# Minimally Corrective, Approximately Recovering Priors to Correct Expert Judgement in Bayesian Parameter Estimation

Thomas J. May

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Mathematics

Jeffrey T. Borggaard (Chair)
Lizette Zietsman (Co-Chair)
John Rossi

May 1, 2015
Blacksburg, Virginia

Keywords: Bayesian Parameter Estimation, Minimally Corrective Priors, Distributed Parameters, Elliptic Equation, Karhunen-Loève Theorem.

# Minimally Corrective, Approximately Recovering Priors to Correct Expert Judgement in Bayesian Parameter Estimation

Thomas J. May

(ABSTRACT)

Bayesian parameter estimation is a popular method to address inverse problems. However, since prior distributions are chosen based on expert judgement, the method can inherently introduce bias into the understanding of the parameters. This can be especially relevant in the case of distributed parameters where it is difficult to check for error. To minimize this bias, we develop the idea of a minimally corrective, approximately recovering prior (MCAR prior) that generates a guide for the prior and corrects the expert supplied prior according to that guide. We demonstrate this approach for the 1D elliptic equation or the elliptic partial differential equation and observe how this method works in cases with significant and without any expert bias. In the case of significant expert bias, the method substantially reduces the bias and, in the case with no expert bias, the method only introduces minor errors. The cost of introducing these small errors for good judgement is worth the benefit of correcting major errors in bad judgement. This is particularly true when the prior is only determined using a heuristic or an assumed distribution.

# Contents

**List of Figures** **v**

**List of Tables** **vii**

**1 Introduction** **1**

**2 The Inverse Problem** **3**

2.1 The General Inverse Problem . . . . . 3

2.2 Bayesian Parameter Estimation . . . . . 4

2.2.1 Formulation . . . . . 4

2.2.2 An Example of Bayesian Parameter Estimation . . . . . 5

2.2.3 Criticism of Bayesian Parameter Estimation . . . . . 8

2.3 Sampling from the Posterior with Monte Carlo Methods . . . . . 10

**3 The Elliptic Equation** **12**

3.1 Identifiability of the Parameters . . . . . 12

3.2 The Finite Difference Approximation of $L_q(u)$ . . . . . 13

**4 Representation of Distributed Parameters** **15**

4.1 Karhunen-Loève Representation . . . . . 15

4.2 Polynomial Regression . . . . . 17

**5 MCAR Priors** **20**

5.1 Recovering and Approximately Recovering Priors . . . . . 20

5.1.1 Recovering Priors . . . . . 21
5.1.2 Approximately Recovering Priors . . . . . 22
5.2 Minimally Corrected Approximately Recovering Priors . . . . . 24
5.2.1 Quantifying Information Loss . . . . . 24
5.2.2 Correcting the Prior . . . . . 24
5.2.3 Determining the Corrected Prior . . . . . 26
5.3 Summary of Determining the MCAR Prior . . . . . 26

**6 Numerical Results** **28**
6.1 Problem Formulation . . . . . 28
6.1.1 Computing Environment . . . . . 30
6.2 Determining the Recovery Requirement . . . . . 30
6.3 An Inaccurate Prior . . . . . 32
6.4 An Accurate Prior . . . . . 36

**7 Conclusions** **41**

**8 Bibliography** **42**

# List of Figures


2.1 The tails biased prior density for Example 2.1. . . . . 6

2.2 The likelihood function with $z = (7, 13)$ for Example 2.1 . . . . . 7

2.3 The posterior density function for Example 2.1 . . . . . 7

2.4 The prior density function for Example 2.2 . . . . . 8

2.5 The posterior density function for Example 2.2 . . . . . 9

6.1 The experimental observations in blue vs. the true data in red. . . . . 29

6.2 The sample mean of the experimental observations in blue vs. the true data in red. . . . . 30

6.3 The training error (4.17) (blue) and the cross-validation error (4.18) (red). . 31

6.4 The approximation $\hat{u}(x)$ (blue) vs. the true $u(x)$ (red). . . . . 32

6.5 The approximation $\hat{u}'(x)$ (blue) vs. the true $u'(x)$ (red). . . . . 33

6.6 The recovery requirement $E[\hat{q}(x)]$ (blue) vs. the true $q(x)$ (red). . . . . 33

6.7 The $E[\hat{f}(x)]$ (blue) vs. the true $f(x)$ (red). . . . . 34

6.8 The prior expected value of $q$ (blue) and the true value of $q$ (red). . . . . 35

6.9 The posterior expected value of $q$ (blue) and the true value of $q$ (red). . . . . 35

6.10 The posterior expected value of $q$ (blue) and the recovery requirement (green). 36

6.11 The corrected posterior expected value of $q$ (blue) and the recovery requirement (green). . . . . 37

6.12 The corrected posterior expected value of $q$ (blue) and the true parameter (red). 37

6.13 The posterior expected value of $q$ (blue) and the true value of $q$ (red). . . . . 38

6.14 The posterior expected value of $q$ (blue) and the recovery requirement of $q$ (green). . . . . 38
6.15 The corrected posterior expected value of $q$ (blue) and the recovery requirement (green). . . . . 39
6.16 The corrected posterior expected value of $q$ (blue) and the true parameter (red). 40

# List of Tables

6.1 The original and minimally corrective hyperparameters for the inaccurate prior 36
6.2 The original and minimally corrective hyperparameters for the accurate prior 39

# Chapter 1

# Introduction

The emergence of computational science has resulted in an increased reliance on simulation to supplement theory and experiments. Consequently, the ability to accurately determine the parameters of mathematical models and to quantify the uncertainty in their estimation, as well as the resulting model predictions, has become paramount. This lead to the creation of the field of *uncertainty quantification* which has found application in nearly every field of engineering and science. Uncertainty quantification, or UQ, has recently become an activity group of the Society of Industrial and Applied Mathematics (SIAM) and several new journals were founded to disseminate research in this area. A typical problem is the estimation and quantification of uncertainty of parameters when the observations are noisy, limited in number, and possibly indirect.

One of the main statistical methods to handle the problems in UQ is *Bayesian parameter estimation*. This method has the advantage of combining information from observations and estimated observational noise, in the form of a *likelihood distribution*, as well as expert judgement on the problem, in the form of a *prior distribution*, to form a distribution that encapsulates all the information one has on a problem. Expert judgement can be beneficial since there is no set of observations that will completely capture the essence of a physical system. However, since the prior distribution is purely based on expert judgement, the prior can introduce expert bias into the distribution produced by Bayesian parameter estimation. Additionally, existing approaches to check for error in this expert judgement are limited in scope or overly cautious.

This thesis presents a novel approach to address the issue of errors in expert judgement. The *minimally corrective, approximately recovering* prior (MCAR prior) is a combination of two steps:

1. The *approximately recovering* step which combines all the noisy observations to generate an approximation of the model via polynomial regression. Using this approximation of the model, we generate a guide for the parameter estimation process by trying to

recover the forcing term in a differential equation.

2. The *minimally corrective* step which takes an initial, expert supplied prior and makes corrections to it, using the result of the approximately recovering step as a guide.

In Chapter 2, we introduce inverse problems and summarize the approach of Bayesian parameter estimation. we introduce the elliptic equation in Chapter 3 and in Chapter 4 we specify a method to represent distributed parameters and a global approximation of the model. Chapter 5 is the main contribution of this thesis which motivates the idea of the minimally corrective, approximately recovering prior and presents an algorithm for determining the MCAR prior. Finally, in Chapter 6, we show a complete example of the MCAR prior applied to the estimation of a distributed parameter for an elliptic PDE.

# Chapter 2

# The Inverse Problem

In this chapter, we provide an overview of parameter estimation, inverse problems, and the Bayesian approach to solving them.

## 2.1 The General Inverse Problem

We begin by setting up a general formulation of the parameter estimation problem to lay out notation that we use throughout this thesis. Let $u$ denote the *state* of a physical phenomenon that we are interested in with $u : D \times Q \to Y$. The *domain* of the model is denoted by $D$, with states in $Y$, and $Q$ is the set of all possible choices of model parameters which we will refer to as the *parameter space*. Often, our state is the solution to a difference or differential equation and thus $u$ may not be available in closed form. In theses cases, $u$ is a solution to

$$L_q(u) = f, \tag{2.1}$$

where $L_q(\cdot)$ is a *differential or difference operator*, dependent on the parameter $q \in Q$, and $f$ is a forcing term that we assume to be independent of $q$. The *inverse problem* is to find $q$ given perfect knowledge of $f$ and $u$. In all practical applications, we have limited knowledge of $u$ and may only have access to *observations* of $u$ (denoted by $z$). The observations $z$ of our *observable space* $Z$ are limited in number and those observations we can make are subject to random noise. We encapsulate this reality into an *observational operator* $\mathcal{G} : Q \to Z$ of states that we can observe and a random variable $\eta$ which will augment the result of an experiment. That is, we assume there exists a parameter $q^* \in Q$, such that

$$z = \mathcal{G}(q^*) + \eta. \tag{2.2}$$

Using this framework, the problem we face is to combine the observations $z$ generated by (2.2) and domain knowledge of the model solution to generate the best estimate $\hat{q}$ of the true parameter set $q^*$. For a more general introduction of inverse problems, the reader is referred to [1] and [2].

## 2.2 Bayesian Parameter Estimation

One popular method of parameter estimation, *Bayesian parameter estimation*, combines expert domain knowledge with noisy observations using Bayes' rule. In this approach, the parameter estimate $\hat{q} = \hat{q}(\omega)$ is considered to be a random variable, with the realization indexed by $\omega$, and will be assigned a density or a measure.

### 2.2.1 Formulation

To set up the Bayesian approach, we have to provide two different but related densities: a prior and a likelihood. The *prior density* $\pi_0(q)$ encapsulates all our knowledge about the parameters we are trying to estimate before looking at the observational data. This is represents expert judgement, parameter limitations, and other beliefs about the system. The next step is to define is a *likelihood density* $\rho(z|q)$ that describes how likely the observed data set is to occur if we take $q$ to be the true parameter set. It is in the likelihood density where we encapsulate our observations of the system we are trying to model, as well as the solution to that model. Combining these two densities via Bayes' rule, we define the *posterior density* of $q$ as

$$\pi_z(q) = \frac{\rho(z|q)\pi_0(q)}{\int_Q \rho(z|q)\pi_0(q)\ \mathrm{d}q}, \tag{2.3}$$

which encapsulates all our information about the parameters and their uncertainties. In many applications, we are solely interested in the shape of the posterior and not the actual values of the density. Since the denominator is a constant independent of $q$, we can also use the proportional version of Bayes' rule,

$$\pi_z(q) \propto \rho(z|q)\pi_0(q), \tag{2.4}$$

which reduces the computational complexity of estimators such as expected values. The Bayesian parameter estimation process can be generalized to define Bayesian parameter estimation in the measure theoretic framework. Instead of defining a prior density, we define a *prior probability space* $(\Omega, \mathcal{F}, \mu_0)$ where $\mu_0$ is called the *prior measure*. This has the same role has the prior density but can be applied to a greater number of problems. Combining the prior measure with the likelihood density $\rho(z|q)$, we get a *posterior measure* $\mu_z$ on the measurable space $(\Omega, \mathcal{F})$ by the Radon-Nikodym derivative

$$\frac{\mathrm{d}\mu_z}{\mathrm{d}\mu_0} = \frac{1}{Z}\rho(z|q) \tag{2.5}$$

where $Z = Z(z)$ is a normalization constant independent of $q$. Equation (2.5) can also be expressed as

$$\mu_z(E) = \frac{1}{Z}\int_E \rho(z|q)\ \mathrm{d}\mu_0(q) \tag{2.6}$$

for $E \in \mathcal{F}$. For a more complete formulation of the Bayesian parameter estimation, the reader is referred to [3], [4], and [5].

The posterior density or measure contains all the information we have about the parameters of the model. In applying the posterior distribution to parameter estimation, we are particularly interested in

1. The *expected value* of the posterior

$$E[q] = \int_Q q \, \mathrm{d}\mu_z(q) = \int_Q q\pi_z(q) \, \mathrm{d}q, \tag{2.7}$$

   which can be interpreted as the average value of the posterior distribution.

2. The *maximum a posteriori* (MAP) given by

$$q_{MAP} = \arg\max_{q \in Q} \pi_z(q), \tag{2.8}$$

   which represents the parameter with the greatest density.

3. *Credible intervals* and *prediction intervals*.

