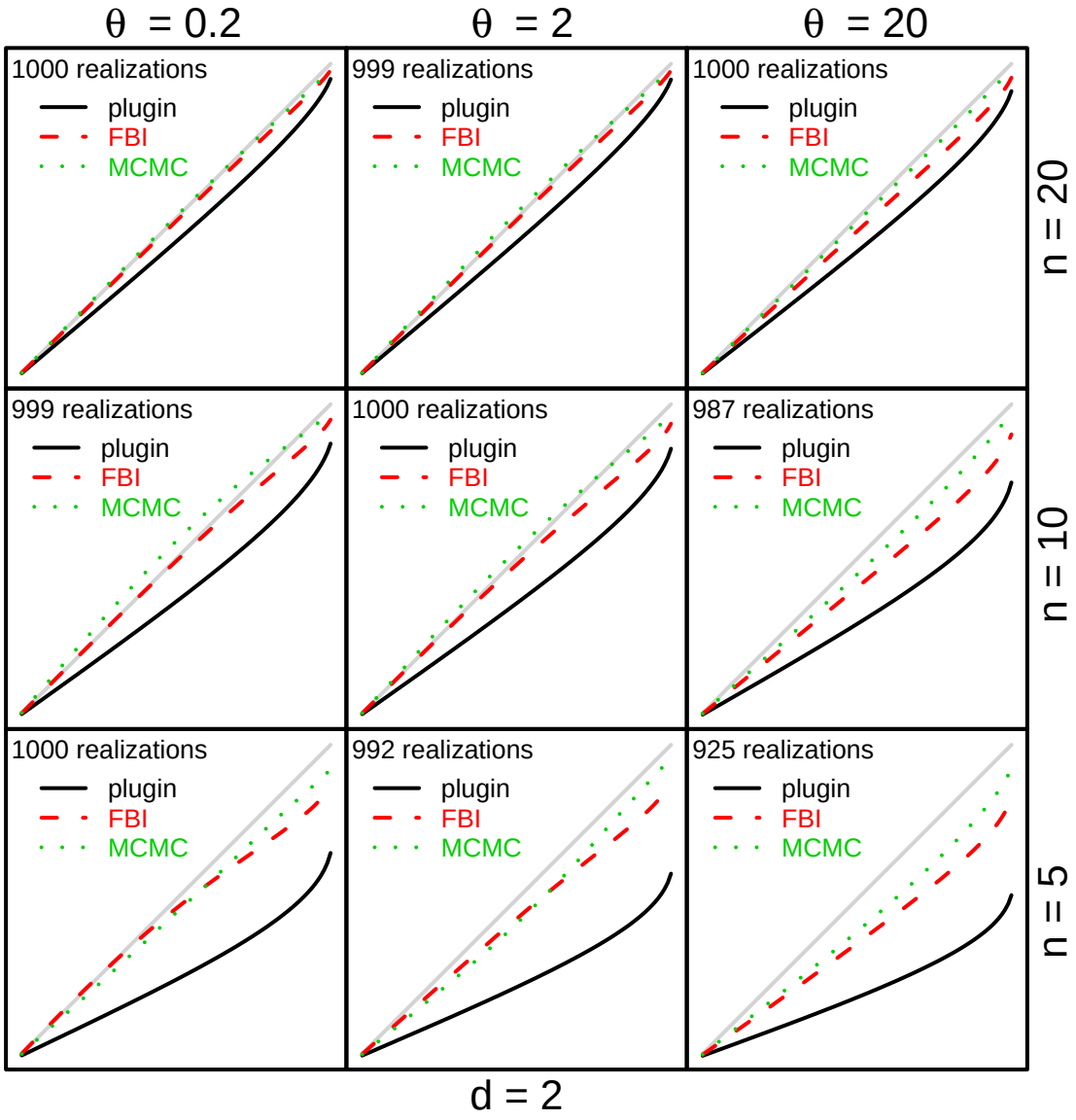
The following table is a summary of the CPs of the pointwise prediction bands of the three competing methods for the nominal 90%, 95%, and 99% confidence levels. The two-digit numbers in the table are truncated percentages without the percent sign and without the fractional parts (rounded down). The 3×3 arrangement inside each cell follows the layout of the plots in the following figures by the three levels of θ horizontally and the three levels of n vertically.

