Bayesian Methods for Mineral Processing Operations

Scott Carl Koermer

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

Doctor of Philosophy
in
Mining Engineering

Christopher A. Noble, Chair
Robert B. Gramacy
Nino S. Ripepi
Emily A. Sarver
Paul F. Ziemkiewicz

April 26, 2022
Blacksburg, Virginia

Keywords: Process Engineering, Uncertainty Quantification,
Bayesian Inference, Gaussian Process Regression

Bayesian Methods for Mineral Processing Operations

Scott Carl Koermer

(ABSTRACT)

Increases in demand have driven the development of complex processing technology for separating mineral resources from exceedingly low grade multi-component resources. Low mineral concentrations and variable feedstocks can make separating signal from noise difficult, while high process complexity and the multi-component nature of a feedstock can make testwork, optimization, and process simulation difficult or infeasible. A prime example of such a scenario is the recovery and separation of rare earth elements (REEs) and other critical minerals from acid mine drainage (AMD) using a solvent extraction (SX) process. In this process the REE concentration found in an AMD source can vary site to site, and season to season. SX processes take a non-trivial amount of time to reach steady state. The separation of numerous individual elements from gangue metals is a high-dimensional problem, and SX simulators can have a prohibitive computation time. Bayesian statistical methods intrinsically quantify uncertainty of model parameters and predictions given a set of data and a prior distribution and model parameter prior distributions. The uncertainty quantification possible with Bayesian methods lend well to statistical simulation, model selection, and sensitivity analysis. Moreover, Bayesian models utilizing Gaussian Process priors can be used for active learning tasks which allow for prediction, optimization, and simulator calibration while reducing data requirements. However, literature on Bayesian methods applied to separations engineering is sparse. The goal of this dissertation is to investigate, illustrate, and test the use of a handful of Bayesian methods applied to process engineering problems. First further details for the background and motivation are provided in the introduction. The literature review provides further information regarding critical minerals, solvent extraction, Bayeisan inference, data reconciliation for separations, and Gaussian process modeling. The body of work contains four chapters containing a mixture of novel applications for Bayesian methods and a novel statistical method derived for the use with the motivating problem. Chapter topics include Bayesian data reconciliation for processes, Bayesian inference for a model intended to aid engineers in deciding if a process has reached steady state, Bayesian optimization of a process with unknown dynamics, and a novel active learning criteria for reducing the computation time required for the Bayesian calibration of simulations to real data. In closing, the utility of a *handfull* of Bayesian methods are displayed. However, the work presented is not intended to be *complete* and suggestions for further improvements to the application of Bayesian methods to separations are provided.

Bayesian Methods for Mineral Processing Operations

Scott Carl Koermer

(GENERAL AUDIENCE ABSTRACT)

Rare earth elements (REEs) are a set of elements used in the manufacture of supplies used in green technologies and defense. Demand for REEs has prompted the development of technology for recovering REEs from unconventional resources. One unconventional resource for REEs under investigation is acid mine drainage (AMD) produced from the exposure of certain geologic strata as part of coal mining. REE concentrations found in AMD are significant, although low compared to REE ore, and can vary from site to site and season to season. Solvent extraction (SX) processes are commonly utilized to concentrate and separate REEs from contaminants using the differing solubilities of specific elements in water and oil based liquid solutions.

The complexity and variability in the processes used to concentrate REEs from AMD with SX motivates the use of modern statistical and machine learning based approaches for filtering noise, uncertainty quantification, and design of experiments for testwork, in order to find the truth and make accurate process performance comparisons. Bayesian statistical methods intrinsically quantify uncertainty. Bayesian methods can be used to quantify uncertainty for predictions as well as select which model better explains a data set. The uncertainty quantification available with Bayesian models can be used for decision making. As a particular example, the uncertainty quantification provided by Gaussian process regression lends well to finding what experiments to conduct, given an already obtained data set, to improve prediction accuracy or to find an optimum. However, literature is sparse for Bayesian statistical methods applied to separation processes.

The goal of this dissertation is to investigate, illustrate, and test the use of a handful of Bayesian methods applied to process engineering problems. First further details for the background and motivation are provided in the introduction. The literature review provides further information regarding critical minerals, solvent extraction, Bayeisan inference, data reconciliation for separations, and Gaussian process modeling. The body of work contains four chapters containing a mixture of novel applications for Bayesian methods and a novel statistical method derived for the use with the motivating problem. Chapter topics include Bayesian data reconciliation for processes, Bayesian inference for a model intended to aid engineers in deciding if a process has reached steady state, Bayesian optimization of a process with unknown dynamics, and a novel active learning criteria for reducing the computation time required for the Bayesian calibration of simulations to real data. In closing, the utility of a *handfull* of Bayesian methods are displayed. However, the work presented is not intended to be *complete* and suggestions for further improvements to the application of Bayesian methods to separations are provided.

I am mostly here because of both luck and an understanding of when I am lucky. I am lucky to have supportive parents, lucky to have stumbled upon extremely supportive faculty in the various accademic programs at Virginia Tech, and lucky that those I am closest with can put up with me during times of high stress.

Acknowledgements

This dissertation was not completed in isolation, and I have many individuals who made this step possible to acknowledge.

I would like to thank my parents. I am lucky they value higher education and have been supportive of my choice to return to school.

A special thanks to my PhD advisor Dr. Aaron Noble for not just taking me as a graduate student, but also providing me the freedom and creativity to find my own ways to tackle a problem.

Additionally a special thanks to all the Mining engineering department staff that work behind the scenes.

I would like to thank Dr. Robert Gramacy for producing an excellent text and course for applied surogate modeling and practically taking me on as an additional graduate student pro-bono.

I would like to thank undergraduate researchers Kelsey Popolizio and Bryce Burmaster for aiding in laboratory experiments and sample preparation. The real data used in this publication would have been significantly more difficult to obtain without their help.

Thank you Inga Haugen for converting the final document to PDF/A-2b, finishing this dissertation would not have been possible without you.

Thank you Rachel for putting up with the chaos and uncertainty over the last 8 months, particularly the month of April 2022.

I would like to acknowledge NETL for providing the funding for the projects which sponsored this work.

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Federal Award Numbers: DE-FE0026927 and DE-FE0031834

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Humans have used more minerals from 1900-present than all of history prior to 1900 (Hartman and Mutmansky 2002). High resource demand driving technological advancement has led to decreases in average mined ore grade over time (Rötzer and Schmidt 2018; Slade 1982; Henckens et al. 2016). The *mineralogical barrier* is the grade, or purity, at which an ore is not reasonable to be mined due to the increase in energy requirements (Skinner 1976).

When incorporating technological advances, many mining companies take the approach of waiting for a new technology to be developed and proven by an equipment vendor before adoption (Bartos 2007), with *proof* implying overwhelming certainty, or lack of uncertainty, that a development is advantageous. As the mineralogical barrier decreases, smaller mineral quantities in the feedstock to a concentration process must be measured for technological performance calculations. When sampling a feedstock sample heterogeneity and particle heterogeneity lead to sampling error. Sample preparation error as well as analytical error further contribute to the overall sample estimation error (Pitard 1993). Uncertainty related to the measurement of small concentrations and variability in the feedstock can lead to further uncertainty in a proposed innovation.