### 2.2.2 An Example of Bayesian Parameter Estimation

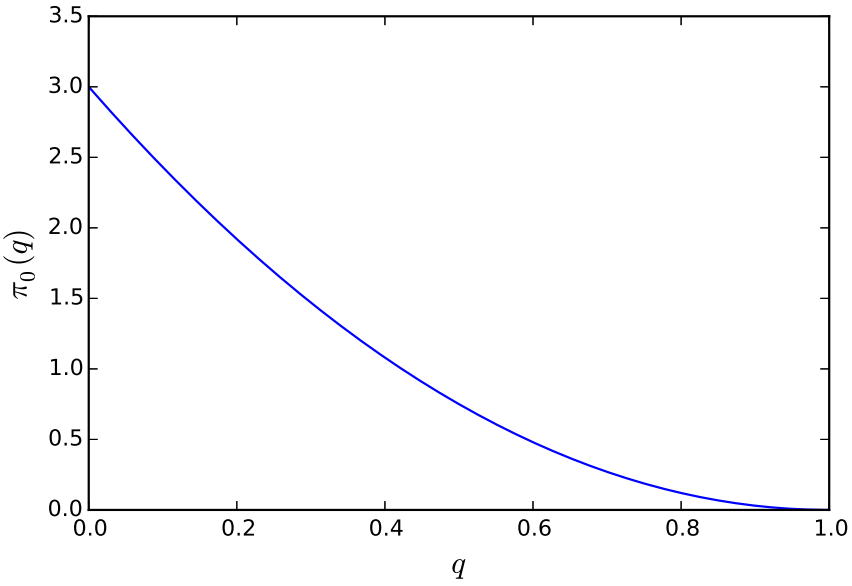
We want to take a moment and present a complete, toy example of the Bayesian parameter estimation process. Since this is a probabilistic approach, the most basic example involves flipping coins.

**Example 2.1.** Suppose we want to determine if a given coin is fair or biased to either heads or tails. Let $Q = [0, 1]$, the goal is to find $q^* \in [0, 1]$ which represents the probability of a tossed coin landing heads. If $q^* = .5$ then the coin is fair, if $q^* > .5$ then the coin is biased towards heads, and if $q^* < .5$ then the coin is biased towards tails. We consult an expert on coin construction who believes the coin is biased towards tails. The standard way to encapsulate this expert judgement is using the beta probability distribution which is defined by two hyper-parameters $\alpha$ and $\beta$ (The parameters specifying the shape of the distribution). In this case, we represent the expert judgement the hyper-parameters $\alpha = 1$ and $\beta = 3$ which has a prior probability density of

$$\pi_0(q) = \frac{1}{B(1, 3)}(1 - q)^2, \tag{2.9}$$

where $B(\alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$. The graph of this probability density function is shown in Figure 2.1. The expected value of this distribution is $E_{\pi_0}[q] = \frac{1}{4}$ which means we expect the coin to come up heads a quarter of the time and tails three quarter of the time.

Figure 2.1: The tails biased prior density for Example 2.1.




Next, we flip the coin twenty times which results in it coming up tails thirteen times and heads seven times. We can think of the observational operator as $z = (h, t) = \mathcal{G}(q^*)$ as the result of twenty flips of a coin with probability of heads is $q^*$ where $h$ is the number of heads and $t$ is the number of tails. So that means there exists a $q^* \in [0, 1]$ such that $z = (7, 13) = \mathcal{G}(q^*)$. The standard likelihood function for this type of problems is the binomial distribution with the number of heads as the number of successes and the number of tails as the number of failures. So the likelihood function when $z = (7, 13)$ is

$$\rho(z|q) = \binom{20}{7} q^7 (1-q)^{13}. \tag{2.10}$$

The likelihood function is shown in Figure 2.2.

Combining (2.9) and (2.10) using Bayes' rule, we get a beta posterior distribution $\alpha' = 8$ and $\beta' = 16$ which has a posterior density function of

$$\pi_z(q) = \frac{1}{B(8, 16)} q^7 (1-q)^{15}. \tag{2.11}$$

The posterior density function is shown in Figure 2.3. The expected value of the posterior is $E_{\pi_z}[q] = \frac{1}{3}$. This means that we should expect the coin to come up heads one-third of the time and tails two-thirds of the time. This means the posterior belief is that the coin is indeed biased towards tails.

Figure 2.2: The likelihood function with $z = (7, 13)$ for Example 2.1

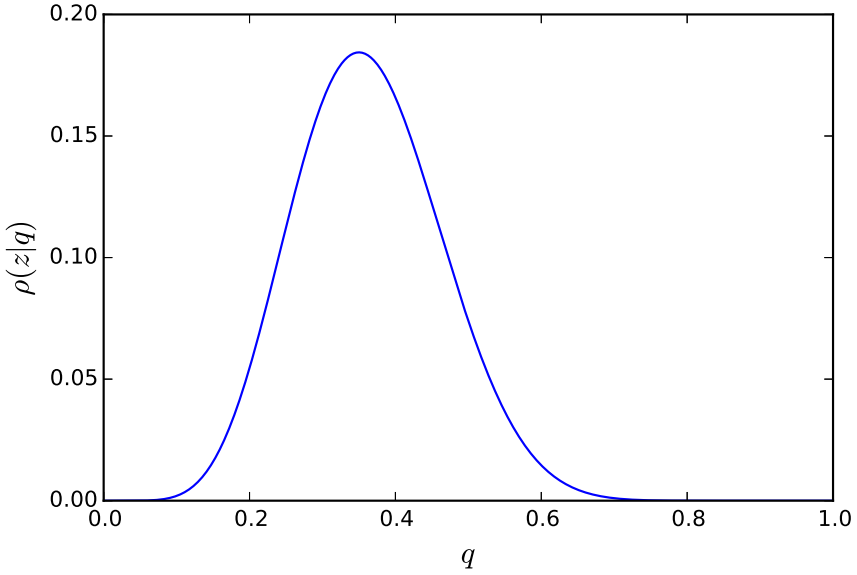
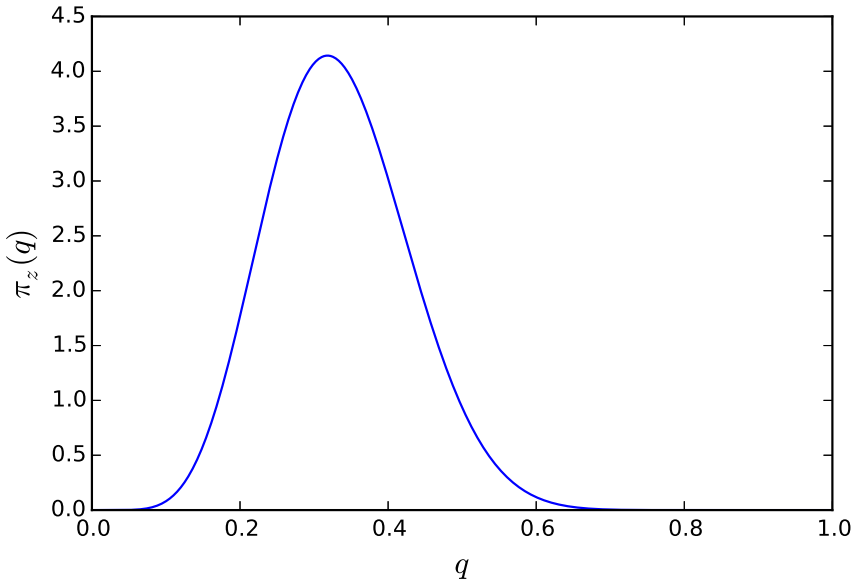



Figure 2.3: The posterior density function for Example 2.1




For a more complicated example of the Bayesian parameter estimation approach, we refer the reader to [6] and [7].

### 2.2.3 Criticism of Bayesian Parameter Estimation

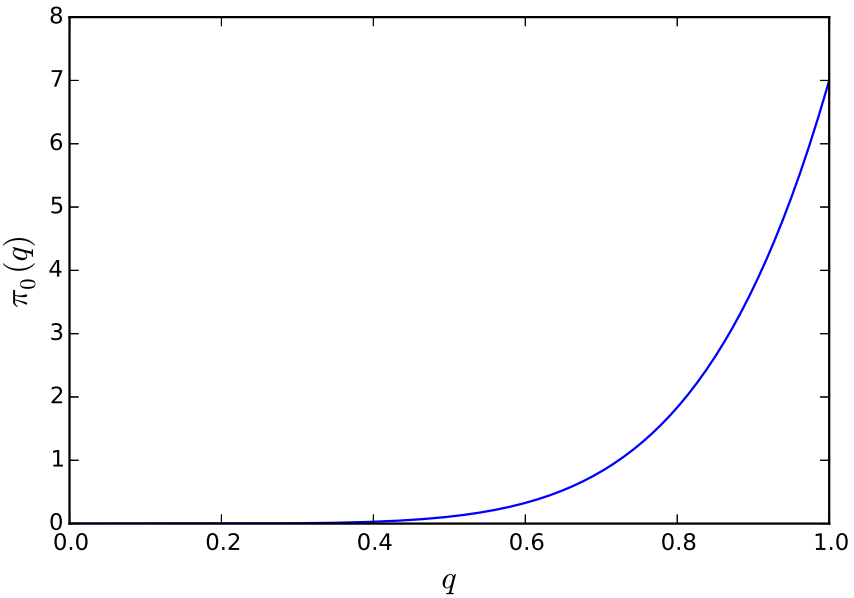
The main criticism of Bayesian parameter estimation is how the prior is defined. The traditional approach is for it to be define solely by the expert's judgement. In the previous example, the prior was defined by an expert and that expert happened to be correct. However, the expert could easily have been wrong and this will affect the posterior.

**Example 2.2.** Again, suppose we are trying to determine if the coin from the Example 2.1 is fair or biased to either heads or tails. In this example, the coin construction expert strongly believes the coin is biased towards heads. This belief is encapsulated into a new beta prior distribution with hyper-parameters $\alpha = 7$ and $\beta = 1$ which has a probability density of

$$\pi_0(q) = \frac{1}{B(7,1)} q^6. \tag{2.12}$$

The prior probability density function is shown in Figure 2.4. Suppose we see the same thir-

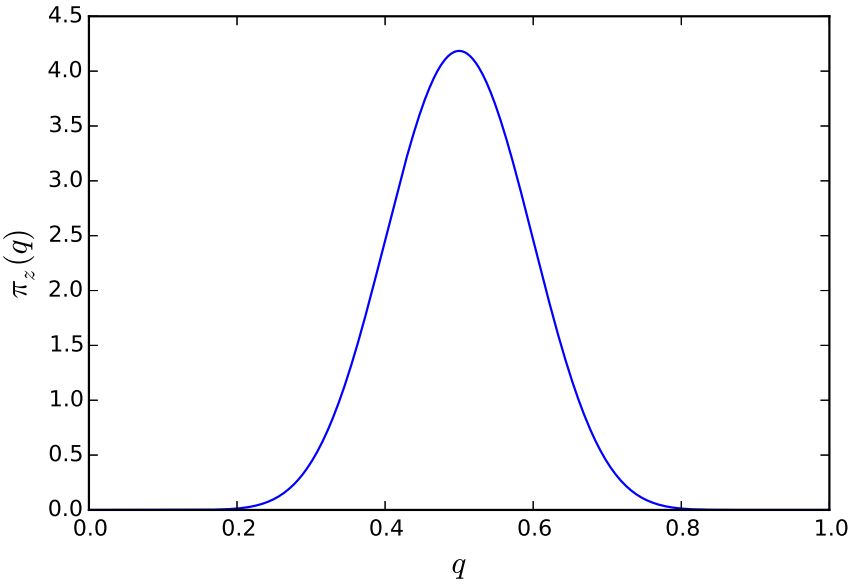
Figure 2.4: The prior density function for Example 2.2




teen tails and seven heads from twenty flips of the coin. This means the likelihood function is again (2.10). Using Bayes' rule on (2.10) and (2.12), we get the posterior distribution is beta with $\alpha' = 14$ and $\beta' = 14$ which has a posterior density function of

$$\pi_z(q) = \frac{1}{B(14,14)} q^{13}(1-q)^{13}. \tag{2.13}$$

Figure 2.5: The posterior density function for Example 2.2




The posterior probability density function is shown in figure 2.5. The expected value of the posterior distribution is $E_{\pi_z}[q] = \frac{1}{2}$. This means that it is the posterior belief that the coin is fair.

These two examples produced significantly different results. The only difference between them is the expert's belief in the coin. With only the information we have now, it is nearly impossible to determine which expert opinion is more correct. Thirteen tails and seven heads in twenty throws isn't improbable with a fair coin; however, the data was generated as a binomial random variable with $q^* = \frac{1}{4}$. This means the first expert was correct.

One method to handle this issue is robust Bayesian analysis described in [8] and [9]. This method primarily focuses on reducing the sensitivity of errors in the priors by defining classes of priors that spread out the probability across more of the parameter space than we would normally desire. The problem with these approaches is that they penalize experts with accurate judgement in preparation for those with poor judgement. This thesis aims to develop an approach that overcomes incorrect bias in the prior.