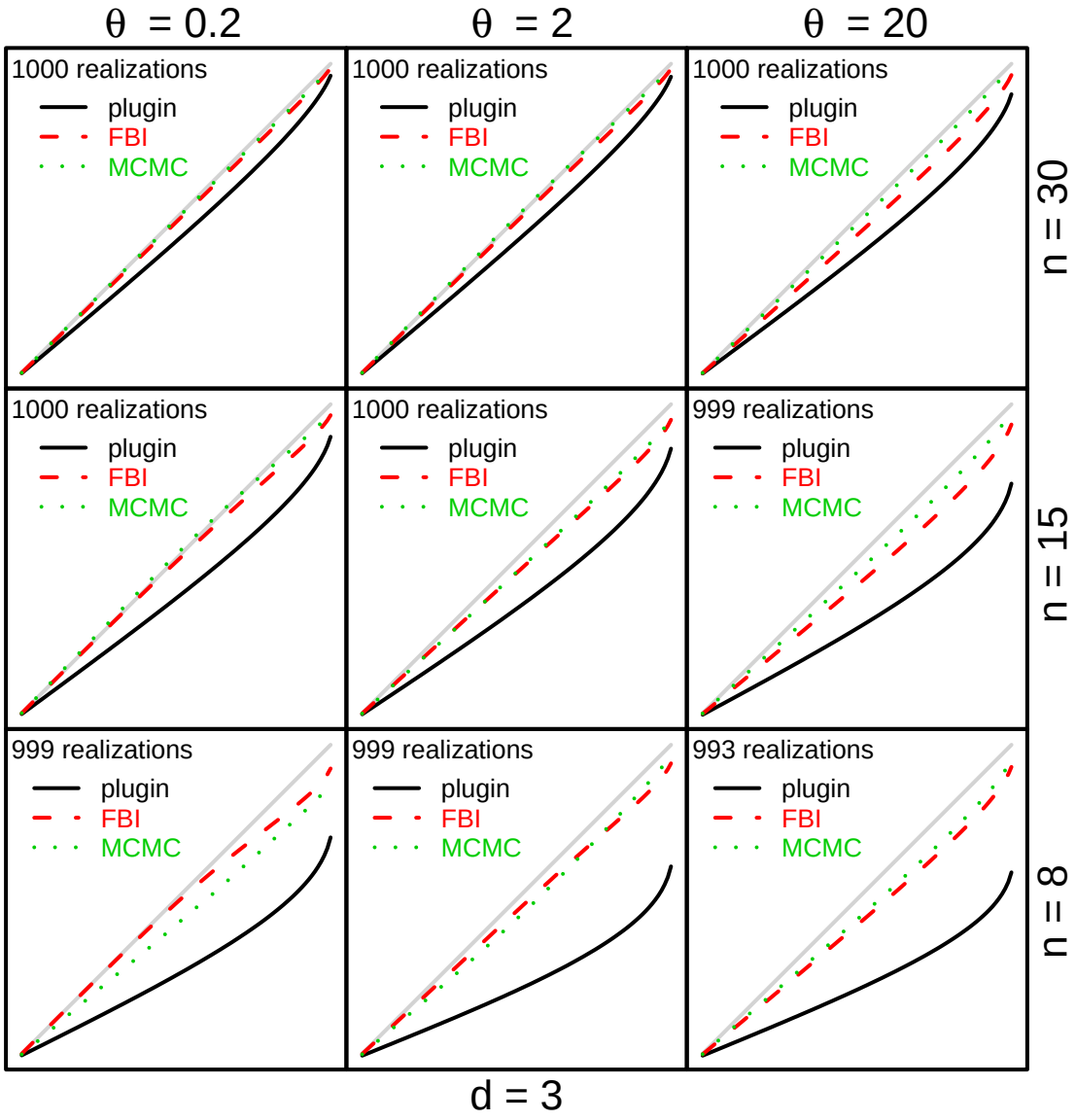
After the table, the following 10 figures compare the validity of the three methods for $d = 1, \ldots, 10$, for all combinations of the three levels of θ and the three levels of n. In addition to the gray diagonal in the middle, three curves were plotted for the three methods relating the true coverage probabilities on the vertical axis (from 1% to 99%) to the nominal coverage on the horizontal axis (from 1% to 99%). This is the same as before in Figures 2.3 and 2.4 in Chapter 