Modeling and the collection of experimental data for process concentrating a low grade multi-component ore is necessary for feasibility evaluations, but difficult. Testing n levels for p process parameters requires n^p tests to observe every parameter combination, quickly becoming infeasible for small values of n and p. For a complicated model, uncertainty related to model parameter estimates generated from a modest data set translates to prediction uncertainty. In the best case, process uncertainty can simply lead to slightly off predictions, and in the worst case a technological innovation can be overlooked due to a false negative error.

As mineral processing technology becomes more complex, a necessity for understanding of the uncertainty of test results for decision making purposes arises. Tools for exploring and optimizing a complex, yet not fully understood, process are advantageous for reducing the time between conceptualization by a researcher and obtaining the proof required by a mining company for adoption.

1.2 Motivation

Rare Earth Elements (REEs; see §2.1) are a subset of 15 elements with increasing demand and supply issues, which has lead to a multitude of governments to declare REEs as *critical minerals* (Nassar and Fortier 2021; Canada 2021; Skirrow et al. 2013; The European Commission 2020). Researchers have been creative in finding unconventional resources for REEs (Honaker et al. 2017; Q. Huang et al. 2018; Ziemkiewicz et al. 2016). Unconventional resources for REEs often have low concentrations. Of particular relevance a survey of the unconventional resources of acid mine drainage (AMD), and the precipitate formed from the neutralization of acid mine drainage (AMDp) found total rare earth element (TREE) concentrations as low as 8 μg/L and 40 g/ton respectively (Vass, Noble, and Ziemkiewicz 2019). For comparison, the more conventional carbonatite bastnaesite found at the Mountain Pass mine in California has a TREE content of approximately 7.98% by mass (Hellman and Duncan 2014).

Processes in development for unconventional REE resources are a textbook example of demand leading to technology which reduces the mineralogical barrier. The projects which motivated the methods presented in this dissertation are the recovery of REEs from acid mine drainage (AMD) and acid mine drainage precipitate (AMDp). Processing AMD and AMDp for REE recovery requires an accounting of large amounts of variability. Such a process would have to take into account the variability in elemental concentrations between regions, between sites (Saber 2018), and between seasons. There is even the potential for individual weather events to effect concentrations found within an individual AMD source. Accounting for such a variable feedstock is paramount for analysis of what process methodologies are environmentally, technically, and economically feasible.

The basic flow for the process used in concentrating REEs from AMD and producing a finished REE product is shown in Figure 1.1 (Ziemkiewicz, Noble, and Vass U.S. Patent 10 954 582, Feb. 19, 2020). First a pre-concentrate is precipitated at an AMD site. This pre-concentrate is then shipped to a central facility where a leachate is made for solvent extraction processing and separation of REEs from gangue metals. After downstream refining of intermediate REE products produced from SX, a finished REE concentrate is produced which can be used in manufacturing.

For the steps in Figure 1.1, there is the potential for variability in initial stages to cascade through the process and effect overall technical, economic, and environmental feasibility. Each stage in the process has a significant degree of complexity. As part of process development, a bench scale SX pilot plant, shown in Figure 1.2, has been built at West Virginia University (WVU) to learn

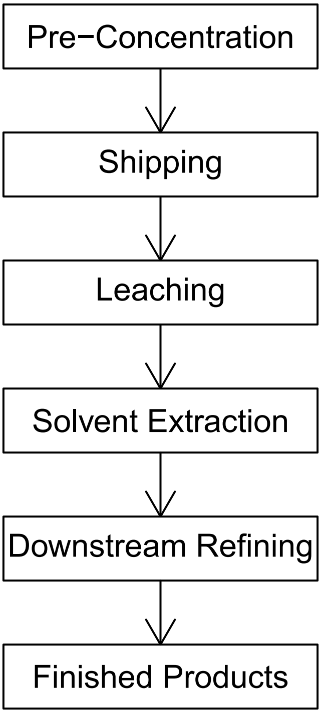

Figure 1.1: Process flow chart for the concentration of REEs from acid mine drainage.

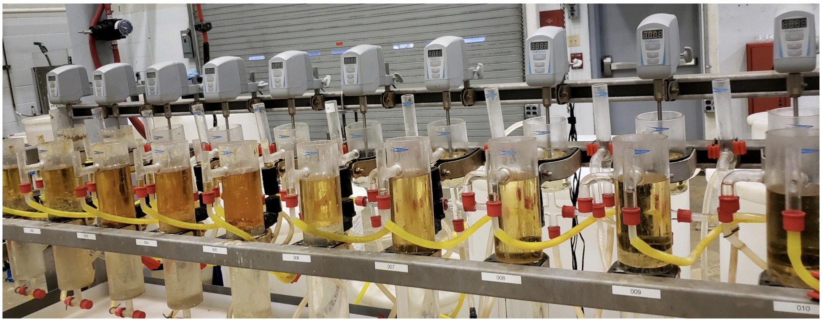

Figure 1.2: Bench scale SX system at WVU.

about the SX process for this particular application. In one test, the SX plant was run for 43 hours with a constant feedstock and operational parameters. Figure 1.3 and 1.4 show elemental concentrations, plotted hourly, from two sampling locations for aqueous flows exiting the process.

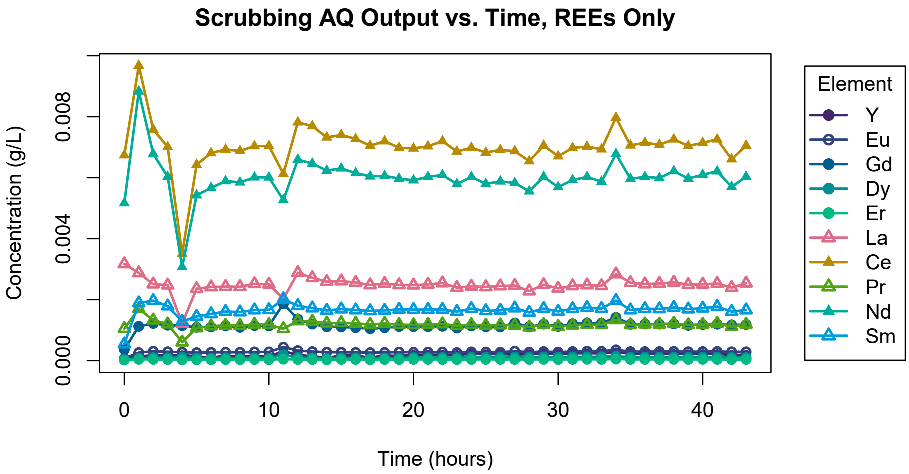

Figure 1.3: Plot of REE concentrations found in AQ output of scrubbing stage

The purpose of the *scrubbing* stage in a REE SX operation is to remove impurities from the organic phase uwithout removing REEs by mixing the organic phase with a moderately acidic solution. A dialed-in scrubbing stage will therefore improve the puritiy of a downstream REE concentrate. Therefore, the aqueous REE concentrations of the scrubbing stage, shown in Figure 1.3, should ideally be low. Notably, Figure 1.3 shows seemingly random perturbations of the low REE concentrations, although feedstock and process parameters are constant. The perturbations could be truly there, or could be related to analytical error. The ability to take into account the variability of the low REE concentrations observed exiting the scrubbing stage is necessary to ensure REEs are not being lost as part of the process.