## 2.3 Sampling from the Posterior with Monte Carlo Methods

Most of the estimators that we are interested in on the posterior require generating a sample from the posterior distribution. This becomes a challenge for two main reasons. First, except in the special case of conjugate priors (see [10]), we shouldn't expect (2.3) to be one of the classical, closed form probability density functions. Second, in most Bayesian parameter estimation problems, the dimension of the parameter space $Q$ is large which makes accurate numerical quadratures computationally intractable. While there has been recent advances in spectral methods [11] and sparse grid quadratures [12], *Markov Chain Monte Carlo* methods are still the work horse of the Bayesian community. We conclude this chapter with a brie f overview of this important method. For more extensive introduction, we recommend consulting [1], [13], and [14].

*Monte Carlo* methods rely on generating a large number of random samples from the posterior distribution and then performing a numerical or statistical analysis on this sample instead of treating the intractable problem of generating those analysis results directly from the posterior. In the context of Bayesian analysis, the main class of Monte Carlo methods are *Markov Chain Monte Carlo* methods (MCMC). These methods are favored because, by constructing Markov Chains on the distribution, and using the posterior as the chain's stationary distribution, we are guaranteed a representative sample of the posterior distribution. In particular, we are going to focus on a version of MCMC called the *Metropolis algorithm*. This algorithm works as follows: For a given starting parameter sample $q^0$, a sample is taken from an easier to sample distribution based on the last point in the chain. The distribution is called the proposal distribution and is denoted by $J$. If the newly sampled point is acceptable to the posterior, then it is added to the chain. Otherwise, we set the next element in the chain to be $q^0$.

Suppose that $q^{k-1}$ was the last point in the Markov chain, the main choices of proposal distribution are $J(q^{k-1}) = \mathcal{N}(q^{k-1}, V)$ with $V$ being the covariance matrix of the posterior or $J(q^{k-1}) = \mathcal{N}(q^{k-1}, D)$ where $D$ is a diagonal matrix whose elements represents the scale of each parameter value. After defining the proposal distribution, we can use the Metropolis algorithm shown in Algorithm 1. When we say "with probability $\alpha = \min\{1, r\}$" in (2.15), we mean that we sample a number $\alpha^*$ from the uniform distribution on $[0, 1]$ and accept $q^*$ if $\alpha \geq \alpha^*$. After running this algorithm, we will have a representative sample of our posterior distribution on which we can perform analysis.

**Algorithm 1** The Metropolis MCMC Algorithm

**Require:** The likelihood function $\rho(z|q)$, the prior $\pi_0(q)$, and the proposal distribution $J(q^k|q^{k-1})$, an initial $q^0$ such that $\rho(z|q^0)\pi_0(q^0) > 0$, and an integer $M > 0$.

**for** $k = 1, \ldots, M$ **do**

Take a sample $q^* \sim J(q^{k-1})$.

Compute the ratio

$$r = \frac{\rho(z|q^*)\pi_0(q^*)}{\rho(z|q^{k-1})\pi_0(q^{k-1})}. \tag{2.14}$$

Set

$$q^k = \begin{cases} q^* & : \text{ with probability } \alpha = \min\{1, r\} \\ q^{k-1} & : \text{ otherwise} \end{cases}. \tag{2.15}$$

**end for**

# Chapter 3

# The Elliptic Equation

Consider parameter estimates for the one dimensional elliptic equation of the form

$$-\frac{\mathrm{d}}{\mathrm{d}x}\left(q(x)\frac{\mathrm{d}}{\mathrm{d}x}u(x)\right) = f(x), \quad x \in D = [a,b] \tag{3.1}$$

where $q(x)$ is a positive, *distributed parameter* and $f(x)$ is the *forcing term*. We focus on solutions in the classical sense, so we add the requirements that $u \in C^2(D)$, $q \in C^1(D)$, and $f \in C(D)$. Additionally, we need to impose boundary conditions of $u(a) = u_a$ and $u(b) = u_b$. One physical interpretation of this equation is the temperature field $u$ in a one dimensional rod subject to a distributed internal and external heat source $f$. Stating this in the form (2.1), we see that

$$L_q(u) = -\frac{\mathrm{d}}{\mathrm{d}x}\left(q(x)\frac{\mathrm{d}}{\mathrm{d}x}u(x)\right), \tag{3.2}$$

which is a linear in both $q$ and $u$. Sometimes, we will find it advantageous to work with the form

$$L_q(u) = -\left(\frac{\mathrm{d}}{\mathrm{d}x}q(x)\frac{\mathrm{d}}{\mathrm{d}x}u(x) + q(x)\frac{\mathrm{d}^2}{\mathrm{d}x^2}u(x)\right). \tag{3.3}$$

There is a vast body of work associated with equations of this type; however, we want to focus on two properties: an idea of uniqueness of parameters and a way to numerically solve (3.1). For more information on elliptic equations, we suggest the reader consult [15].

## 3.1 Identifiability of the Parameters

In this section, we present conditions that guarantee existence and uniqueness of solutions of the parameter estimation problem. For existence, we provide conditions such that there must be a $q$ for which (3.1) generates the observation $z$. For uniqueness, we refer to the usual notion of *identifiability*.

**Definition 3.1** (Identifiability)**.** Let $\Phi$ be the parameter-to-output mapping of (3.1). The parameter $q$ is identifiable at $q^*$ with respect to $Q$ if for any $q \in Q$, we have that $\Phi(q) = \Phi(q^*)$ implies that $q = q^*$.

While this seems to be a one-to-one requirement on the parameter-to-output map, the fact that we fix $q^*$ means that we are only concerned about the mapping being one-to-one when one of the inputs is our desired parameter set. Checking if a parameter $q^*$ is identifiable, we pick any $q \in Q$. Then we determine the solutions $u^*$ and $u$ to (3.1) for those parameters with the same forcing function. Identifiability can be guaranteed by applying the following theorem.

**Theorem 3.2.** *If $q^*, q \in L^\infty$, $f \in (H^1)^*$, and $u^*, u \in H^1$, then*

$$\|(q^* - q)u_x^*\|_{L^2} \leq 2\|q\|_{L^\infty}\|u^* - u\|_{H^1}. \tag{3.4}$$

*Proof.* See [16]. □

Using this theorem, we have that, when $u = u^*$, we know $q^* = q$ except at those values of $x \in D$ where $u_x^*(x) = 0$. That is $q^*$ will be identifiable (from full observation) at all $x \in D$ except where $\nabla u(x) = 0$. While the preceding theorem has specific technical requirements, we have restricted our attention to classical solutions to the elliptic equation and can assume they are automatically satisfied. Theorem 3.2 will be sufficient for this thesis, however there are several other results on identifiability of the elliptic equation that generalize the regularity assumption and include other observation operators. For those results, we recommend consulting [16], [17], and [18].

## 3.2 The Finite Difference Approximation of $L_q(u)$

We need a method to approximate solutions to (3.1) and choose to use the finite difference method described in [19]. Start by defining a spatial discretization parameter $N_h > 1$ which is the number of equally spaced subintervals of $[a, b]$. Define $h = \frac{b-a}{N_h}$ and our difference nodes as $\mathcal{D}_h = \{x_i : x_i = a + i \times h, i = 0, 1, \ldots, N_h\}$. We define approximations of $\nabla u(x)$ and $\nabla^2 u(x)$ by

$$\nabla_h u(x_i) = \frac{u(x_{i+1}) - u(x_{i-1})}{2h} \tag{3.5}$$

and

$$\nabla_h^2 u(x_i) = \frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1})}{h^2}. \tag{3.6}$$

Applying these approximations to (3.3), we can approximate of $L_q(u)$ as

$$L_q^{(h)}(u)[x_i] = -\left(\nabla q(x_i)\frac{u(x_{i+1}) - u(x_{i-1})}{2h} + q(x_i)\frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1})}{h^2}\right). \tag{3.7}$$

Next, we need to discretize $\nabla q$ for which we use the centered derivative of

$$\nabla_h q(x_i) = \frac{q(x_{i+1}) - q(x_{i-1})}{2h}. \tag{3.8}$$

For this to satisfy (3.1) at $x_i$, we need

$$-\left(\frac{q(x_{i+1}) - q(x_{i-1})}{2h}\frac{u(x_{i+1}) - u(x_{i-1})}{2h} + q(x_i)\frac{u(x_{i+1}) - 2u(x_i) + u(x_{i-1})}{h^2}\right) = f(x_i) \tag{3.9}$$

which can be rearranged as

$$a_i u(x_{i-1}) + b_i u(x_i) + c_i u(x_{i+1}) = -h^2 f(x_i) \tag{3.10}$$

where

$$a_i = \frac{1}{4}q(x_{i-1}) + q(x_i) - \frac{1}{4}q(x_{i+1}) \tag{3.11}$$

$$b_i = -2q(x_i) \tag{3.12}$$

$$c_i = -\frac{1}{4}q(x_{i-1}) + q(x_i) + \frac{1}{4}q(x_{i+1}). \tag{3.13}$$

We know the values of $u(x_0) = u(a) = u_a$ and $u(x_{N_h}) = u(b) = u_b$, so they can be applied directly to the system of equations (3.10). The other values of $u(x_i)$ are unknown so we will need to solve for them. By evaluating (3.10) for $i = 1, \ldots, N_h - 1$, we get a tridiagonal system. Solving that system will give us our approximate value for $u(x)$ at each $x_i$.

# Chapter 4

# Representation of Distributed Parameters

The focus of this thesis is on distributed parameters, so we need to specify a representation for distributed parameters. Since we treat the parameters as stochastic processes that are distributed according to the posterior, we will use the Karhunen-Loève representation. This represents a distributed parameter using a series of the eigenfunctions derived from the parameters covariance function. Additionally, we will use polynomial regression to convert the noisy, finite observations into a right-hand side function for (2.1).

## 4.1 Karhunen-Loève Representation

Suppose we treat our unknown distributed parameter $q(x)$ as a stochastic process $q(x, \omega)$, distributed according to the probability space $(\Omega, \mathcal{F}, P)$ for each $x \in D$. We assume that $q(x, \omega) \in L^2(D) \times L^2(\Omega, \mathcal{F}, P)$. The standard way to represent stochastic processes is using the Karhunen-Loève representation (KL representation) which is developed using the Karhunen-Loève theorem. We start by defining $K_q(x, y) = E[q(x, \cdot)q(y, \cdot)]$ to be the parameter's covariance process. Next, define a linear operator $T_{K_q} : L^2(D) \to L^2(D)$ by

$$T_{K_q}(f)[y] = \int_D K_q(x, y) f(x) \, \mathrm{d}x. \tag{4.1}$$

Using this linear operator, we define the eigenvalues $\lambda_k$ and eigenfunctions $e_k(x)$ that satisfy

$$\int_D K_q(x, y) e_k(x) \, \mathrm{d}x = \lambda_k e_k(y). \tag{4.2}$$

This now allows us to define the Karhunen-Loève theorem.

**Theorem 4.1.** *Let $q \in L^2(D) \times L^2(\Omega, \mathcal{F}, P)$ with a continuous covariance function $K_q(x, y)$. Then the eigenfunctions $\{e_k(x)\}$ of $K_q(x, y)$ from (4.2) form an orthonormal basis of $L^2(D)$. Additionally, we can represent $q(x, \omega)$ by*

$$q(x, \omega) = E\left[q(x, \omega)\right] + \sum_{k=1}^{\infty} Q_n(\omega) e_k(x), \tag{4.3}$$

*where*

$$Q_n(\omega) = \int_D \left(q(x, \omega) - E\left[q(x, \omega)\right]\right) e_k(x)\, \mathrm{d}x \tag{4.4}$$

*and the random variables $Q_n(\omega)$ are zero-mean, uncorrelated, and have variance $\lambda_k$.*

*Proof.* See [20]. □

We now have a way to represent the stochastic process using an orthonormal decomposition. However, it is inconvenient that we have to handle the expected value and centered version of the process separately. Thus we modify the representation as follows. Since the set $\{e_k\}$ is a basis for $L^2(D)$, we can represent the expected value as

$$E\left[q(x, \omega)\right] = \sum_{k=1}^{\infty} A_k e_k(x) \tag{4.5}$$

where

$$A_k = \int_D E\left[q(x, \omega)\right] e_k(x)\, \mathrm{d}x. \tag{4.6}$$

We apply (4.5) and (4.6) to (4.3) and (4.4) to get a modified representation.

**Definition 4.2** (Modified KL Representation)**.** The *modified KL representation* is defined by

$$q(x, \omega) = \sum_{k=1}^{\infty} \hat{Q}_n(\omega) e_k(x) \tag{4.7}$$

where

$$\hat{Q}_k = Q_k + A_k = \int_D q(x, \omega) e_k(x)\, \mathrm{d}x. \tag{4.8}$$

The modification results in the KL representation taking the form of a standard orthonormal decomposition for the basis $\{e_k\}$. We can also use the truncated version for approximations. Clearly the coefficients of the modification do not have zero mean, but they are uncorrelated.