2, just this time the axis labels are not shown, to be able to present 9 plots in the same figure in a 3×3 arrangement.

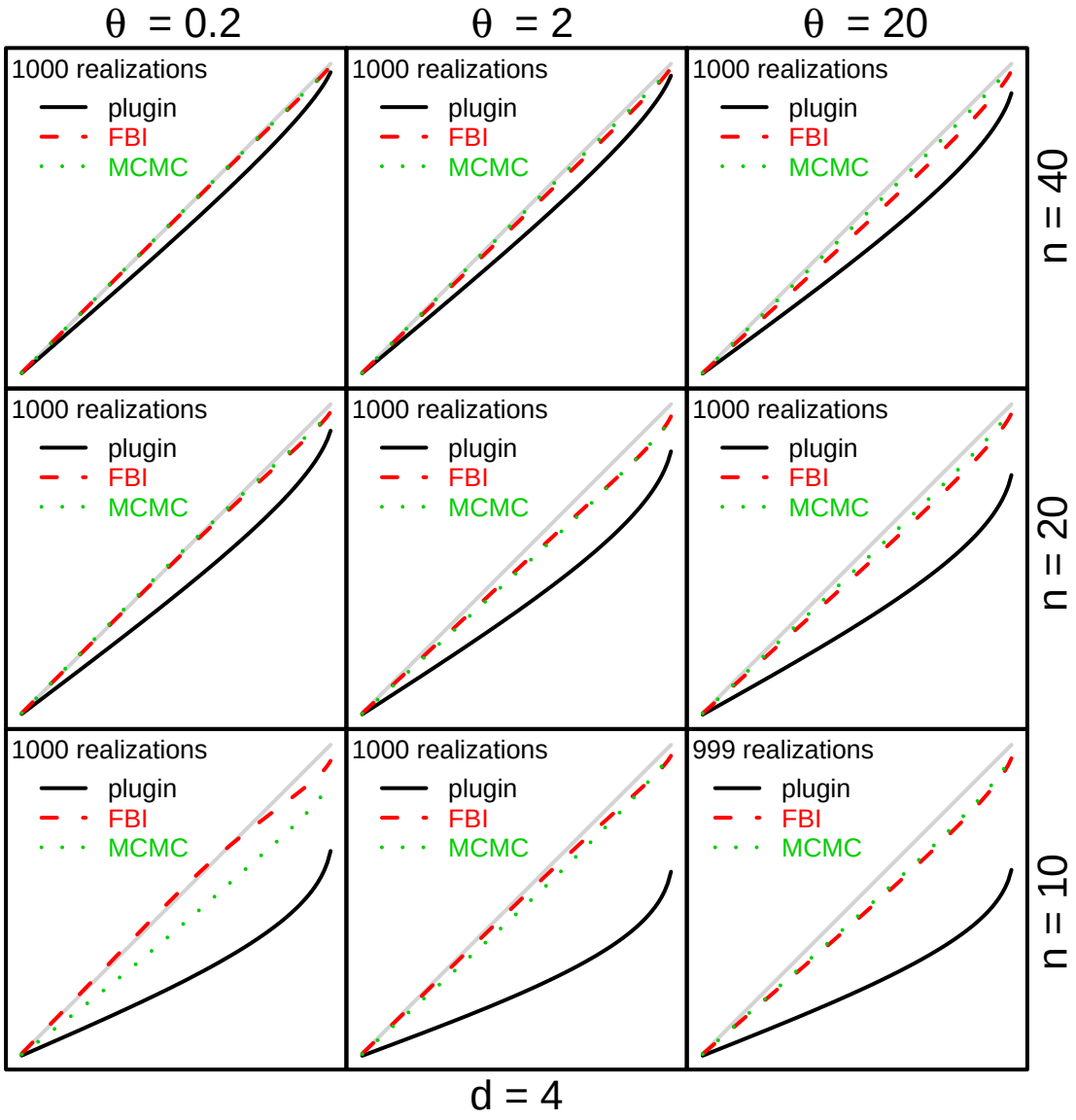
Plots are based on the realizations that were classified as successful, out of $1,000$ attempts in total. Counts for the number of realizations included in the final calculations are shown in the top-left corner of each plot. Calculations of the CPs were always restricted to the successful subset of the $1,000$ realizations and all failures were excluded.