Figure 1.4 shows the concentrations of total rare earth elements (TREE), heavy rare earth elements (HREE), and light rare earth elements (LREE; see §2.1) exiting the stripping stage, sampled simultaneous to the concentrations in Figure 1.3. The stripping stage is meant to remove REEs from the organic phase using a strong acid. The aqueous output of the stripping stage is sent for further processing and elemental separations, and therefore should have a relatively high concentration of REEs. The trend for the TREE concentrations in Figure 1.4 is unclear. Possibly, the process has not reached steady state after 43 hours of run time. For a process that can take so long to reach steady state, a metric for estimation of if the process is at steady state, and steady state is useful for making process performance comparisons.

The variability, low concentrations, and complexity of the SX process for concentrating REEs from AMD necessitate uncertainty quantification in order to

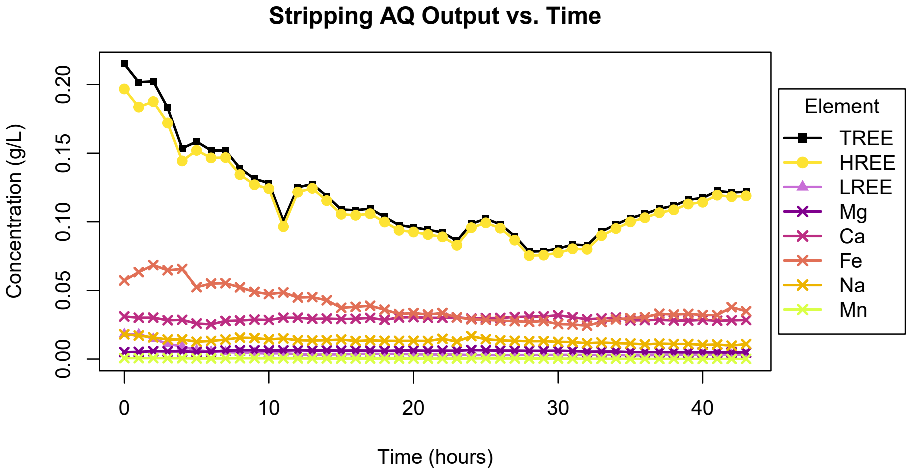

Figure 1.4: Plot of concentrations found in AQ output of stripping stage of a bench scale SX plant.

better understand and optimize these processes. Bayesian statistical methods leverage Bayes' theorem (§2.2, Equation (2.1)) to provide uncertainty quantification when inferring the parameters of a model, which can in turn provide uncertainty quantification of a predictions. Surrogate modeling with Bayesian models (§5, §6) can provide probabilistic methods for data collection, which reduce the amount of real data required for the same level of accuracy, through active learning.

1.3 Objectives

The goal of this dissertation is to demonstrate how utilizing Bayesian methods for separation processes can provide a framework for improving engineering decisions when process data is highly variable, scarce, and expensive. The motivating problem of recovering REEs from the low and variable concentrations found in AMD serves as a case study. The methods which follow can be applied to general separation problems.

Explicit research objectives are stated below.

1. Explore the potential for Bayesian methods to be utilzed in process engineering problems for the purposes of:
   a. Uncertainty quantification.
   b. Reduction in data requirements.
   c. Improvements in modeling accuracy and model parameter estimation.
2. Adapt Bayesian statistical methodologies commonly used in other industries for separations engineering applications.
3. Derive novel statistical methods motivated by the unique problems encountered in REE recovery.
4. Present methods by:

a. either testing on real data or illustrating how methods can be applied using simulated data when real data is not available.
b. providing resources, including ample citations and code, so that other process engineers can adapt the methods explored for their own purposes.

The dissertation first reviews relevant literature (§2), providing further details on REEs and the motivating problem (§2.1), the solvent extraction (SX) methods used to concentrate and separate REEs (§2.4), typical SX modeling techniques (§2.4.1), process mass balancing (§2.3), Bayesian inference (§2.2), and Gaussian Process (GP) surrogate methods (§2.5). The main body of work is broken up into four manuscripts of work related to Bayesian methods for mass balancing (§3), Bayesian estimation of steady state and steady state conditions (§4), Bayesian optimization of an unknown process (§5), and Bayesian calibration of a simulated process to provide predictions of a real process (§6). Concluding remarks (§3.6) summarize the methods presented and their implications for process analysis, as well as propose future work in the area of Bayesian Methods for separation processes.

Chapter 2

Literature Review

2.1 Rare Earth Elements

Rare Earth Elements (REEs) include elements from the lanthanide series, Yttrium, and Scandium (Gosen et al. 2014). Chemical symbol, atomic number, and crustal abundance is shown in Table 2.1. REEs are grouped into *heavy* and *light* categories based on their chemical properties. The term *Rare Earth Elements* is somewhat misleading, as many REEs are found in similar abundance as common metals such as copper and lead (Gosen et al. 2014), but are less often found at high concentrations (USGS 2021).

Uses for REEs include making glass, lighting, metallurgy, catalysts, batteries, ceramics, and magnets (Goonan 2011). A significant portion of REE usage is in *high growth* green technologies, such as battery alloys and magnets, for which usage grows faster than the economy (Goonan 2011; Balaram 2019).

Rare earth elements are listed as critical minerals, minerals which have a supply risk and are economically important, by the United States (Nassar and Fortier 2021), Canadian (Canada 2021), Australian (Skirrow et al. 2013), and European Union (The European Commission 2020) governments. The United States has approximately a 100% import reliance for REEs from China, Estonia, Japan, and Malaysia (USGS 2021). China governs the REE marketplace with a known production of greater than 50% of global production and 37% of global reserve estimates (USGS 2021). China has even further dominance over REE processing, accounting for almost 90% of the global supply of REEs processed (IEA 2021). Comparatively, the US accounts for approximately 15% of global production and 1.2% of global reserve estimates (USGS 2021).

Finding additional REE resources is critical to economic growth and shifting towards more sustainable technologies. In the search for additional REE resources, previously unconsidered low grade feed stocks are being explored for process feasibility and economic viability. Recent advances include recovering REEs from coal and coal byproducts (Honaker et al. 2017; Q. Huang et al.

Table 2.1: Rare earth elements and their crustal abundance (Lide 2004).

Element	Symbol	Atomic Number	Crustal Abundance (ppm)
Scandium	Sc	21	22.00
Light REEs			
Lanthanum	La	57	39.00
Cerium	Ce	58	66.50
Praseodymium	Pr	59	9.20
Neodymium	Nd	60	41.50
Samarium	Sm	62	7.05
Europium	Eu	63	2.00
Galodinium	Gd	64	6.20
Heavy REEs			
Terbium	Tb	65	1.20
Dysprosium	Dy	66	5.20
Holmium	Ho	67	1.30
Erbium	Er	68	3.50
Thulium	Tm	69	0.52
Ytterbium	Yb	70	3.20
Lutetium	Lu	71	0.80
Yttrium	Yb	39	33.00

2018), coal mine drainage (Ziemkiewicz et al. 2016), and coal mine drainage treatment products (Vass, Noble, and Ziemkiewicz 2019).