**Lemma 4.3.** The $\hat{Q}_k$ defined by (4.8) are uncorrelated.

*Proof.* For any indices $i, j$,

$$E\left[\hat{Q}_i\hat{Q}_j\right] = E\left[Q_iQ_j\right] + A_iE\left[Q_j\right] + A_jE\left[Q_i\right] + A_iA_j = A_iA_j. \tag{4.9}$$

Additionally, we see that

$$E[\hat{Q}_i] = E\left[Q_i\right] + A_i = A_i \qquad \text{and} \qquad E[\hat{Q}_j] = E\left[Q_j\right] + A_j = A_j. \tag{4.10}$$

Therefore,

$$E\left[\hat{Q}_i\hat{Q}_j\right] - E[\hat{Q}_i]E[\hat{Q}_j] = 0 \tag{4.11}$$

which implies $\hat{Q}_i$ and $\hat{Q}_j$ are uncorrelated. □

To conclude our discussion of the Karhunen-Loève representation, we discuss the case when we only have a probability distribution of $q(x, \omega)$ at a finite set $\{x_i\}_{i=1}^N \subset D$. In this case, instead of a covariance function, we get a covariance matrix $\Sigma$ where $\Sigma_{ij} = E\left[q(x_i, \omega)q(x_j, \omega)\right]$. We develop a reduced basis $\{e_i\}_{i=1}^N$ from the solutions to the eigenvalue problem

$$\Sigma e_i = \lambda_i e_i. \tag{4.12}$$

To develop basis of functions for f $L^2(D)$, we take the values of $e_i$ and use them as interpolation points at the points $\{x_i\}_{i=1}^N$. Then we develop the coefficients (4.8) using numerical quadrature techniques. An important result requires $q(x_i, \omega)$ being uncorrelated.

**Lemma 4.4.** If the random variables $\{q(x_i, \omega)\}_{i=1}^N$ are uncorrelated, then $e_i$ will be the $i$th canonical basis vector of $\mathbb{R}^N$.

*Proof.* Since the random variables are uncorrelated, we know $\Sigma_{ij} = E\left[q(x_i, \omega)q(x_j, \omega)\right] = 0$ for all $i \neq j$. Thus, $\Sigma$ will be a diagonal matrix and its eigenvectors are the canonical basis vectors of $\mathbb{R}^N$. □

The importance of this lemma is that the interpolation of $e_k$ will be one at $x_k$ and zero at all other points. In this case instead of interpolating, we will use the standard piecewise-linear finite element basis.

## 4.2 Polynomial Regression

Now that we have a representation for our parameter $q$, we need to handle how to turn our noisy, finite observations of the state $u$ into a global approximation across $D$. To handle this challenge, we use polynomial regression. Polynomial regression is a special case of multiple linear regression where we fit the coefficients of a polynomial to the data. Suppose we have a set of data $\{y_i\}_{i=1}^M$ where $y_i$ is observed at $x_i$. We assume that there is a function $f$ which

maps the inputs $x_i$ to $y_i$ with addtional zero mean, random noise. In polynomial regression, we are trying to approximated $f$ using a polynomial

$$y_i = \alpha_0 + \alpha_1 x_i + \alpha_2 x_i^2 + \ldots + \alpha_n x_i^n + \varepsilon_i, \tag{4.13}$$

where $\varepsilon_i$ is a zero mean, unobserved random error. Evaluating this at every $i = 1, \ldots, M$, we can write this as a linear system

$$\boldsymbol{y} = X\boldsymbol{\alpha} + \boldsymbol{\varepsilon} \tag{4.14}$$

where $\boldsymbol{y} = (y_1, \ldots, y_M)^T$, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^T$, $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_M)^T$, and $X_{i,j} = x_i^{j-1}$. This has an easy to compute ordinary least squares estimate for the weights $\boldsymbol{\alpha}$.

**Lemma 4.5.** The ordinary least squares estimate of $\boldsymbol{\alpha}$ in (4.14) is

$$\boldsymbol{\alpha} = (X^T X)^{-1} X^T \boldsymbol{y}. \tag{4.15}$$

*Proof.* See [21]. □

Using these coefficients, our approximation of $f$ is

$$f_n(x) = \alpha_0 + \alpha_1 x + \ldots + \alpha_n x^n. \tag{4.16}$$

The final step is to choose the degree of the polynomial. For this, we will consider the bias vs. variance tradeoff presented in [22]. Bias measures the difference between the regression model's prediction and the true mean caused by our model being too simple: high bias is an indicator of under-fitting. Variance in this context is the variance of the regression model with respect to the mean caused by our model being too complex: high variance is an indicator of overfitting. Simply put, if we have high bias then we must increase the degree of polynomial we are using and if we have high variance then we must decrease the degree of the polynomial we are using. Since we only have a finite number of observations of the state to determine our fit, we can't get exact measures of our bias and variance; however, we can diagnose when (4.16) is suffering from high variance or high bias. We start by randomly partitioning our data set into two different sets: around 80% of the data goes into a training set $S_t$ that we use to generate $\boldsymbol{\alpha}$ by (4.15) and the other 20% goes into a cross-validation set $S_{cv}$ that we will use to test the fit. Next, define two error functions, a training error

$$J_t(n) = \frac{1}{2|S_t|} \sum_{(x,y)\in S_t} (f_n(x) - y)^2 \tag{4.17}$$

and a cross validation error

$$J_{cv}(n) = \frac{1}{2|S_{cv}|} \sum_{(x,y)\in S_{cv}} (f_n(x) - y)^2 \tag{4.18}$$

where $|S_t|$ represents the number of elements in $S_t$ and $|S_{cv}|$ represents the number of elements in $S_{cv}$. Finally we plot these two error functions. When both the bias and cross-validation lines are high, (4.16) is experiencing high bias. When the bias line is small and the cross-validation line is high, (4.16) is experiencing high variance. The ideal $n$ is one that simultaneously minimizes $J_{cv}$ and $J_t$.

# Chapter 5

# MCAR Priors

In the general setting of Bayesian parameter estimation, determining the accuracy of a parameter estimate is extremely difficult since we only have access to the observed experimental data and expert opinion on the parameters. However, we can defer back to the model that we are attempting to fit as a guide to develop ideal priors for the Bayesian parameter estimation process. This will lead to the idea of the class of approximately recovering priors developed in Section 5.1.2. Commonly, we will find that the original prior is outside of this class. To correct the original prior to become approximately recovering while preserving as much of the original prior's information as possible, we will develop the idea of a minimally corrected prior in Section 5.2 and combine these two new developments into one concept of a *minimally corrected, approximately recovering prior* (MCAR prior). To conclude, in Section 5.3 we will present an algorithm of how to determine the MCAR prior.

## 5.1 Recovering and Approximately Recovering Priors

Recall that our model $u$ is a solution to a differential or difference operator of the form

$$L(u \,|\, q) = f. \tag{5.1}$$

During parameter estimation, $f$ is deterministic and completely known. Using (5.1) to combine an estimate $\hat{q}$ of $q$, developed though the Bayesian parameter estimation process, with an estimate $\hat{u}$ of $u$, we get an estimate $\hat{f}$ of $f$. An accurate prior will result in this estimator being unbiased, that is $E[\hat{f}] = f$ which we can think of as the prior recovering our forcing term.

### 5.1.1 Recovering Priors

To start, we are going to assume we have perfect information about the values of our model $u$ on the domain $D$. Throughout the Bayesian parameter estimation process, we treat our parameter estimate $\hat{q} = \hat{q}(\omega)$ as a random variable distributed according to the posterior density $\pi_z(q)$ which induced from the prior by (2.3). From our posterior, we will develop a stochastic process estimating the forcing term $f$ via

$$\hat{f}\left(u \,|\, \hat{q}(\omega)\right) = L\left(u \,|\, \hat{q}(\omega)\right). \tag{5.2}$$

It is key to note that $\hat{f}$ is a random variable. Since we have perfect knowledge of $u$ and the operator $L$ is deterministic, $\hat{q}$ is the only source of randomness for $\hat{f}$. This means the distribution of $\hat{f}$ will be dependent on the Bayesian posterior of $\hat{q}$. If we keep the likelihood density fixed, the posterior $\pi_y$ is solely dependent on the choice of prior $\pi_0$ meaning that the only input to the distribution of $\hat{f}$ is the prior. This gives us a natural definition of recovering priors.

**Definition 5.1** (Recovering Priors)**.** A prior density $\pi_0$ is *recovering* if, by (2.3), it generates a posterior $\pi_y(\cdot)$ such that $E_{\pi_y}\left[\hat{f}\left(u \,|\, \hat{q}(\omega)\right)\right] = f$ in the sense of equivalence classes of functions.

The next lemma will show how satisfying this requirement will provide a parameter set for a large class of model operators.

**Lemma 5.2.** If $\pi_0$ is a recovering prior and the operator $L(u \,|\, q)$ is affine with respect to the parameter, then $u$ satisfies

$$L\left(u \,\middle|\, E_{\pi_y}\left[\hat{q}(\omega)\right]\right) = f. \tag{5.3}$$

That is $E_{\pi_y}[\hat{q}]$ is a parameter satisfying our model operator.

*Proof.* Let $\pi_y$ be the posterior density generated from $\pi_0$ by (2.3). Since $L_q(u)$ is affine with respect to $q$, we can write it as

$$L(u \,|\, q) = A(u \,|\, q) + B(u) \tag{5.4}$$

where $A(u \,|\, q)$ is linear with respect to $q$ and $B(u)$ is independent of $q$. By linearity, we have

$$\begin{aligned} f &= E_{\pi_y}\left[\hat{f}\left(u \,|\, \hat{q}(\omega)\right)\right] = E_{\pi_y}\left[A\left(u \,|\, \hat{q}(\omega)\right) + B(u)\right] = A\left(u \,\middle|\, E_{\pi_y}\left[\hat{q}(\omega)\right]\right) + B(u) \\ &= L\left(u \,\middle|\, E_{\pi_y}\left[\hat{q}(\omega)\right]\right). \end{aligned}$$

□

This means that if our prior is recovering, we can easily calculate a parameter value satisfying any inverse problem where the model operator is affine in the parameter space. Affine model operators include a wide variety of ordinary and partial differential equations used in practice.

### 5.1.2 Approximately Recovering Priors

While the idea of recovering priors and its property Lemma 5.2 are ideal, they require complete knowledge of the model $u(x)$ which we do not have. Since we don't have that, we are going to use an approximation $\hat{u}$ of $u$ generated by (4.16) to develop a best approximation of a recovering prior. An ideal approximation of a recovering prior would both converge to a recovering prior as $\hat{u}(x) \to u(x)$ and would have a version of the affine parameter property of Lemma 5.2. Since we are using a polynomial regression estimate of $u(x)$ for $\hat{u}(x)$, we know as the number of sensors $N_s$, number of readings $N_r$, and the regression model degree go to infinity, $\hat{u}(x) \to u(x)$. Additionally, we will see that as $\hat{u}(x) \to u(x)$, the approximations of the recovering prior will converge to a true recovering prior.

The first step is to take the observed data of $u(x)$ and form a polynomial regression approximation according the procedure discussed in Section 4.2. Since we are interested in the derivative of this approximation, minimizing the variance of the regression is extremely important. Using this approximation, define a new estimate of $f$ by

$$\hat{f}^{\hat{u}}\left(x \,|\, \omega\right) = L\left(\hat{u}(x) \,|\, \hat{q}(x \,|\, \omega)\right). \tag{5.5}$$

This new estimate gives us a natural definition for our approximately recovering priors.

**Definition 5.3** (Approximately Recovering Priors)**.** A prior density $\pi_0^{\hat{u}}$ is recovering if, by (2.3), it generates a posterior $\pi_z^{\hat{u}}$ such that $E_{\pi_z^{\hat{u}}}\left[\hat{f}^{\hat{u}}(x \,|\, \omega)\right] = f$ in the sense of equivalence classes of functions.

It is key to note that the expected value of the approximately recovering posterior should not be expected to be an acceptable parameter set as it was for the recovering prior in Lemma 5.2. The inaccuracies in the approximation $\hat{u}$ of $u$ will propagate into $\hat{f}^{\hat{u}}$ and then into $\pi_0^{\hat{u}}$. However, we know that $\hat{u}$ is the best linear approximation with the information we have and of that model degree, so we can expect it to be reasonably accurate. What this means is we should think of the expected value of our approximately recovering posterior as a guide to correct large inaccuracies in our Bayesian parameter estimation process.

The first thing that we would like to verify is that our approximately recovering priors will converge to a recovering prior in case of parameter affine model operators.

**Lemma 5.4.** Suppose we have a sequence $\hat{u}_n$ such that $\hat{u}_n \to u$ and suppose $L$ is continuous with respect to the state argument, then any sequence of approximately recovering prior $\left\{\pi_0^{\hat{u}_n}\right\}$ will converge to a recovering prior.