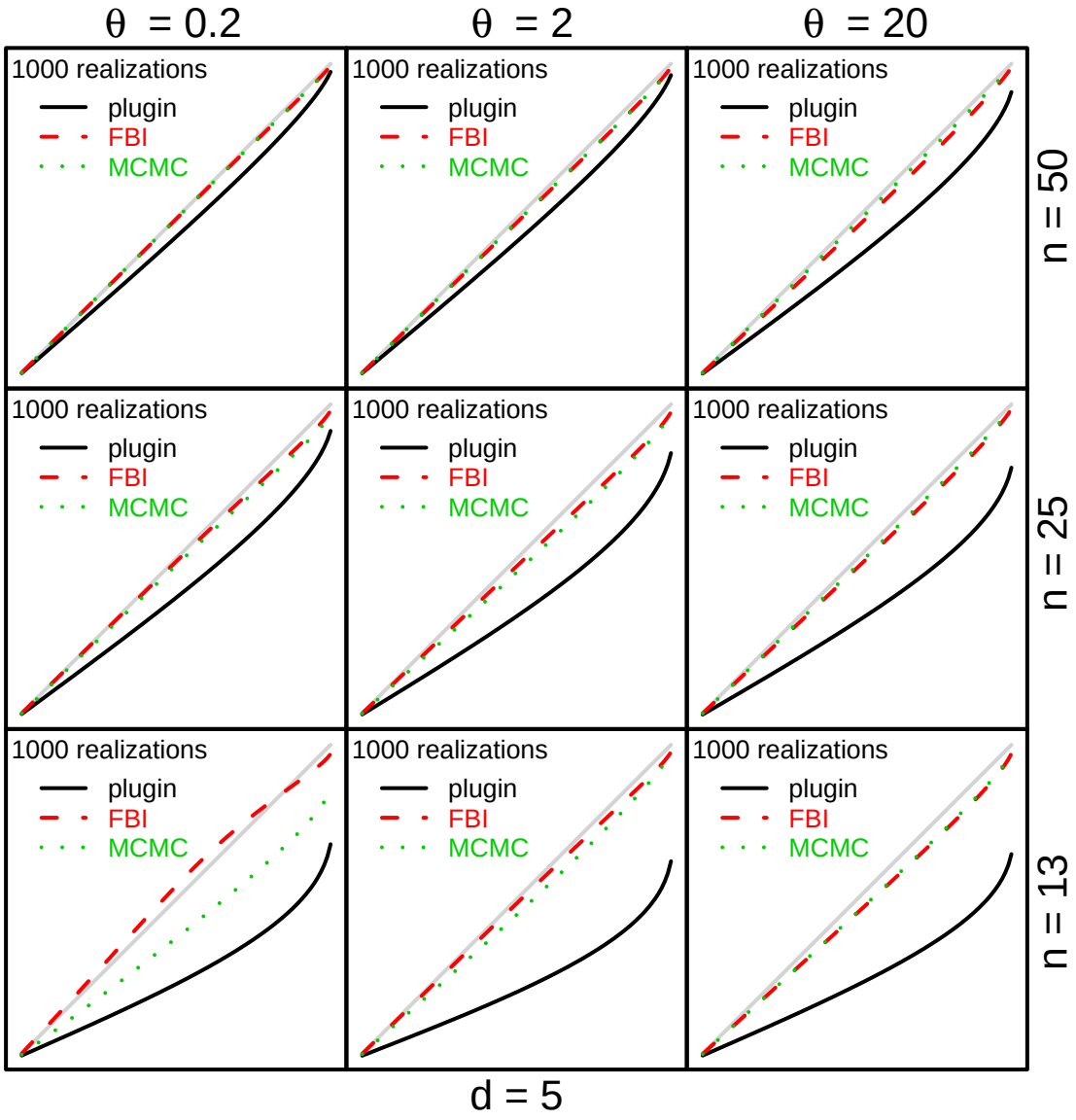
d	90%			95%			99%		
	plugin	FBI	MCMC	plugin	FBI	MCMC	plugin	FBI	MCMC
1	72 76 67 66 64 61 59 52	85 85 82 81 75 85 81 65	89 94 94 95 88 94 87 79	78 82 73 72 69 66 64 57	90 89 86 84 79 87 84 71	93 96 95 97 94 96 92 85	85 89 81 80 76 73 70 63	95 94 90 89 84 90 88 78	96 98 97 98 97 98 96 90
2	80 80 75 71 70 59 50 45 39	87 87 83 84 82 76 76 76 68	88 89 88 89 87 85 82 85 78	87 86 82 78 76 65 56 50 44	92 92 88 89 87 82 80 81 74	93 93 93 93 92 91 87 90 84	94 94 90 86 84 74 64 58 51	96 96 94 94 92 89 86 87 82	97 97 97 96 96 96 91 95 91
3	81 81 74 73 68 57 53 45 44	87 87 83 85 82 78 81 82 78	88 88 88 87 84 85 77 84 81	88 87 81 80 75 64 60 51 50	92 92 89 90 87 85 85 87 84	93 93 93 92 89 91 82 90 88	95 94 89 88 84 73 69 60 58	97 97 95 95 94 92 91 93 91	97 97 98 96 94 96 87 95 94
4	83 81 74 75 67 60 49 43 44	88 87 84 86 83 84 84 85 81	89 88 88 87 83 86 75 85 83	89 88 81 82 74 66 55 49 50	93 92 90 91 89 89 89 90 87	94 93 93 92 89 92 81 90 89	96 95 89 90 84 76 65 58 59	98 97 96 96 95 96 93 95 94	98 97 98 96 94 97 87 96 95
5	83 81 74 74 65 62 50 45 48	88 87 87 86 85 86 88 86 84	88 87 88 85 84 87 72 84 84	89 88 81 81 73 69 56 51 54	93 92 92 91 91 92 92 91 90	93 92 93 90 90 92 78 90 90	96 95 90 90 83 78 67 62 64	98 97 97 96 96 97 96 96 96	98 97 98 95 95 98 84 96 96
6	83 80 74 74 65 63 49 46 49	88 87 88 87 87 87 90 88 85	88 86 88 82 85 87 72 85 85	89 87 81 81 72 70 56 53 55	93 92 93 92 92 92 94 93 91	93 91 94 88 91 93 78 91 91	96 95 90 90 83 80 67 63 65	98 97 98 97 97 97 97 97 97	98 96 98 94 96 98 85 96 97
7	84 79 76 72 64 65 50 47 51	89 88 88 86 88 87 92 88 86	88 85 88 81 86 88 73 86 86	90 86 83 80 72 73 57 54 57	94 93 94 92 93 93 95 93 92	93 91 94 87 91 93 79 91 91	96 94 91 89 83 82 68 64 67	98 98 98 97 98 98 98 98 97	98 96 98 93 96 98 86 97 97
8	84 79 76 71 64 66 49 48 51	89 88 89 87 89 88 93 88 86	89 86 89 80 86 88 73 86 86	90 86 83 79 72 74 56 55 58	94 93 94 92 94 93 96 93 92	94 91 94 86 92 93 79 92 92	97 94 91 89 83 83 67 66 68	98 98 98 97 98 98 98 98 97	98 97 98 93 97 98 87 97 97
9	84 78 77 71 65 67 50 50 52	89 89 89 88 89 88 94 88 87	88 86 89 79 87 88 74 86 86	90 85 84 79 73 74 57 57 59	94 94 94 93 94 93 97 93 92	94 92 94 85 93 93 80 92 92	97 93 92 88 84 84 69 68 69	98 98 98 97 98 98 99 98 97	98 97 98 93 97 98 88 97 97
10	84 77 78 70 66 68 49 50 54	89 89 89 90 89 88 94 88 87	88 87 89 79 88 88 75 87 87	90 85 84 78 74 75 57 57 61	94 94 94 94 94 93 97 93 93	94 92 94 85 93 93 81 92 92	97 93 93 88 84 85 68 69 72	98 98 98 98 98 98 99 98 98	98 97 98 93 98 98 89 97 97