REEs are typically separated from one another using solvent extraction (§2.4), and the separation of individual REEs is regarded as one of the more difficult types of separations in hydrometallurgy (J. Zhang, Zhao, and Schreiner 2016). The difficulty and high dimensionality of separating REEs concentrated from a low grade feedstock necessitates improvements in process modeling, experimental design, and process data analysis.

2.2 Bayesian Inference

Many popular statistical methods rely on asymptotic statistical theory including generation of point estimates for non-normally distributed independent and identically distributed random variables, confidence intervals, and calculation of p-values (DasGupta 2008). For a sample of size n, $\{Y_1, Y_2, \ldots, Y_n\}$, form a population, as $n \to \infty$ the mean of the sample one would observe $\bar{Y} = \frac{1}{n}\sum_i^n Y_i$ is *asymptotically normally distributed*, and confidence intervals can be constructed. Relying on the asymptotic normality assumption can be problematic for a small n sample size Hoff (2009).

Borrowing an example from Hoff (2009), multiple coin flips can be modeled using the binomial distribution $\binom{n}{k}p^k(1-p)^{n-k}$, where n is the number of coin flips, k is the number of times the coin landed on heads, and p is the probability of heads. Through some quick math, we can see that the unbiased estimator for p is $\hat{p} = \frac{k}{n}$. To build a confidence interval from the data, one could then use the confidence interval $\hat{p} \pm \sqrt{\hat{p}(1-\hat{p})/n}$. One can see that for large values of $\hat{p}$ and small n parts of the interval lie outside of $[0, 1]$. Hoff (2009) further points out the problem that if $\hat{p} = 0$, the width of the confidence interval 0.

In recent years, there has been serious critique and reaction to how some of these classical methods are used including, linking p-values to the scientific reproducibility crisis (Nuzzo 2014), and a ban on using null-hypothesis testing and confidence intervals by the journal *Basic and Applied Social Psychology* because the editors felt these methods were used to support shoddy research (Trafimow and Marks 2015).

One alternative to frequentest methods is Bayesian methods. The cornerstone of Bayesian statistical inference is Bayes' rule, shown in Equation (2.1).

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \tag{2.1}$$

With y being considered as data, and θ being a set of parameters in a statistical model, line 1 of Equation (2.1) reads as *the probability of theta, given the data, is equal to the probability of observing the data, given the value(s) of theta times the probability of theta, divided by the probability of observing the data.* $p(y)$ is a constant, so often Bayes rule is stated as $p(\theta|y) \propto p(y|\theta)p(\theta)$. Proportionality is often all that is necessary for parameter inference.

Using Bayes' rule a posterior distribution for model parameters $p(\theta|y)$ is inferred from the likelihood of observing the data $p(y|\theta)$ and a prior distribution on the model parameters $p(\theta)$. The subjective choice of a prior distribution is a common critique of Bayesian inference (Gelman 2008). However, for many applications noninformative prior distributions are chosen (Yang and Berger 1996). Conjugate prior distributions, where the prior and the posterior have the same form, are chosen for analytical purposes and can be adjusted to reduce influence over the posterior. Other prior distributions are chosen for objective reasons, such as choosing a truncated normal distribution with a lower bound of 0 to model a mass flow rate (Koermer and Noble 2021). Empirical Bayes is when the data informs the prior distribution, and some discussion on the topic can be found in (Kass and Steffey 1989).

One benefit of Bayesian inference is the full quantification of uncertainty for model parameters, instead of point estimates. Bayesian inference often takes advantage of statistical simulation and Markov Chain Monte Carlo (MCMC) methods to draw samples from a posterior distribution (Hoff 2009; Robert and Casella 2013). A Gibbs sampler iterates over draws from the posterior distribution of each model parameter, conditioned on the data and values of other model parameters ($p(\theta_1|\theta_2, Y)$ and $p(\theta_2|\theta_1, Y)$), to approximate the marginal distribution of each model parameter ($p(\theta_1|Y)$ and $p(\theta_2|Y)$) (Gelfand et al. 1990; Casella and George 1992).

Bayesian inference can be used for model parameters where there is no clear method for estimation through non-Bayesian methods (Hoff 2009). In the event that a posterior distribution of a model parameter does not have a known distributional form, algorithms such as Metropolis-Hastings (Hastings 1970) or a slice sampler (Neal 2003) can be used.

For a linear model of the form $Y = X\beta + \epsilon$, where Y and β are vectors and X is a matrix, Bayesian inference allows for the predictors represented in the columns of X to be chosen as significant probabilisticaly (Mitchell and Beauchamp 1988; George and McCulloch 1993), and predictions can be made using all models considered weighted by their model probability (i.e., Fernandez, Ley, and Steel 2001).

Similar to a hypothesis test, the ratio $p(y|\mathcal{M}_1)/p(y|\mathcal{M}_2)$ known as a Bayes Factor can be used to directly compare two models $\mathcal{M}_1$ and $\mathcal{M}_2$ by comparing how well each model explains the data (Jeffreys 1935). Bayes factors can be used to give evidence of a null hypothesis, differing from how p-values should be used (Kass and Raftery 1995). Finding $p(y|\mathcal{M})$ can require solving intractable integrals, one method for approximating $p(y|\mathcal{M})$ using the output of a Gibbs sampler can be found in Chib (1995). Model selection procedures can aid engineers in decision making regarding plant design and operation. For more on Bayesian analysis and Decision theory see Berger (2013).

The popularity of machine learning has promoted the popularity of Bayesian inference. Bayesian inferences has applications in evolutionary biology (Huelsenbeck et al. 2001); particle physics (Feroz, Hobson, and Bridges 2009), cosmology (Feroz, Hobson, and Bridges 2009; Trotta 2008), and gravitational wave astronomy (Ashton et al. 2019); and in estimating model parameters in stochastic chemical kinetic models for biological systems (A. Gupta and Rawlings 2014).

2.3 Mass Balancing

Mass balancing, or data reconciliation, in mineral processing systems estimates true average sample values from data under the constraints of mass sometimes energy conservation (Wills 2006; Romagnoli and Sanchez 1999). For a process at steady state, the Feed (F) mass into a system equals the Concentrate (C) mass plus the Tailings (T) mass. This is also true for individual components, which can be calculated as the mass of a process stream times the mass fraction (f, c, t) of a component (ie., Ff). Reconciled data has many uses including plant monitoring and testing (Nel, Martin, and Rabbe-Sgs 2004), process control (Bai, Thibault, and McLean 2006), and simulation calibration (Bai, Thibault, and McLean 2006).

BILMAT is a popular data reconciliation algorithm designed to work with data in discrete time intervals (Makni and Hodouin 1994). The user manually tunes a *backward horizon* to use when reconciling the results over time. Bilmat can produce poor results when the time window chosen is large.

In Makni and Hodouin (1994), only diagonal covariance matrices were considered in the parameter estimation procedure. Good estimates of the covariance in the errors between sample components and samples is critical for good estimates of true masses (Romagnoli and Sanchez 1999; Vasebi, Poulin, and Hodouin 2014). Maximum likelihood estimation (MLE) methods have been developed to estimate covariance of the error between samples and sample components (J. Chen, Bandoni, and Romagnoli 1997; Keller, Zasadzinski, and Darouach 1992), however these methods require large data sets, and can have computational convergence problems (Darouach et al. 1990; Keller, Zasadzinski, and Darouach 1992).