*Proof.* Let $\hat{u}_n$ be a sequence of approximations of $u$ such that $\hat{u}_n \to u$ and let $\pi_0^{\hat{u}_n}$ be an approximately recovering prior with respect to $\hat{u}_n$. We know that $E_{\pi_z^{\hat{u}_n}}\left[\hat{f}^{\hat{u}_n}(x \,|\, \omega)\right] = f$ for all $n$. Passing $\hat{u}_n \to u$, we know that $E_{\pi_z^{\hat{u}}}\left[\hat{f}^{\hat{u}}(x \,|\, \omega)\right] = f$. In this case the definition of

approximately recovering and recovering are the same meaning that the limit of $\pi_0^{\hat{u}_n}$ is a recovering prior. □

Now that we have seen that our approximately recovering priors converge to a recovering prior, we need to turn our attention to generating the approximately recovering prior which we will focus on the affine case. Define $E[\hat{q}(x)]$ to be approximated by a truncated power series with unknown coefficients, that is

$$E_{\pi_z^{\hat{u}}}[\hat{q}(x)] = \sum_{j=1}^{N_q} \alpha_j x^j. \tag{5.6}$$

We consider the power series over a different representation because of the ease of taking derivatives and because it multiplies nicely with different derivatives of $\hat{u}(x)$. To determine the values of $\alpha_j$ to make our posterior recovering, we start by using the same argument as in the proof of Lemma 5.2 to get

$$f(x) = E_{\pi_z}\left[L\left(\hat{u}(x) \,|\, \hat{q}(x \,|\, \omega)\right)\right] = L\left(\hat{u}(x) \,|\, E_{\pi_z}\left[\hat{q}(x \,|\, \omega)\right]\right) = L\left(\hat{u}(x) \,\middle|\, \sum_{j=1}^{N_q} \alpha_j x^j\right). \tag{5.7}$$

Since we are only considering $L$ which are affine on the parameter space, (5.7) can be rewritten as

$$f(x) - A(\hat{u}(x)) = \sum_{j=1}^{N_q} \alpha_j B\left(\hat{u}(x) \,\middle|\, x^j\right). \tag{5.8}$$

where $A(\cdot)$ is the parts of the model operator independent of the parameter and $B$ is linear with respect to the parameter space. At any $x \in D$, (5.8) gives a linear combination of the unknown coefficients $\alpha_j$. There are numerous ways to now handle the calculations, we choose to make an over determined system and calculate the linear least squares approximation. This is to reduce the impact of local approximation errors in $\hat{u}$. To complete the calculation start by choosing $\{x_1, \ldots, x_{N_s}\} \subseteq D$ such that $N_s > N_q$. Then calculate the vector $\mathbf{b} = (b_1, \ldots, b_{N_s})^T$ such that $b_i = f(x_i) - A(\hat{u}(x_i))$ as well as the $N_s \times N_q$ matrix $X$ where $[X]_{i,j} = B\left(\hat{u}(x_i) \,\middle|\, x_i^j\right)$. The resultant $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{N_q})$ is defined by

$$\boldsymbol{\alpha} = (X^T X)^{-1} X^T \mathbf{b}. \tag{5.9}$$

For the full derivation of the solution to the general least squares problem, the reader is referred to [23]. After determining $\boldsymbol{\alpha}$, we can say that our prior is recovering if it induces a posterior such that $E_{\pi_z^{\hat{u}}}[\hat{q}(x)] = \sum_{j=1}^{N_q} \alpha_j x^j$ for every $x \in D$.

## 5.2 Minimally Corrected Approximately Recovering Priors

Bayesian priors are normally defined by the judgement of subject matter experts. While the errors in judgement of the experts cause errors in the parameter estimation process, the priors may still contain useful information meaning we don't want to completely ignore the expert judgement. However, we also want our priors to be recovering. To handle these two goals, we will develop an optimization problem of minimizing information loss subject to our prior being recovering.

### 5.2.1 Quantifying Information Loss

Let $\pi_0$ be the prior density supplied by the subject matter expert. Before correcting this density, we need to determine how to quantify the amount of information lost by modifying the prior. There are several different ways to quantify the amount of information lost by using a probability distribution $Q$ to a approximate a probability distribution $P$. We choose the Kullback-Leibler divergence of

$$D_{KL}(P\|Q) = \int_X p(x) \ln\left(\frac{p(x)}{q(x)}\right)\ \mathrm{d}x \tag{5.10}$$

where $p$ and $q$ are the density functions for $P$ and $Q$ respectfully. This is equivalent to

$$D_{KL}(P\|Q) = \int_X p(x) \ln\left(p(x)\right)\ \mathrm{d}x - \int_X p(x) \ln\left(q(x)\right)\ \mathrm{d}x. \tag{5.11}$$

We are concerned with the minimization of $D_{KL}(P\|Q)$ for fixed $P$ and can therefore focus on maximizing

$$d_p(q) = \int_X p(x) \ln\left(q(x)\right)\ \mathrm{d}x. \tag{5.12}$$

Using (5.12), we can easily define the amount of information lost by using $\tilde{\pi}_0$ in place of $\pi_0$.

**Definition 5.5** (Prior Loss Function)**.** Let $\pi_0$ be our prior density supplied by our subject matter expert and $\tilde{\pi}_0$ be the prior density we are consider replacing it with. We define the *prior loss function* $\ell_{\pi_0}(\tilde{\pi}_0)$ by

$$\ell_{\pi_0}(\tilde{\pi}_0) = d_{\pi_0}(\tilde{\pi}_0). \tag{5.13}$$

### 5.2.2 Correcting the Prior

Now that we have a quantification for the amount of information we are losing by modifying the prior density, we can talk about what our desired modification will be.

**Definition 5.6** (Correcting Mapping and Corrected Prior)**.** Let $\mathbb{P}$ be the space of all prior densities. A mapping $T : \mathbb{P} \to \mathbb{P}$ is a *corrective mapping for* $\pi_0$ if $T(\pi_0)$ is a recovering prior. The new prior $T(\pi_0)$ is called the *corrected prior*. The set $\mathbb{P}^C$ is the set of all corrective mappings for $\pi_0$.

It will be an extremely rare case where there is only one corrective mapping for $\pi_0$. It is our objective to find the mapping that corrects the prior while preserving as much information from the original prior as possible. Such a mapping is called a *minimally corrective mapping* and is defined as follows:

**Definition 5.7** (Minimally Corrective Mapping)**.** Given a provided prior $\pi_0$, a mapping $T^* \in \mathbb{P}^C$ is a *minimally corrective mapping* if

$$\ell_{\pi_0}\left(T^*(\pi_0)\right) \leq \ell_{\pi_0}\left(T(\pi_0)\right) \tag{5.14}$$

for all $T \in \mathbb{P}^C$, where $\ell_{\pi_0}$ is defined as in Definition 5.5.

These definitions requires our priors to be recovering, however we can also apply these with approximately recovering priors.

**Definition 5.8** (Minimally Corrective, Approximately Recovering Priors)**.** Let $\hat{\mathbb{P}}^C$ be the set of all mappings $T : \mathbb{P} \to \mathbb{P}$ such that $T(\pi_0)$ is approximately recovering. An element $T^* \in \hat{\mathbb{P}}^C$ is a *minimally corrective, approximately recovering mapping (MCAR mapping)* if

$$\ell_{\pi_0}\left(T^*(\pi_0)\right) \leq \ell_{\pi_0}\left(T(\pi_0)\right) \tag{5.15}$$

for all $T \in \hat{\mathbb{P}}^C$, where $\ell_{\pi_0}$ is defined as in Definition 5.5. The prior $T^*(\pi_0)$ is called the *minimally corrected, approximately recovering prior (MCAR prior).*

This definition can easily be reformulated as an optimization problem.

**Problem 5.9.** Find a mapping $T^* \in \hat{\mathbb{P}}^C$ which satisfies

$$\min \ell_{\pi_0}(T(\pi_0)) \tag{5.16}$$

subject to

$$E_{\pi_z}[q] = \sum_{j=1}^{N_q} \alpha_j x^j \tag{5.17}$$

where $\pi_z$ is the posterior induced by $T(\pi_0)$.

### 5.2.3 Determining the Corrected Prior

We have not yet placed any requirements on $\mathbb{P}$ except that each member is a probability density function. In order to simplify our computations, we place two requirements on the space. The first is that the members of $\mathbb{P}$ are from the same type of parametrized distribution. With out this restriction, we would need to simultaneously handle a model selection problem along side a hyper-parameter estimation problem. By adding this restriction, determining the optimal member of $\mathbb{P}$ solely becomes a hyper-parameter optimization problem. Also it makes members of $\mathbb{P}^C$ functions on the hyper-parameters of the family of distributions. The second requirement we place on $\mathbb{P}$ is that the family of distributions must be differentiable with respect to the hyper-parameters. This requirement eases the computational complexity of the optimization Problem 5.9 while not restricting us from using most of the popular families of probability distributions.

With these new requirements, we can now attack Problem 5.9. The main tool we are going to use is the penalty method described in [24]. This changes a constrained optimization problem into an unconstrained optimization problem by defining an auxiliary function

$$h(q, \lambda) = \ell_{\pi_0}(T(\pi_0)) + \lambda \left\| E_{\pi_z}[q(\boldsymbol{x}_s)] - E[\hat{q}(\boldsymbol{x}_s)] \right\|_\infty \tag{5.18}$$

for a $\lambda > 0$ and where $\boldsymbol{x}_s$ is the locations of the sensors. We start by fixing a small initial $\lambda$ and finding the $q$ that minimizes $h(q, \lambda)$. If $\left\| E_{\pi_z}[q(x)] - \sum_{j=1}^{N_q} \alpha_j x^j \right\| < \tau$ for some tolerance $\tau > 0$, then we accept $q$ as the solution of Problem 5.9. Otherwise, we increase the value of $\lambda$ and start again. To minimize (5.18), we take advantage of being able to take derivatives with respect to the hyper-parameters of the density functions using *simultaneous perturbation stochastic approximation* discussed in [25]. This method has the advantage of only requiring two evaluations of (5.18) per iteration, where as other methods would require significantly more.

## 5.3 Summary of Determining the MCAR Prior

Before moving to a numerical example, we want to finish this chapter with a summary of the steps required to calculate the MCAR prior. There are three main steps starting with an expert supplied $\pi_0$.

1. Take the observational data of $u(x)$ and form a polynomial regression $\hat{u}(x)$ that is the best linear approximation of $u(x)$. The procedure calculates the ordinary least squares approximation which was discussed in Section 4.2.

2. Next, we use $\hat{u}(x)$ to generate a recovery requirement on $E_{\pi_z^{\hat{u}}}[q]$ by (5.9). This was another least squares problem.

3. Finally, find a solution to Problem 5.9 using the penalty method on the auxiliary function (5.18).

The result is the approximately recovering prior which is closest, in the KL divergence sense, to our original prior.

# Chapter 6

# Numerical Results

We move to applying the idea of an MCAR prior to a distributed parameter estimation problem with the elliptic equation.

## 6.1 Problem Formulation

One way model the steady state of the heat distribution along a one dimensional rod subject to an internal heat source is Poisson's equation of

$$-\frac{\mathrm{d}}{\mathrm{d}x}\left(q(x)\frac{\mathrm{d}}{\mathrm{d}x}u(x)\right) = f(x) \quad \text{On } D \tag{6.1}$$

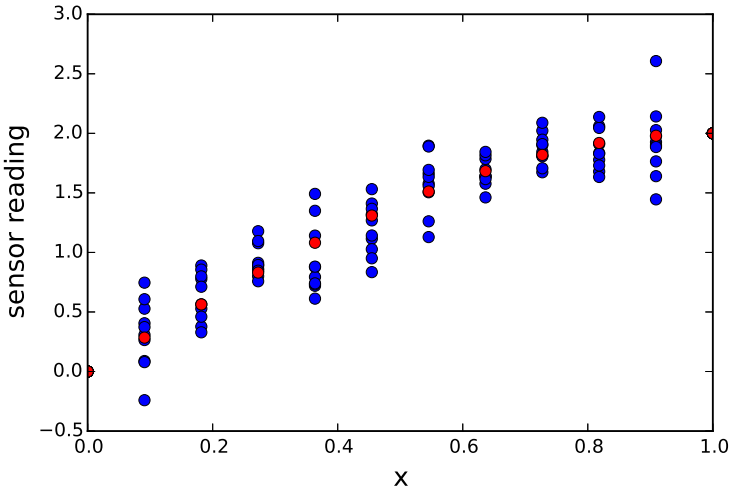
which we discussed in Chapter 4. Start by setting up $N_s$ number of sensors along the rod and then take $N_e$ number of readings from those sensors. In order to have complete knowledge of the true values, we will numerically generate this data instead of collecting it from an experimental set up.

For the rest of this chapter, we consider the following example:

$$D = [0, 1], \tag{6.2}$$
$$u(0) = 0, \tag{6.3}$$
$$u(1) = 2, \tag{6.4}$$
$$q(x) = 1 + x, \tag{6.5}$$
$$f(x) = \frac{\pi^2}{2}(1+x)\sin\left(\frac{\pi}{2}x\right) - \pi\cos\left(\frac{\pi}{2}x\right). \tag{6.6}$$

The forcing term is chosen so that the solution of (6.1) is $u(x) = 2\sin\left(\frac{\pi}{2}x\right)$. To generate the data, we set $N_s = 10$ which are equally spaced along the interior of the rod and take

Figure 6.1: The experimental observations in blue vs. the true data in red.