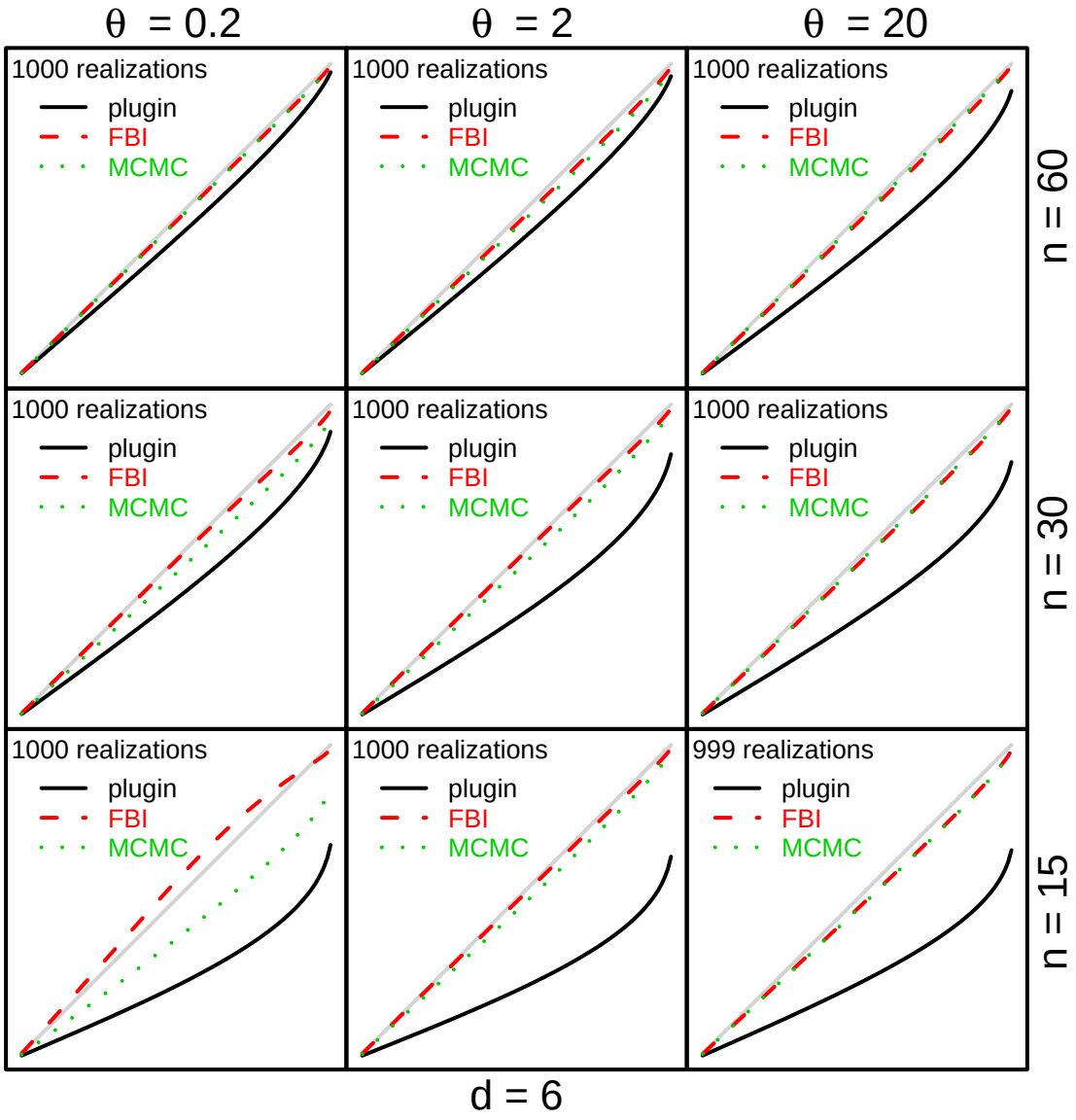

θ = 0.2
θ = 2
θ = 20
0 realizations
999 realizations
1000 realizations
plugin
FBI
MCMC
n = 10
n = 5
n = 3
d = 1