A classical mass balance which assumes Gaussian error can provide poor estimates of true sample grade (Koermer and Noble 2021). The problem is metaphorically equivalent to *putting a square peg in a round hole.* An unbounded Gaussian distribution models random variables which can range from $-\infty$ to ∞, while the grade expressed as a percentage ranges from 0% to 100%. Gaussian error is inappropriate for a low or high grade sample, especially when there is high variability. The use of Gaussian error in mass balances has been problematic for low grade material flows in the platinum industry. The sampling theory of particulate material (Pitard 1993; Lyman 2020) recognizes this problem and suggests modeling low grade samples using a Poisson distribution.

Bayesian methods for mass balancing data in mineral and chemical processing systems is notably sparse. Some models using Bayesian logic and methods for gross error detection entered the literature decades ago with mixed results (Tamhane, Iordache, and Mah 1988; Romagnoli and Sanchez 1999). More recently, Bayesian methods for data reconciliation when there is no data for some of the sampling locations has been published with successful results (Cencic and Frühwirth 2015).

2.4 Solvent Extraction

Solvent Extraction (SX), also known as liquid-liquid extraction, is a means of separating components using the solubility characteristics of components of interest between two immiscible liquid phases (Lo, Baird, and Hanson 1983; J. Zhang, Zhao, and Schreiner 2016). Typically one imiscible liquid is an aqueous, water based, solution, while another is an organic liquid similar to kerosene. For example, in nickel and cobalt recovery an ore can be leached using sulfuric acid or ammonia aqueous solution, which is mixed with an organic phase (*oil*) made of 10% Dinonylnaphthylsulfonic acid and kerosene (Lo, Baird, and Hanson 1983). The Dinonylnaphthylsulfonic acid is an *extractant*, chosen for it's properties for selectively extracting cobalt and nickel from the aqueous phase. Both of these phases can be mixed to create an emulsion, increasing the surface area between the two phases. The increase of interfacial surface area allows for greater rate of diffusion across the boundary. After agitation stops the imiscible liquids separate, similar to how oil and vinegar separate in salad dressing.

For the nickel/cobalt process, the organic phase can then be mixed with an aqueous hydrochloric acid solution to *strip* the cobalt and nickel from the

Table 2.2: Survey of methods which can be used to model solvent extraction processes.

Method	Theoretical Basis	Model Deficency	Relevant Works
Distribution Ratios	Nernsts Distribution Law	Assumes activity coefficients are independent of solute concentration.	J. Zhang, Zhao, and Schreiner (2016) C. K. Gupta and Krishnamurthy (1992)
McCabe-Thiele Diagram	Nernsts Distribution Law	Graphical method not generalizable to multiple elements and condition sets.	McCabe and Thiele (1925)
Emperical distribution ratio	Nernsts Distribution Law	Requires selection of the form of $\hat{f}$.	C. Zhang, Zhang, and Schreiner (1995)
Separation Factors	Nernsts Distribution Law	Assumption of constant separation factors.	C. K. Gupta and Krishnamurthy (1992) Larochelle and Kasaini (2016)
Kinetic Modeling	Law of Mass Action	Necessary to assume chemical reaction steps.	Espenson (1995) Arnaut and Burrows (2006) Temkin, Zeigarnik, and Bonchev (1996)
Stochastic Modeling	Markov property	Necessary to assume elementary chemical reaction steps. Analytically intensive.	McQuarrie (1967)

organic phase into an aqueous solution. This aqueous solution is then mixed with an organic phase containing the extractant Triisooctylamine dissolved in the diluant toluene. Cobalt is stripped from the organic phase by mixing with water and can be precipitated through further processing of the aqueous solution, including electrolysis (Lo, Baird, and Hanson 1983). In this way, elements can be targeted, separated, and concentrated based on solubility properties.

Processes for concentrating REEs often use organophosphorus acids for extractants (C. K. Gupta and Krishnamurthy 1992). As an element moves between the aqueous and organic phases, a chemical reaction occurs, possibly the one later listed as Equation (2.6).

2.4.1 Solvent Extraction Modeling

There are a variety of methods for modeling SX processes. Some methods model equilibrium conditions, while others can be used to model the reaction over time. A survey of methods is tabulated in Table 2.2. Often, there is a trade off between the severity of the assumptions made and model complexity. Methods based on Nernsts Distribution Law are the simplest, but assume constant behavior over a variety of concentrations. Kinetic methods require a scientist to assume a set of elementary reaction steps which are not possible to know (Espenson 1995). Stochastic methods have the potential to be overkill and more useful for systems on the microscale. Further discussions on the details of these methods are in the proceeding sections.

A primary factor for selecting a model is related to the reason for modeling in the first place. For example, if one wanted to infer kinetic constants, they would have to choose a kinetic model. However, along the lines of *all models are wrong, but some are useful* (Box 1976), no matter what model is selected, there will always be some *bias* (skip ahead to §2.5.3) between observations and model predictions. If the purpose of modeling is for prediction, selecting which model to use can be difficult.

Any phenomenological model has to be fit to data. An experimental procedure where data is collected for SX equilibrium modeling is dependent on the purpose and required precision of the model in development. One may want to control interfacial surface area so precisely, that the moving drops technique, where a single droplet of one phase is allowed to move through a vertical tube of another phase is employed (Roberto Danesi, Chiarizia, and Coleman 1980).

Often, when collecting data, a volume of an aqueous liquid and organic liquid are placed into a vessel at various organic to aqueous volumetric ratios (O:A), and agitated until it is assumed the mixture is at an equilibrium state J. Zhang, Zhao, and Schreiner (2016). The two phases are allowed to disengage and separated for analysis. Such a test is referred to as a *shake test* throughout this article. Solvent extraction data can also be gathered as observations of a full SX process. As long as the data required for the model is collected, such data can be useful for informing model parameters.

2.4.1.1 Distribution Ratios

The use of distribution ratios for modeling the equilibrium concentrations of a solute in two immiscible liquid phases is common in liquid-liquid extraction literature for rare earth elements (J. Zhang, Zhao, and Schreiner 2016; C. K. Gupta and Krishnamurthy 1992). Lyon, Utgikar, and Greenhalgh (2017) uses distribution ratios within a system of differential equations to model cascades of mixer-settlers.

J. Zhang, Zhao, and Schreiner (2016) define a distribution ratio, as shown in Equation (2.2), using Nernst's distribution law. The distribution ratio, K_d, provides the ratio of concentrations of solute M in solvents A and B. J. Zhang, Zhao, and Schreiner (2016) notes that because distribution ratios are valid for a constant pressure, temperature, and chemical potential for immescible pure solvents the *"distribution ratio is not the distribution constant"*.

$$K_d = \frac{[M_{(A)}]}{[M_{(B)}]} \tag{2.2}$$

J. Zhang, Zhao, and Schreiner (2016) then use chemical potential to derive Equation (2.3), showing that the distribution ratio $K_d^\star$ is only a constant if the activity coefficients, γ, are independent of solute concentration.

$$K_d^\star = K_d \frac{\gamma_A}{\gamma_B} \tag{2.3}$$

Assuming the independence of activity coefficients from solute concentration is questionable, and assuming a constant distribution ratio can lead to serious errors when developing liquid-liquid extraction processes (J. Zhang, Zhao, and Schreiner 2016). Since K_d dependent on a variety of conditions, it is possible to measure K_d for specified conditions, such as temperature and impurity concentrations, to predict SX behavior.