$N_e = 10$ readings. Additionally, we assume that each sensors is subject to uncorrelated Gaussian white noise with variance 0.1. To numerically generate the data, first determine the finite difference solution $\mathbf{u} = (u_1, \ldots, u_{N_s})$ of (6.1) at the sensor locations $\{x_1, \ldots, x_{N_s}\}$. Then generate the observation data $\tilde{\mathbf{u}}^{(i)}$ with $i \in \{1, \ldots, N_e\}$ by the process

$$\mathbf{u}^{(j)} = \mathbf{u} + \boldsymbol{\xi}^{(j)} \tag{6.7}$$

for $\boldsymbol{\xi}^{(j)} \sim \mathcal{N}(\mathbf{0}_{N_s}, 0.1 I_{N_s \times N_s})$. The generated data is shown in Figure 6.1 and the sample mean is shown in Figure 6.2. Since the draws of the noise distribution are independent, we can define the likelihood function $\rho(q|\mathbf{u})$ by

$$\rho(q|\mathbf{u}) = \prod_{j=1}^{N_e} \rho(q|\mathbf{u}^{(j)}) \tag{6.8}$$

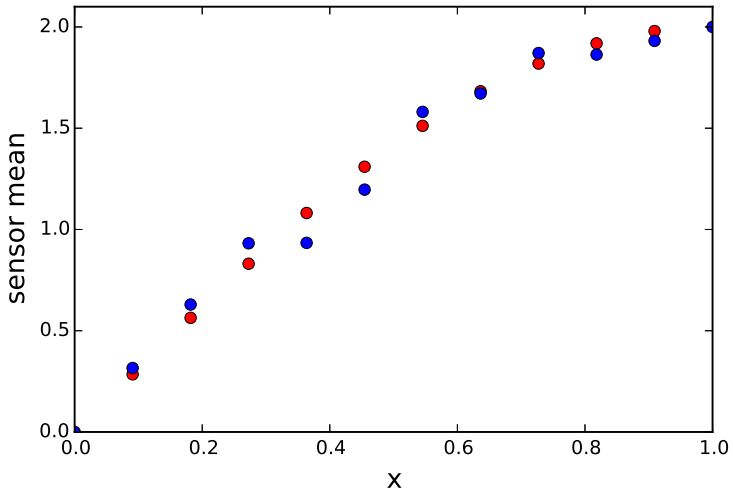
where, by the suggestion of [7],

$$\rho(q|\mathbf{u}^{(j)}) = \frac{1}{(2\pi(0.1)^2)^5} \exp\left(-\frac{\|F(q) - \mathbf{u}^{(j)}\|_2^2}{2(0.1)^2}\right) \tag{6.9}$$

where $F$ is the finite difference solution of (6.1). For our actual calculations, it is more convenient to use the log-likelihood form of (6.8) which is

$$\ln(\rho(q|\mathbf{u})) = -5N_e \ln(2\pi(0.1)^2) + \sum_{j=1}^{\infty} -\frac{\|F(q) - \mathbf{u}^{(j)}\|_2^2}{2(0.1)^2}. \tag{6.10}$$

Figure 6.2: The sample mean of the experimental observations in blue vs. the true data in red.




For the prior, we consider the case of the expert assigning a prior probability distribution of the parameter value at each sensor location as well as the boundary points. Since the elliptic equation requires that $q(x) > 0$ for all $x \in D$, the prior distribution at each sensor point will have to be a gamma distribution which has a probability density function of

$$\gamma(x|k,\theta) = \frac{1}{\Gamma(k)\theta^k} x^{k-1} e^{-x/\theta} \tag{6.11}$$

for some shape parameter $k > 0$ and scale parameter $\theta > 0$. Furthermore, we will consider each sensor's probability distribution independent.

### 6.1.1 Computing Environment

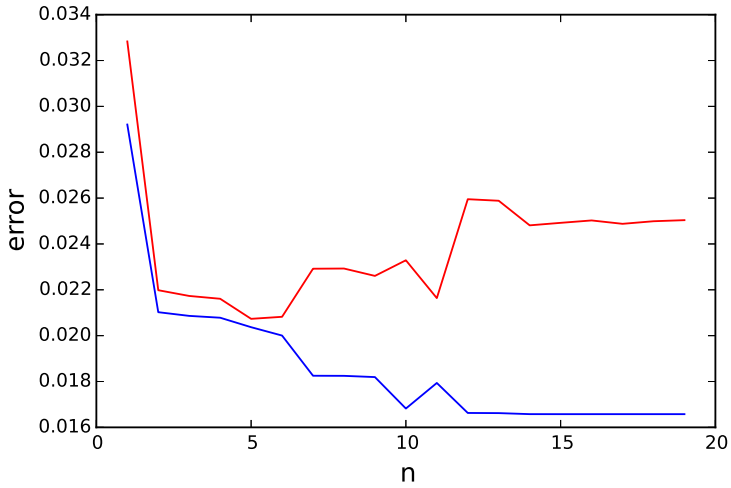
This example was performed with Python 2.7 [26] using the numerical packages NumPy [27] and SciPy [28] as well as the graphics library Matplotlib [29]. To compute the system for the approximate recovery requirement, we use symbolic computation via SymPy [30].

## 6.2 Determining the Recovery Requirement

Before being able to determine the MCAR prior, we must determine the recovery requirement. To accomplish this, we start by turning the observed data set $\mathcal{D} = \left\{\tilde{\mathbf{u}}^{(1)} \ldots, \tilde{\mathbf{u}}^{(N_e)}\right\}$ into an approximation $\hat{u}(x)$ of the true $u(x)$. The approximation is developed by using the

polynomial regression approach discussed in section 4.2. The first step is to determine what degree of polynomial regression to use. This is done by the bias vs. variance procedure. Computing (4.17) and (4.18) for $n = 1, \ldots, 20$, we get the results shown in Figure 6.3. As

Figure 6.3: The training error (4.17) (blue) and the cross-validation error (4.18) (red).




we see from Figure 6.3 that both the training and cross-validation error rapidly decreases until the cross validation error levels off at $n = 5$ and then increases. Examining the higher degree polynomial regression shows this behavior is caused by small oscillations being generated by the higher order terms. To minimize these effects, we set the degree of our regression to be the argmin of cross validation which is $n_u = 5$. The regression is shown in Figure 6.4 and we see that this provides a good approximation of the true value of $u(x)$. Since we need to take a derivative of $\hat{u}(x)$ in applying (5.5), it is good to verify that the derivative of $\hat{u}(x)$ is consistent with the derivative of $u(x)$. The derivative of both functions are shown in Figure 6.5. While the approximation of $\hat{u}'(x)$ is not as accurate to $u'(x)$ as $\hat{u}(x)$ was to $u(x)$, it still provides a good enough approximation for our purposes.

Using $\hat{u}$, we can now develop the recovery requirement. Start by defining a truncated power series

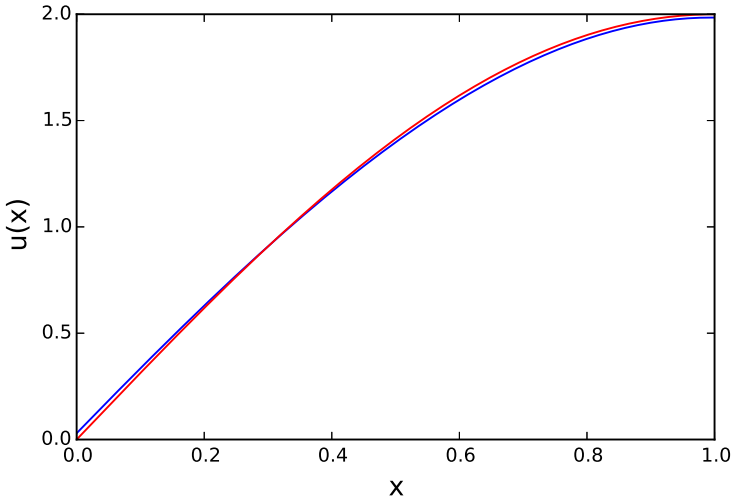
$$E[\hat{q}(x)] = \sum_{n=0}^{N_q} E\left[\hat{Q}_n\right] x^n \tag{6.12}$$

with which we define

$$E[\hat{f}(x)] = -\frac{\mathrm{d}}{\mathrm{d}x}\left(E[\hat{q}(x)]\frac{\mathrm{d}}{\mathrm{d}x}\hat{u}(x)\right). \tag{6.13}$$

Through experimentation, we determined that the best point to truncate at was $N_q = 10$. Since (6.13) is linear with respect to $q$, we know that $E[\hat{f}(x)]$ at any $x \in D$, will result in

Figure 6.4: The approximation $\hat{u}(x)$ (blue) vs. the true $u(x)$ (red).




a linear combination of the $E\left[\hat{Q}_n\right]$. So we take the value of $E[\hat{f}(x)]$ at the sensor points, set each linear combination equal to the values of $f$ at the sensor points, and then solve the resulting linear least squares problem. This results in the recovery requirement function $E[\hat{q}(x)]$ shown in Figure 6.6. True to its name, Figure 6.7 shows that the recovery requirement $E[\hat{q}(x)]$ does recover $f(x)$. Even though $E[\hat{q}(x)]$ does deviate from the true parameter $q(x)$, it is close enough to serve our purpose as a guide to correct the Bayesian prior.

## 6.3 An Inaccurate Prior

For our first prior to correct, let's look at a prior which is biased away from the true parameter. Suppose the expert is confident in his belief that the true parameter is $\tilde{q}(x) = 2 + 2x^2$ and defines the prior density by

$$\pi_0(q) = \prod_{j=1}^{N_s} \gamma(q_j | k_j, \theta_j) \tag{6.14}$$

where $\gamma(\cdot)$ is the gamma probability density function defined by (6.11), $q_j$ is the inputted parameter value at sensor $j$, $\theta_j = 0.25$, and $k_j = \tilde{q}(x_j)/\theta_j$. The choice of $k_j$ is such that $E_{\pi_0}[q(x_j)] = \tilde{q}(x_j)$. The prior expected value of $q$ is shown in Figure 6.8. From the figure, it is clear that the true parameter is statistically insignificant as far as the prior is concerned.

Figure 6.5: The approximation $\hat{u}'(x)$ (blue) vs. the true $u'(x)$ (red).

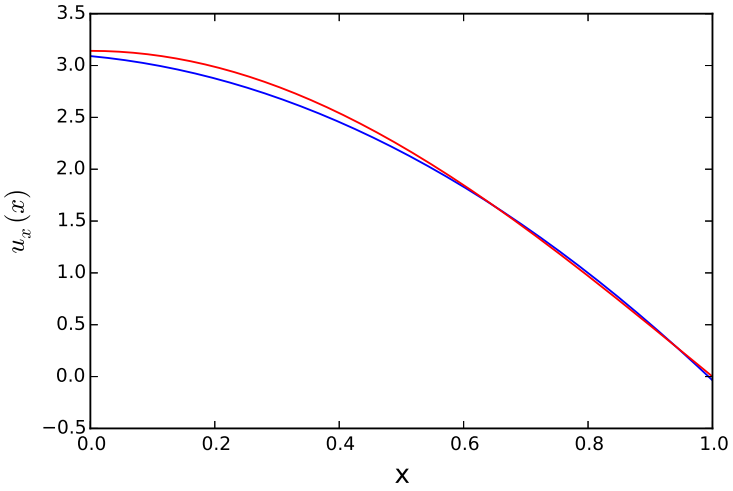
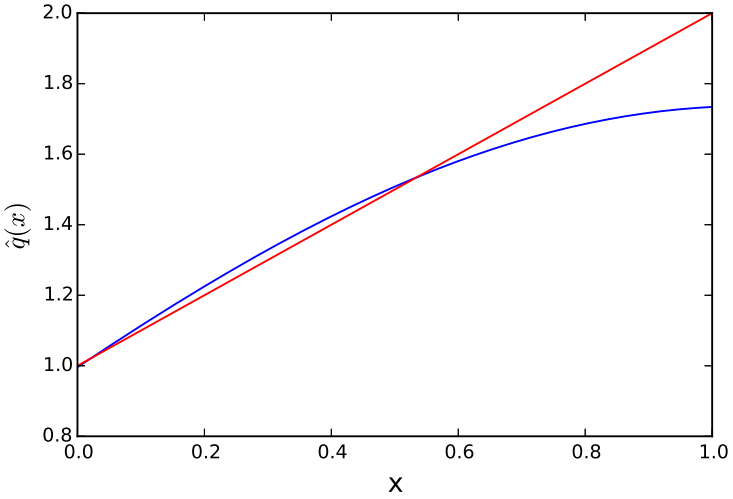


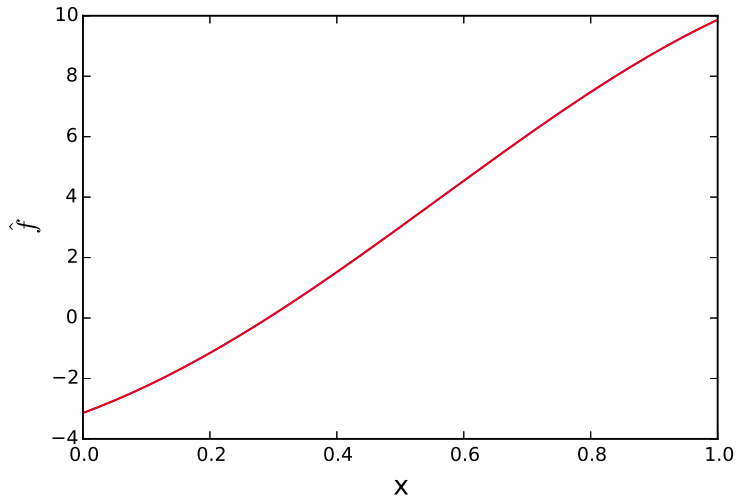

Figure 6.6: The recovery requirement $E[\hat{q}(x)]$ (blue) vs. the true $q(x)$ (red).