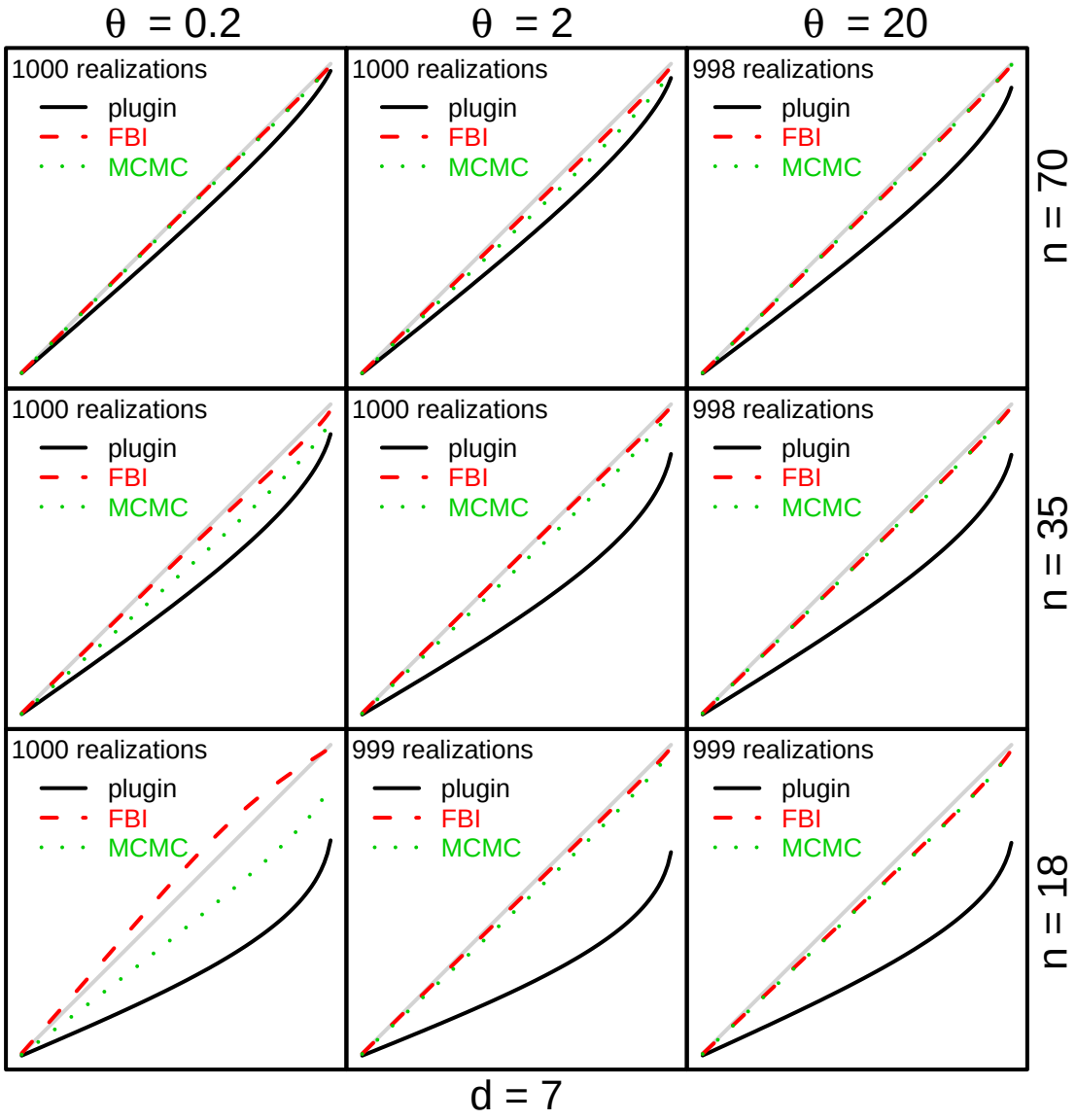

θ = 0.2
θ = 2
θ = 20
1000 realizations
999 realizations
1000 realizations
999 realizations
1000 realizations
987 realizations
1000 realizations
992 realizations
925 realizations
plugin
FBI
MCMC
n = 20
n = 10
n = 5
d = 2