A well known method for estimating how K_d changes with varying parameters is the isotherm, or McCabe-Thiele, method (McCabe and Thiele 1925). Using this graphical method, one can estimate K_d for varying concentrations of the solute in one of the solvent phases, using the relationship $[M_{(o)}] = f[M_{(a)}]$. An example isotherm of the same form as a figure in J. Zhang, Zhao, and Schreiner (2016) is shown in Figure 2.1.

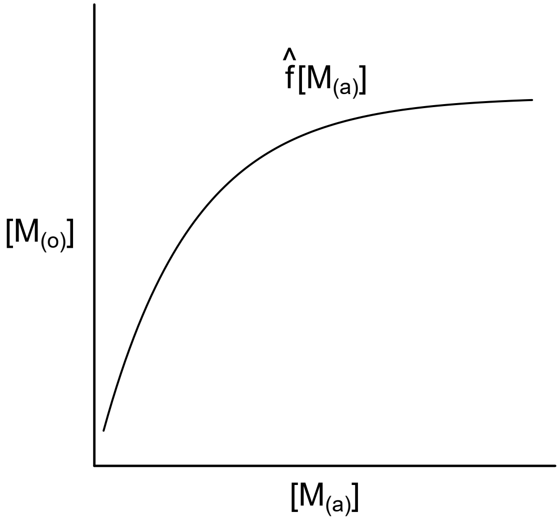

Figure 2.1: Example McCabe-Thiele isotherm, similar to example in (J. Zhang, Zhao, and Schreiner 2016).

Unfortunately estimating the function f, as $\hat{f}$, for concentrations of p elements in the aqueous phase to jointly predict the concentration of p elements in the organic phase cannot be completed graphically when $p > 1$. Other methods are necessary for taking interactions between concentrations into account.

C. Zhang, Zhang, and Schreiner (1995) fit an equation, of the form shown in Equation (2.4), to produce $\hat{f}$. This model is able to make predictions of the concentration of M in the organic phase, given the equilibrium concentration of M in the aqueous phase and hydrogen ion concentration, by estimating fitting parameters $\alpha_1, \ldots, \alpha_6$ from laboratory equilibrium data. This particular approach allows $\hat{f}$ to be written down as an equation, so calculation of K_d at various conditions does not rely on a hand drawn sketch of $\hat{f}$ when using a graphical method.

$$[M_{(o)}] = \alpha_1 [M_{(a)}]^{\alpha_2} e^{\alpha_2 [M_{(a)}]} H^{(\alpha_4 + \alpha_5 H + \alpha_6 H^2)} \tag{2.4}$$

While fitting an empirical equation to data improves on the issues related to sketching a curve by hand to find $\hat{f}$ for the graphical isotherm method, fitting an empirical equation for the equilibrium concentration of a single element does not take into account interactions between elements present.

A simple method for handling the $p > 1$ case is using separation factors (C. K. Gupta and Krishnamurthy 1992), a ratio of distribution ratios between two elements as specified as in Equation (2.5).

$$\beta_{M,N} = \frac{K_M}{K_N} \tag{2.5}$$

Typically separation factors are compared to determine if a separation is possible. In the case that $K_d \approx 1$ for a set of conditions, a separation between elements will be difficult or impossible. Larochelle and Kasaini (2016) introduced an iterative method of predicting distribution ratios and equilibrium concentrations, by specifying a constant distribution ratio of Nd and predicting distribution ratios for other elements using separation factors relative to Nd. Even with this simplification removing some information, Larochelle and Kasaini (2016) reported good results between results from a simulation and observations from a process plant at steady state.

Pavón et al. (2019) introduced a method of simultaneously modeling cerium, europium, and yttrium concentrations using chemical equilibrium relationships. The model was calibrated using experimental data

2.4.1.2 Kinetic Models

When an element moves from one liquid phase to another, a chemical reaction takes place. For an organophosphorus acid in the organic phase, reacting with a Rare Earth Element (REE) in the aqueous phase, the reaction can be written as in Equation (2.6) (C. K. Gupta and Krishnamurthy 1992).

$$[\mathrm{RE}^{3+}]_{\mathrm{aq}} + [3(\mathrm{HA})_2]_{\mathrm{org}} \rightarrow [\mathrm{RE}(\mathrm{HA}_2)_3]_{\mathrm{org}} + 3[\mathrm{H}^+]_{\mathrm{aq}} \tag{2.6}$$

Kinetic models in chemistry are useful for understanding the rates of chemical reactions (Espenson 1995), and can be used to study the SX equilibria under investigation. A chemical reaction can be written as shown in Equation (2.6) can be rewritten in the general form shown in Equation (2.7), representing chemical species with an uppercase letter, and taking into account stoichiometry with a lowercase letter.

$$aA + bB \rightarrow cC + dD \tag{2.7}$$

The rate law for the reaction in (2.7) can be written as $\nu = k[A]^a[B]^b$ (Arnaut and Burrows 2006). The rate constant k is dependent on reaction medium, pressure, and temperature, but independent of concentrations of A and B. Given the reaction and stoichiometric relationship in Equation (2.7), the differential equation relationships in Equation (2.8) can be written (Espenson 1995).

$$\nu = -\frac{1}{a}\frac{d[A]}{dt} = -\frac{1}{b}\frac{d[B]}{dt} = \frac{1}{c}\frac{d[C]}{dt} = \frac{1}{d}\frac{d[D]}{dt} = k_1[A]^a[B]^b \tag{2.8}$$

Where k_1 is a rate constant and $[X]$ is the concentration in mol/L. A reversible reaction is shown in Equation (2.9).

$$aA + bB \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} cD + dD \tag{2.9}$$

The rate law for Equation (2.9) can be written as

$$\nu = k_1[A]^a[B]^b - k_{-1}[C]^c[D]^d = \nu_1 - \nu_{-1} \tag{2.10}$$

When the system is at equilibrium, the reaction rates ν_1 and ν_{-1} are equal, and the equilibrium constant K_{eq} can be derived (Arnaut and Burrows 2006).

$$\left(\frac{k_1}{k_{-1}}\right) = \frac{[C]^c_{\text{eq}}[D]^d_{\text{eq}}}{[A]^a_{\text{eq}}[B]^b_{\text{eq}}} = K_{eq} \tag{2.11}$$

The differential equation for the concentration $[A]$ *could* be written as a relationship between the forward and reverse reactions as shown below.

$$\frac{1}{a}\frac{d[A]}{dt} = k_{-1}[C]^c[D]^d - k_1[A]^a[B]^b \tag{2.12}$$

However Equation (2.12) is a notable mis-specification as it assumes the simultaneous collision of more than three molecules. Firstly, the probability of more than three particles colliding simultaneously approaches zero (Arnaut and Burrows 2006). On the macro scale, Equation (2.9) is a valid representation of the reaction, but on the micro scale it is necessary to work with elementary reactions, where the order of each reaction is two or less. The order of a reaction is defined as the sum of the stochiometric coefficients of the reactants for the reaction's rate law (Arnaut and Burrows 2006). A clear example of why it is advantageous to model equilibrium using elementary reactions is shown on page 102 of Temkin, Zeigarnik, and Bonchev (1996).