Applying Bayes' rule with (6.14) and the likelihood function (6.8), we get a posterior of

$$\pi_z(q) = \frac{1}{C}\left(\prod_{j=1}^{N_e} \rho\left(q|\mathbf{u}^{(j)}\right)\right)\left(\prod_{i=1}^{N_s} \gamma(q_j|k_j,\theta_j)\right) \tag{6.15}$$

Figure 6.7: The $E[\hat{f}(x)]$ (blue) vs. the true $f(x)$ (red).




for some normalizing constant $C$. It is computationally more convenient to use the log probability form of

$$\ln(\pi_z(q)) = C + \sum_{j=1}^{N_e} -\frac{\|F(q) - \mathbf{u}^{(j)}\|_2^2}{2(0.1)^2} + \sum_{j=1}^{N_s} \left[ (k_j - 1)\ln(q_j) - \frac{q_j}{\theta_j} - \ln(\Gamma(k_j)) - k_j \ln(\theta_j) \right] \tag{6.16}$$

for some constant $C$. The graph of the expected value of the posterior is shown in Figure 6.9. From the figure, we see that the expected value of the posterior is significantly different from the true parameter set. Additionally, in Figure 6.10, we see that the expected value is also significantly different from the recovery requirement. We now move on to correcting the prior such that the expected value of the posterior converges to the recovery requirement. As discussed in Section 5.2.3, we will use the simultaneous perturbation stochastic approximation algorithm to determine the minimally corrective, approximately recovering prior. For the cost function, we convert the penalty function (5.18) into

$$J(\boldsymbol{k}, \boldsymbol{\theta}|\lambda) = \ell_{\pi_0^*}(\pi_0(\cdot|\boldsymbol{k}, \boldsymbol{\theta})) + \lambda \left\| E_{\pi_z(\cdot|\boldsymbol{k},\boldsymbol{\theta})}[q(\boldsymbol{x}_s)] - E[\hat{q}(\boldsymbol{x}_s)] \right\|_\infty \tag{6.17}$$

where $\boldsymbol{k}$ and $\boldsymbol{\theta}$ are candidate hyper-parameters for the gamma prior and $\boldsymbol{x}_s$ are the domain locations for the sensors. We start with $\lambda = 0.01$ and run simultaneous perturbation stochastic approximation on (6.17). This process will generate a candidate minimizer $(\boldsymbol{k}^*, \boldsymbol{\theta}^*)$ which we will accept if $\left\| E_{\pi_z(\cdot|\boldsymbol{k}^*,\boldsymbol{\theta}^*)}[q(\boldsymbol{x}_s)] - E[\hat{q}(\boldsymbol{x}_s)] \right\|_\infty < 0.1$. If the candidate minimizer is reject, we set $\lambda = 10 * \lambda$ and run the process again.

After four iterations penalty subproblem, we get a accepted candidate minimizer. The resultant, and original, prior hyperparameters are shown in Table 6.1. Also, the expected value

Figure 6.8: The prior expected value of $q$ (blue) and the true value of $q$ (red).

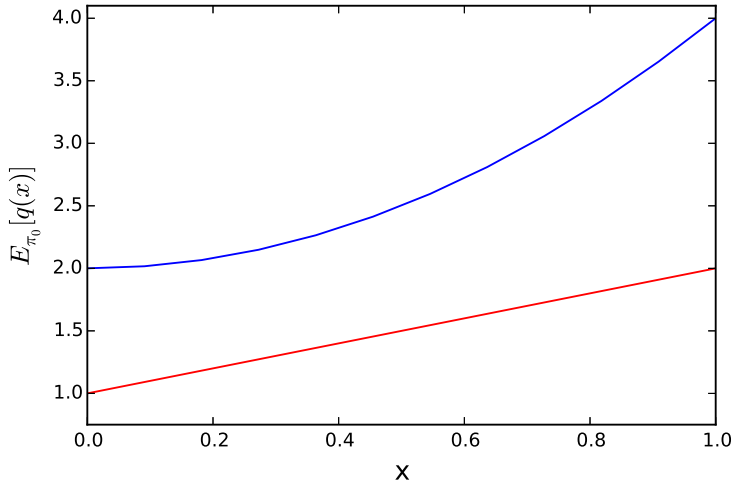
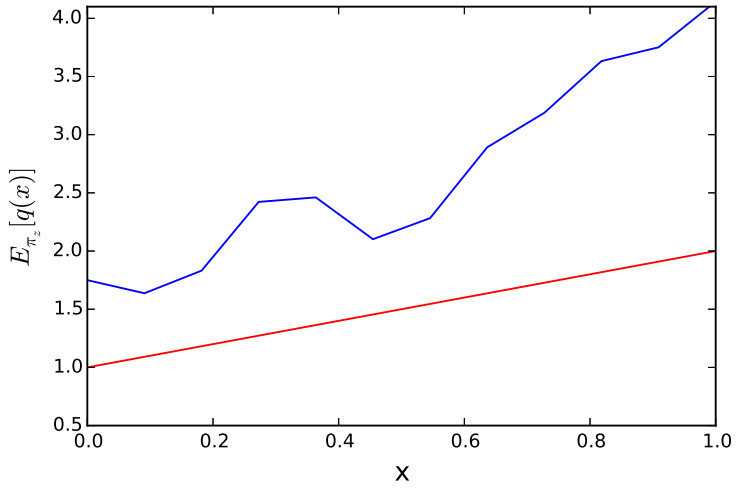



Figure 6.9: The posterior expected value of $q$ (blue) and the true value of $q$ (red).




of the posterior generated by the minimally corrective prior is shown in Figure 6.11. Comparing the corrected posterior expected value to the true parameter, we get the result shown in Figure 6.12 Comparing this result with the original posterior (Figure 6.9), we see that we have made a significant improvement in recovering the original distributed parameter.

Figure 6.10: The posterior expected value of $q$ (blue) and the recovery requirement (green).

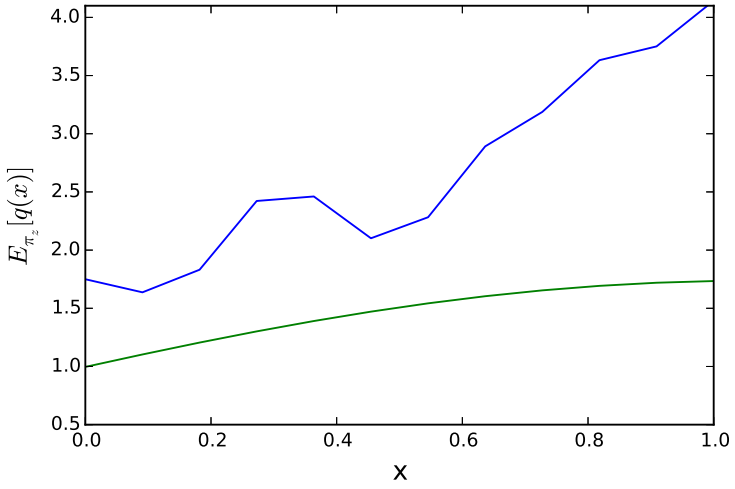



Table 6.1: The original and minimally corrective hyperparameters for the inaccurate prior

| $i$ | $x_i$ | $k_i$ | $\theta_i$ | $k_i^*$ | $\theta_i^*$ |
|---|---|---|---|---|---|
| 0 | 0 | 4 | .25 | 3.4488 | 0.2554 |
| 1 | .0909 | 4.3636 | .25 | 6.3487 | 0.1694 |
| 2 | 0.1818 | 4.7272 | .25 | 5.4072 | 0.2345 |
| 3 | 0.2727 | 5.0909 | .25 | 4.9016 | 0.2545 |
| 4 | 0.3636 | 5.4545 | .25 | 6.0751 | 0.2477 |
| 5 | 0.4545 | 5.8181 | .25 | 6.6934 | 0.2383 |
| 6 | 0.5454 | 6.1818 | .25 | 7.1685 | 0.2211 |
| 7 | 0.6363 | 6.5454 | .25 | 5.9860 | 0.2592 |
| 8 | 0.7272 | 6.9090 | .25 | 6.7440 | 0.2333 |
| 9 | 0.8181 | 7.2727 | .25 | 7.5633 | 0.2320 |
| 10 | 0.9090 | 7.6363 | .25 | 7.1842 | 0.2349 |
| 11 | 1 | 8 | .25 | 8.8121 | 0.1940 |

## 6.4 An Accurate Prior

After seeing how the developing an MCAR prior can significantly reduce the error in incorrect expert judgement, we want to see how developing an MCAR prior will affect correct expert judgement. So in this case, lets suppose the expert suspects the true parameter is $\tilde{q}(x) = 1+x$ and is feeling confident in this suspicion. Again, they encode this judgement into a prior of the form (6.14), except this time the hyper parameters are $\theta_j = 0.25$ and $k_j = (1 + x_j)/\theta_j$. This makes it so that $E_{\pi_0}[q(x_j)] = 1 + x_j$ which is the correct parameter values. Applying

Figure 6.11: The corrected posterior expected value of $q$ (blue) and the recovery requirement (green).

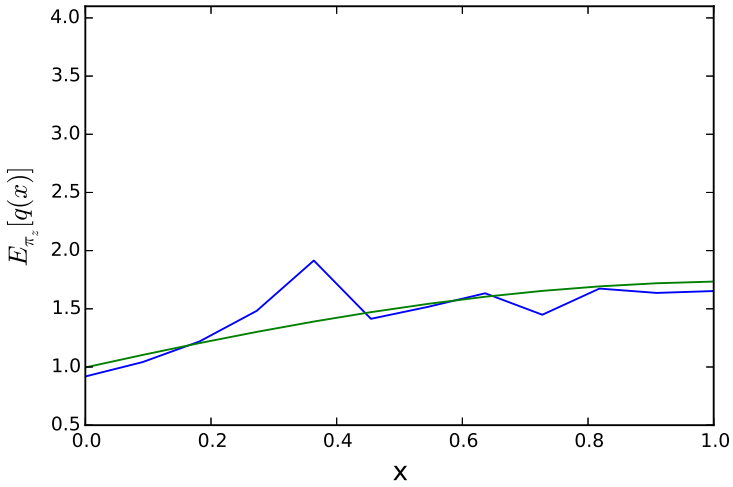
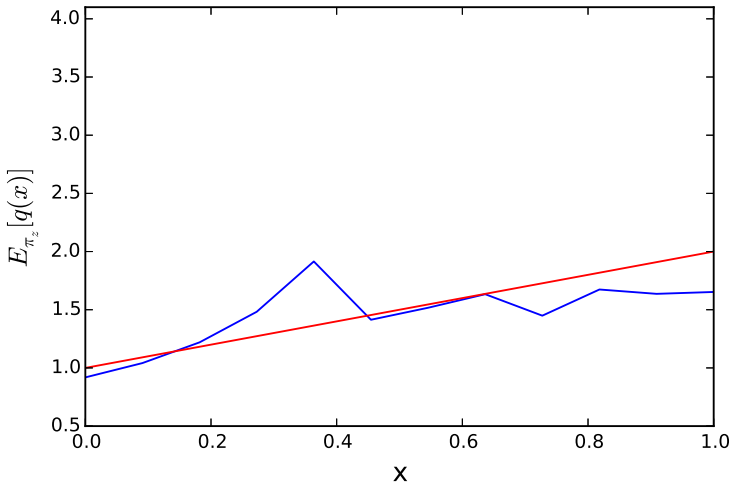



Figure 6.12: The corrected posterior expected value of $q$ (blue) and the true parameter (red).




Bayes' rule, we get a posterior of the form of (6.15). The posterior of this distribution with the true parameter set is shown in Figure 6.13 and with the recovering parameter set is shown in Figure 6.14. This time, the posterior is already relatively accurate to the true parameter. The current errors are mainly from the limited amount of observational data we have available. We use the same penalty functions as (6.17) and the same procedure with the new initial prior hyperparameters.