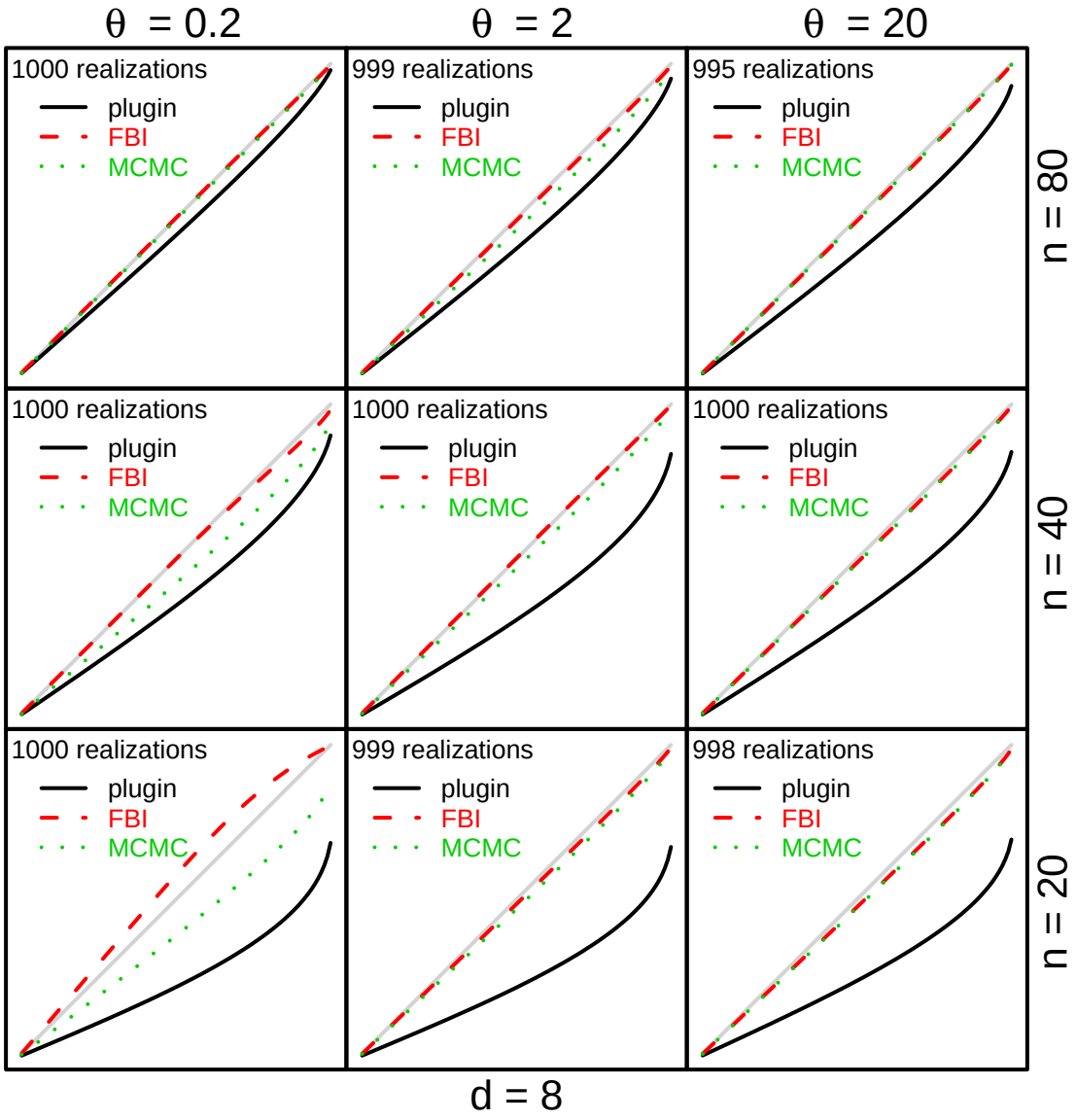

θ = 0.2
θ = 2
θ = 20
n = 30
n = 15
n = 8
d = 3
1000 realizations
1000 realizations
1000 realizations
1000 realizations
1000 realizations
999 realizations
999 realizations
999 realizations
993 realizations
plugin
FBI
MCMC

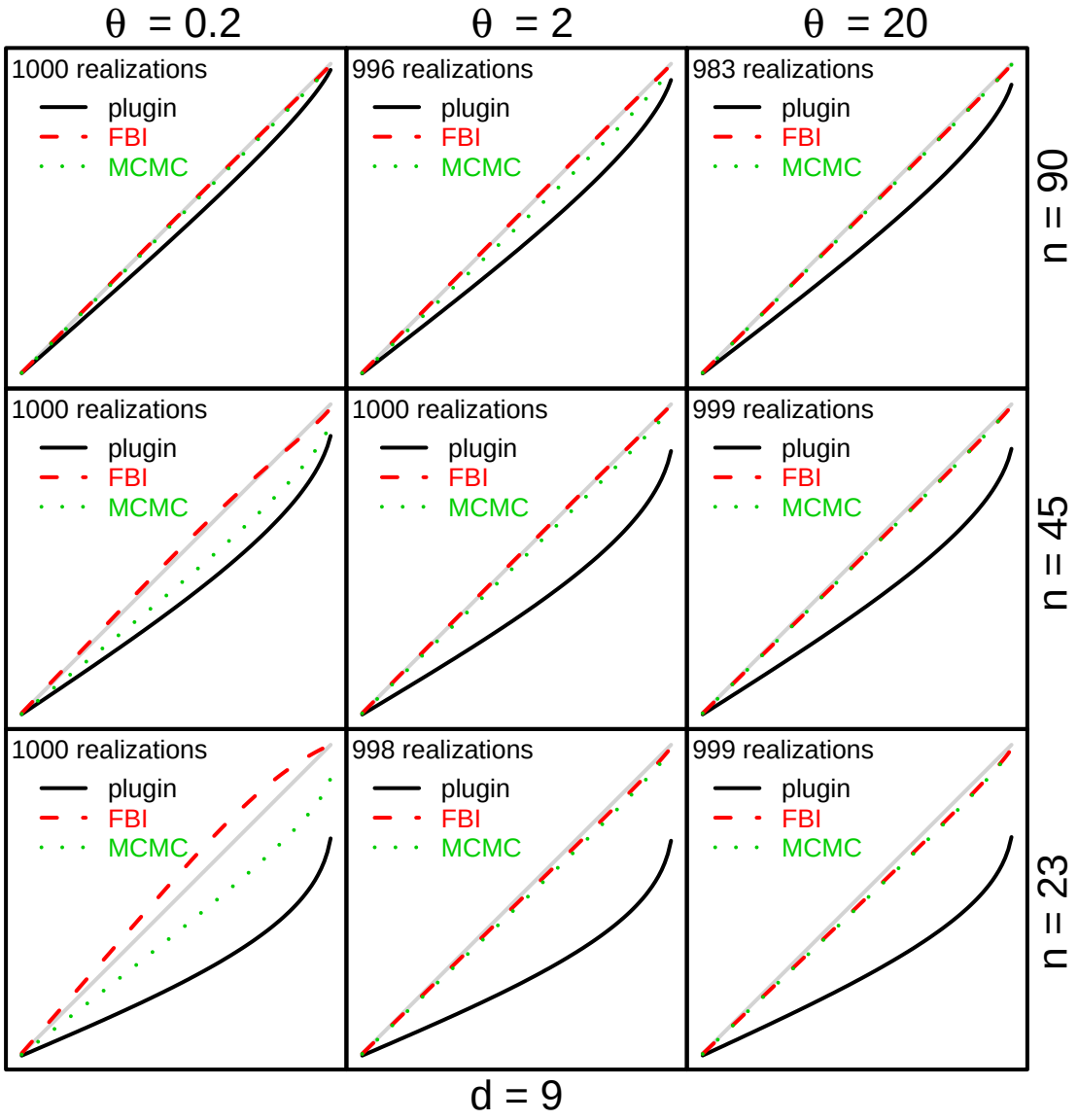

θ = 0.2
θ = 2
θ = 20
n = 40
n = 20
n = 10
d = 4
1000 realizations
999 realizations
plugin
FBI
MCMC