Reaction order is determined experimentally, and the reaction mechanism is not possible to prove, but can be disproven through experimental results (Espenson 1995). For some reactions, data can suggest a fractional reaction order (Espenson 1995), typically for chain reactions such as a reaction in the form of $A+nB \leftarrow AB_n$ (Temkin, Zeigarnik, and Bonchev 1996). A reaction can also be of zero order with respect to a component, i.e. $k[A]^0[B] = k[B]$, implying the rate is independent of the concentration of A (Espenson 1995). For an example where the determination of a zero order reaction is from experimental data, the equilibrium model structure itself would be inferred from the data.

In breaking down a reaction into its elementary reactions, one would account for all intermediate chemical complexes formed. Instead of using experimentally determined reaction orders for a model, the law of mass action can be used to derive differential equations for a system of elementary reactions (Horn and Jackson 1972). The law of mass action is shown by Equation (2.13) and Equation (2.14), taken from L. Chen et al. (2010). Equation (2.13) specifies a single reaction with reactants which could be $R_1, R_2, \ldots, R_m$, which have stoichiometric coefficients $r_1, r_2, \ldots, r_m$, and products labeled similarly from a set of size n. k is the rate constant of the reaction. The reaction rate of Equation (2.13) is given in Equation (2.14) as a differential of a reactant or product concentration with respect to time. A system of differential equations for multiple reactions and all reactants can be derived by summing over all reaction rates found using Equation (2.14).

$$r_1R_1 + r_2R_2 + \cdots + r_mR_m \xrightarrow{k} p_1P_1 + p_2P_2 + \cdots + p_nP_n \tag{2.13}$$

$$r = -\frac{1}{r_i}\frac{d[R_i]}{dt} = \frac{1}{p_j}\frac{d[P_j]}{d_t} = k\prod_{l=1}^{m}[R_l]^{r_l} \tag{2.14}$$

The law of mass action provides a mechanism for modeling equilibrium concentrations, given the elementary reactions. A critique of using the law of mass action for modeling chemical equilibrium is that the law has been shown to only be approximately valid, is deterministic, and is not well suited for small systems (McQuarrie 1967).

2.4.2 Stochastic Methods

McQuarrie (1967) provides methods for stochastic kinetic modeling for first order reactions using both discrete (DTMC) and continuous (CTMC) time Markov chains. McQuarrie (1967) argued that there is some evidence for a Markovian basis for chemical reactions and that the law of mass action is an approximation and does not generalize well to smaller systems.

For the bimolecular reversible reaction $\mathrm{A} + \mathrm{B} \underset{k_2}{\overset{k_1}{\rightleftharpoons}} \mathrm{C} + \mathrm{D}$ a CTMC system can be solved analytically, with solutions and procedures for model parameter interpretation for the system given in Darvey, Ninham, and Staff (1966). More on CTMCs can be found in Anderson (2012).

2.5 Gaussian Processes

2.5.1 Gaussian Process Models

Mineral process engineers are frequently employed to technically or economically model a process to assess feasibility, predict revenue and costs, and suggest efficiency improvements. Process models are derived based on a mixture of literature and data observed in a laboratory or processing plant.

In general terms, any model will aim to predict an outcome given a set of conditions. Equation (2.15) is an example of how an outcome or observation, y, can be modeled as a function f, with conditions $x_1, x_2, x_3, \ldots, x_m = X$, plus some error ϵ.

$$y = f(X) + \epsilon \tag{2.15}$$

Sometimes $f(X)$ can be written down, because the mechanisms leading to the outcome are well understood. Often, phenomena are not fully understood.

Response surface methodology may be chosen to estimate $f(X)$ when the relationship between a set of conditions and an outcome are unknown (Box and Draper 2007). An estimate of $f(X)$ can be written as $\hat{f}(X)$ When using response surface methodology a polynomial can be used to approximate $f(X)$ for a sufficiently small range for all $x_1, x_2, x_3, \ldots, x_m$, and aid in gradient based optimization (Box and Draper 2007). There are various experimental designs which can be employed to better estimate polynomial coefficients with less data (Box and Draper 2007).

Response surface methods have been used in mining engineering applications to model the removal of uranium from mine water (Nariyan, Sillanpää, and Wolkersdorfer 2018), aid in the removal of lead from gold tailings (Demir and Derun 2019), and for the removal of iron and manganese from acid mine drainage (Núñez-Gómez et al. 2020). Response surface methods are useful for mineral processing engineers.

However, response surface methodology is not the only option for estimating $\hat{f}(X)$, and does have some disadvantages compared to other methods. Gaussian Process (GP) regression Gramacy (2020), is an alternate to response surface methodology which has been utilized in predicting the output of computationally intensive computer simulations (T. J. Santner et al. 2003), robotic controls (Deisenroth, Fox, and Rasmussen 2013), and spatial statistics (Gelfand and Schliep 2016).

A Gaussian process has similarities to the Kriging (Matheron 1963) techniques used in resource estimation. After drilling and analysis of boreholes, borehole data can be weighted using methods including the area of influence method, polygon method, extended area method, and inverse distance weighting to estimate ore body properties (Hartman and Mutmansky 2002). For inverse distance weighting methods, as the distance between a prediction location and a data point increases, the data point has less influence over the prediction. An illustration is shown in Figure 2.2.

A GP uses a function, often distance based, to weight data for making predictions at a location Gramacy (2020). While borehole data can be two or three dimensional, Euclidean distance calculations are generalizable to higher dimension by taking $\sqrt{\sum_{i=1}^{d}(a_i - b_i)^2}$ (Tabak 2014), where d is the number of dimensions.

The ability to calculate distance in higher dimension means that a GP is not limited to modeling two conditions, such as longitude and latitude. For a

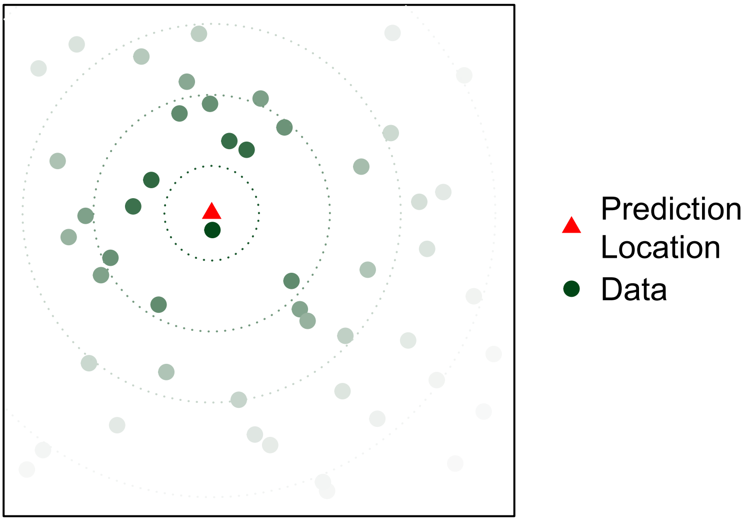

Figure 2.2: Example of how distance between a prediction location and data can be used to weight data for prediction.

mineral process this could mean modeling flotation time and various reagent additions simultaneously, with each unique set of test conditions representing a different set of coordinates or test *location*.