Figure 6.13: The posterior expected value of $q$ (blue) and the true value of $q$ (red).

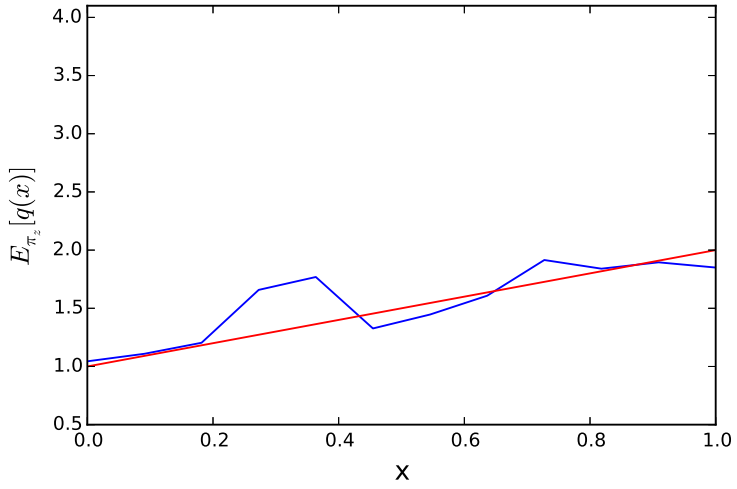
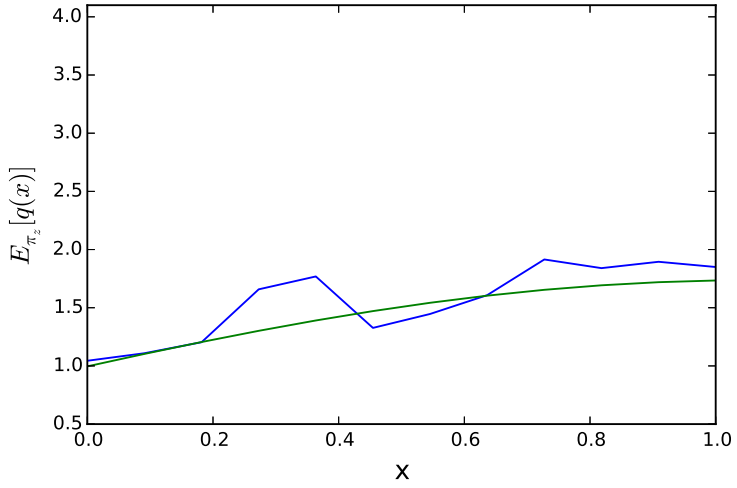



Figure 6.14: The posterior expected value of $q$ (blue) and the recovery requirement of $q$ (green).

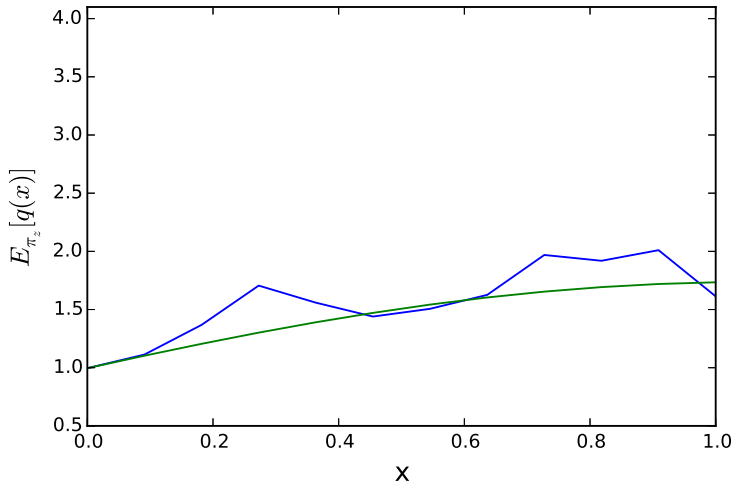



After three iterations penalty subproblem, we get a accepted candidate minimizer. The resultant and original prior hyper-parameters are shown in Table 6.2. Also, the expected value of the posterior generated by the minimally corrective prior is shown in Figure 6.15. Comparing the corrected posterior expected value to the true parameter, we get the result shown in Figure 6.16. In this case, we see that the expected value of the MCAR posterior has more bias in it from the posterior generated by the original expert judgement. However,

Table 6.2: The original and minimally corrective hyperparameters for the accurate prior

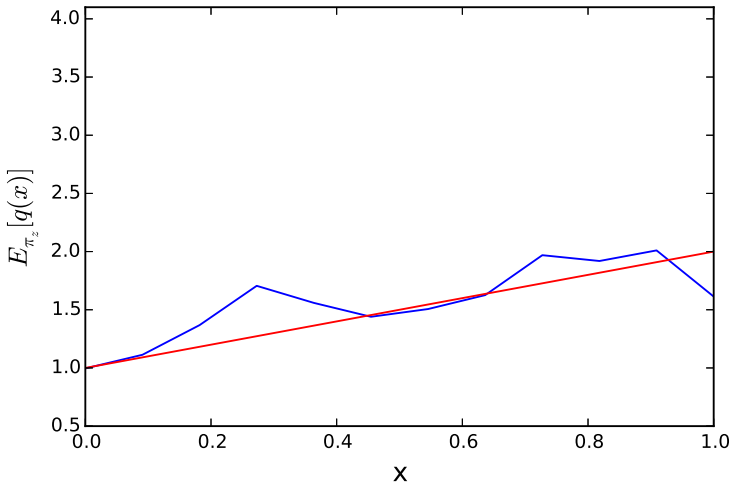
| $i$ | $x_i$ | $k_i$ | $\theta_i$ | $k_i^*$ | $\theta_i^*$ |
|---|---|---|---|---|---|
| 0 | 0 | 8 | .25 | 5.3816 | 0.2030 |
| 1 | .0909 | 8.066 | .25 | 8.0288 | 0.1473 |
| 2 | 0.1818 | 8.2644 | .25 | 9.0241 | 0.1418 |
| 3 | 0.2727 | 8.5950 | .25 | 8.4224 | 0.1595 |
| 4 | 0.3636 | 9.0578 | .25 | 8.0129 | 0.1751 |
| 5 | 0.4545 | 9.6528 | .25 | 8.1657 | 0.1921 |
| 6 | 0.5454 | 10.3801 | .25 | 8.0561 | 0.2031 |
| 7 | 0.6363 | 11.2396 | .25 | 7.4337 | 0.2281 |
| 8 | 0.7272 | 12.2314 | .25 | 11.2298 | 0.1539 |
| 9 | 0.8181 | 13.3553 | .25 | 11.4420 | 0.1603 |
| 10 | 0.9090 | 14.6115 | .25 | 11.0652 | 0.1627 |
| 11 | 1 | 16 | .25 | 14.3141 | 0.1269 |

Figure 6.15: The corrected posterior expected value of $q$ (blue) and the recovery requirement (green).




this is a minor bias compared with the bias introduced by the first expert. So the benefit of correcting bad expert judgement out weighs the newly introduced error.

Figure 6.16: The corrected posterior expected value of $q$ (blue) and the true parameter (red).



# Chapter 7

# Conclusions

We considered the problem of reducing expert bias in Bayesian parameter estimation. In contrast to the current approaches that focus on restricting prior distributions to robust classes, we presented a novel approach that focuses on correcting expert bias. This approach begins by developing a guide for the parameter set called the recovery requirement using polynomial regression. Then we correct the prior using this guide. The MCAR approach was applied to an example where we estimate the distributed parameter in an elliptic equation. For this example, we considered one case with significant expert bias and a second case with no expert bias. In the case of significant expert bias, this method resulted in a prior that caused the posterior distribution that produces the true parameter with increased statistical significance. In the case of no expert bias, the prior was shifted in a way that reduced the statistical significance of the true parameter; however, this shift and the associated reduction in statistical significance was minor. Since we wouldn't know the true parameter set in practice, we are willing to accept the introduction of small errors from this method for the benefit of correcting large errors from expert bias.

These results are promising and further work is required to fully develop the method. Some questions we hope to answer in the future are:

1. In the context of this method, is there a better choice of a basis for representing the stochastic distributed parameter process instead of using the basis generated by the Karhunen-Loève theorem? If not, can we prove optimality?

2. One significant, computational bottle neck is that an evaluation of (5.18) requires a Monte Carlo integration. Is there a way to eliminate this Monte Carlo step? Applying a method as described in [31] has potential.

3. For differential operators that are nonlinear with respect to the parameter, is there any relationship between the expected value of the recovering prior and the parameter set as we found in Lemma 5.2?

# Chapter 8

# Bibliography


[1] Ralph C. Smith. *Uncertainty Qunatification: Theory, Implementation, and Applications*. SIAM, 2014.

[2] Benard C. Levy. *Principles of Signal Detection and Parameter Estimation*. Springer, first edition, 2008.

[3] A. M. Stuart. Inverse problems: A Bayesian perspective. *Acta Numerica*, 19:451–559, 2010.

[4] David Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[5] Hanns L. Harney. *Bayesian Inference: Parameter Estimation and Decisions*. Advanced Texts in Physics. Springer, first edition, 2003.

[6] Moritz Allmaras, Wolfgang Bengerth, Jean Marie Linhart, Javier Polanco, Fang Wang, Kainan Wang, Jennifer Webster, and Sarah Zedler. Estimating parameters in physcial models though bayesian inversion: A complete example. *SIAM Review*, 55(1):149–167, 2013.

[7] Jingbo Wang and Nicholas Zabaras. A Bayesian inference approach to the inverse heat conduction problem. *International Journal of Heat and Mass Transfer*, 47:3927–3941, 2004.

[8] David Rios Insua and Fabrizio Ruggeri, editors. *Robust Bayesian Analysis*, volume 152 of *Lecture Notes in Statistics*. Springer, first edition, 2000.

[9] Jairo A. Fúquene, John D. Cook, and Luis R. Pericchi. A case for robust Bayesian priors with applications to clinical trials. *Bayesian Analysis*, 4(4):817–846, 2009.

[10] Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. Texts in Statistical Science. CRC Press, third edition, 2013.

[11] Youssef M. Marzouk, Habib N. Najm, and Larry A. Rahn. Stochastic spectral methods for efficient Bayeisan solutions of inverse problems. *Journal of Computational Physics*, 244:560–586, 2007.

[12] Xiang Ma and Nicholas Zabaras. An efficient Bayesian inference approach to inverse problems based on an adaptive sparse grid collocation method. *Inverse Problems*, 25, 2009.

[13] Persi Diaconis. The Markov Chain Monte Carlo revolution. *Bulletin of the American Mathematical Society*, 26(2):179–205, 2009.

[14] Dani Gamerman and Hedibert F. Lopes. *Markov Chain Monte Carlo: Stochastic Simulations for Bayesian Inference*. Texts in Statistical Science. Chapman & Hall, second edition, 2006.

[15] Lawrence C. Evans. *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, Rhode Island, second edition, 1998.

[16] Karl Kunisch. Inherent identifiabillity of parameter in elliptic differential equations. *Journal of Mathematical Analysis and Applications*, 132(2):453–472, 1986.

[17] Carmen Chicone and Jürgen Gerlach. A note on the identifiability of distributed parameters in elliptic equations. *SIAM Journal on Mathematical Analysis*, 18(5):1378–1384, 1987.

[18] H. Thomas Banks and Karl Kunisch. *Estimation Techniques for Distributed Parameter Systems*. Systems & Control: Foundations & Applications. Birkhäuser, first edition, 1989.

[19] J.W. Thomas. *Numerical Partial Differential Equations: Finite Difference Methods*, volume 22 of *Texts in Applied Mathematics*. Springer, 1998.

[20] M. Loève. *Probability Theory II*, volume 46 of *Graduate Texts in Mathematics*. Springer, fourth edition, 1994.

[21] Norman R. Draper and Harry Smith. *Applied Regression Analysis*. Wiley Series in Probability and Statistics. Wiley-Interscience, 1998.

[22] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer Series in Statistics. Springer, second edition, 2011.

[23] Lloyd N. Trefethen and David Bau III. *Numerical Linear Algebra*. SIAM, 1997.

[24] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, second edition, 2006.

[25] James C. Spall. *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. Wiley-Intercience Series in Discrete Mathematics and Optimization. Wiley, 2003.

[26] Guido Rossum. Python reference manual. Technical report, Amsterdam, The Netherlands, 1995.

[27] Stéfan van der Walt, S. Chris Colbert, and Gaël Varoquaux. The NumPy array: A structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

[28] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001.

[29] John D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.

[30] SymPy Development Team. *SymPy: Python library for symbolic mathematics*, 2014.

[31] T. A. Moselhy and Y. M. Marzouk. Bayesian inference with optimal maps. *Journal of Computational Physics*, 231(23):7815–7850, 2012.