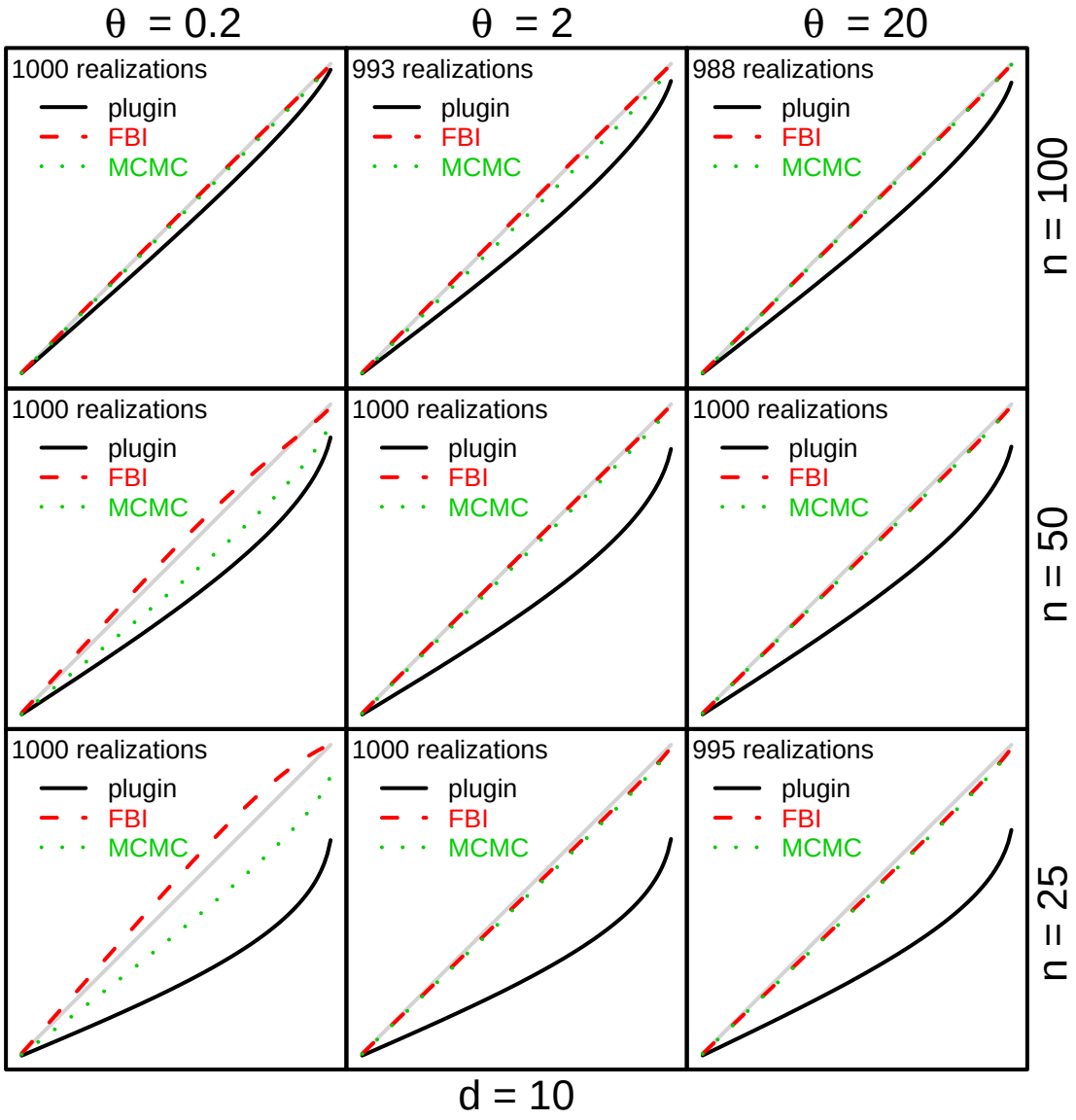

θ = 0.2
θ = 2
θ = 20
n = 50
n = 25
n = 13
d = 5
1000 realizations
plugin
FBI
MCMC


θ = 0.2
θ = 2
θ = 20
n = 60
n = 30
n = 15
d = 6
1000 realizations
999 realizations
plugin
FBI
MCMC


θ = 0.2
θ = 2
θ = 20
1000 realizations
1000 realizations
998 realizations
1000 realizations
1000 realizations
998 realizations
1000 realizations
999 realizations
999 realizations
plugin
FBI
MCMC
n = 70
n = 35
n = 18
d = 7


θ = 0.2
θ = 2
θ = 20
n = 80
n = 40
n = 20
d = 8
1000 realizations
999 realizations
995 realizations
1000 realizations
1000 realizations
1000 realizations
1000 realizations
999 realizations
998 realizations
plugin
FBI
MCMC


θ = 0.2
θ = 2
θ = 20
1000 realizations
996 realizations
983 realizations
1000 realizations
1000 realizations
999 realizations
1000 realizations
998 realizations
999 realizations
plugin
FBI
MCMC
n = 90
n = 45
n = 23
d = 9


θ = 0.2
θ = 2
θ = 20
n = 100
n = 50
n = 25
d = 10
1000 realizations
993 realizations
988 realizations
1000 realizations
1000 realizations
1000 realizations
1000 realizations
1000 realizations
995 realizations
plugin
FBI
MCMC

Bibliography

Carter, C. K. and Kohn, R. (1994), "On Gibbs Sampling for State Space Models," *Biometrika*, 81, 541–553.

McKay, M. D., Beckman, R. J., and Conover, W. J. (1979), "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, 21, 239–245.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953), "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, 21, 1087–1091.

Nagy, B., Loeppky, J. L., and Welch, W. J. (2007), "Fast Bayesian Inference for Gaussian Process Models," Tech. Rep. 230, Department of Statistics, The University of British Columbia.