Simply weighting the importance of each point using distance can be problematic. For example, the effect of two reagents can be drastically different. A small change in reagent one may only produce a small difference in the outcome, which would imply close by as well as far away data points are useful for prediction. However, if a small change in reagent two produces a large difference in the outcome, more distant points along the reagent two concentration dimension are less influential for prediction.

A common choice for the function which weights the points, the GP correlation function between two locations, is the exponentiated negative squared distance between the two points, shown in (2.16) (Rasmussen 2003; Gramacy 2020).

$$\sigma^2(x, x') = \exp\left\{ -\sum_{i=1}^{m} \frac{(x_i - x_i')^2}{\theta_i} \right\} \tag{2.16}$$

The summation in (2.16) gives the squared distance between two locations x and x'. The distance in each dimension $i = 1, \ldots, m$ is rescaled by θ_i, allowing the distance along each dimension to be weighted differently.

For a seperable, deterministic, GP $\theta_1, \ldots, \theta_m$ are the fitting parameters in the model. $\sigma^2(x, x')$, in Equation (2.16), can be thought of as the correlation between responses at locations x and x'. Inspecting the GP model structure in (2.17) jointly illustrates how distance between two locations is used to estimate correlation between these locations.

$$f(y) = \frac{1}{(2\pi\tau)^{\frac{N}{2}}}|\Sigma|^{-\frac{1}{2}}e^{-\frac{1}{2}\tau Y^T \Sigma^{-1} Y} \tag{2.17}$$

Equation (2.17) is in the form of a multivariate normal distribution. In this case the mean of the distribution is 0 for all elements $y_1, y_2, \ldots y_N$ in vector Y. Each y_j is an observation at a corresponding location x_j, and one can think of $y_j = y(x_j)$. Σ is a covariance matrix, holding information about the correlation between two observations $y(x)$ and $y(x')$. The a, b entry of $Sigma$ is equal to $\sigma^2(x_a, x_b)$ in Equation (2.16). τ is a scaling parameter, on which details of the importance can be found in (Gramacy 2020).

Given the a covariance function, such as the exponential function in Equation (2.16), and some of the properties of the multivariate normal distribution, it is possible to use conditional probability identities to derive equations for mean predictive values as well as predictive variance at a given location x (Gramacy 2020).

2.5.2 Sequential Design

When traveling and using a new shower, there is uncertainty as to how many radians to turn the shower knob for maximal comfort. Often one will start off and turn the shower on a little (small θ), realize the water is too cold, then turn the knob up (large θ) and recognize the water is too hot. Then one can use the temperatures they have experienced to estimate where a good knob position may be. The shower knob can be turned to a third location and the temperature is ok, but it is a little cold, and one can tweak the knob location to gain some improved comfort level over the previously experienced temperatures. The knob position is then further tweaked sequentially changing the position to where one would expect to find the most improvement over the best previously observed temperature given the previously collected data.

Such a process is in Figure 2.3, illustrating how far and the sequence one may use when finding their optimal shower temperature. Importantly, when we search for this optimum, we do not use a known function for shower comfort.

Such a series of *experiments* in the pursuit of finding the best shower temperature can be considered a sequential experimental design, where experiments run are not fixed prior to the start of experimentation (Robbins 1952). Figure 2.3 shows a sequential design for optimization. While it would be believable that a human would make the sequential choices shown in Figure 2.3, the sequence was sequentially run using the Expected Improvement (EI) sequential design criteria (Schonlau 1997) and some functions for implementing Bayesian Gaussian Processes with the `tgp` package (Gramacy and Taddy 2009) in `R` for Efficient Global Optimization (EGO) (Jones, Schonlau, and Welch 1998).

Optimization using EI has advantages over gradient based methods, which require the experimenter to repeatedly approximate derivatives of the unknown,

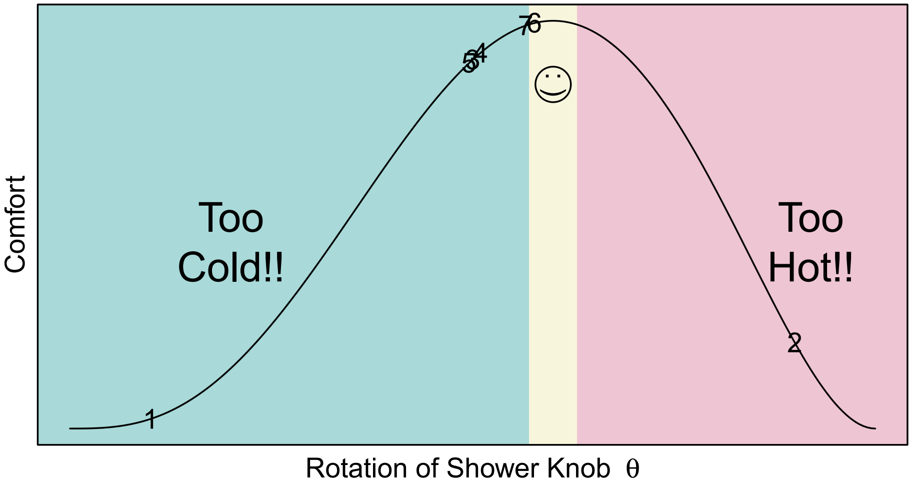

Figure 2.3: Learning the right temperature for a novel shower is an example of sequentially searching for the optimum of an unknown function.

black box, function using finite differences (Box and Draper 2007), requiring a large number of experiments to find and test the optimum. From the simulation surrogate literature, such methods can make computation time for optimization infeasible (Gramacy 2020). Imagine turning the shower on, finding it too hot, and then turning the knob to make the water colder, then hotter, and estimating the first derivative before proceeding! Further issues with gradient based sequential design optimization methods occur when a response surface is multi-modal, as gradient based optimization can converge to the local, and not necessarily global, optima (Gramacy 2020).

To find the point where one may find the greatest expected improvement at a test location x, given the best observed value along with the rest of the data, first one has to define EI mathematically. Improvement for maximization can be defined as $I(x) = max\{0, Y(x) - f^N_{max}\}$ where $Y(x) = f(x)$ for a function mapping $x \to Y$ and f^N_{max} is the best observed value for the previously obtained N data points (Schonlau 1997). The expectation, or expected value, is the mean value for a random variable, noted as $\mathbb{E}[Y] = \int_a^b Yf(Y)dY$ for the random variable $Y \in [a, b]$ which has a probability density function of $f(Y)$. The expected improvement at location x, and the analytical solution for the integral if $Y(x)$ is modeled as a GP, is shown in Equation (2.18) (Schonlau 1997).

$$\begin{aligned}\mathbb{E}[I(x)] &= \int_{f^N_{max}}^{\infty} (Y(x) - f^N_{max})\phi(Y(x))dY(x) \\ &= (\mu_N(x) - f^N_{max})\phi\left(\frac{\mu_N(x) - f^N_{max}}{\sigma_N(x)}\right) + \sigma_n(x)\Phi\left(\frac{\mu_N(x) - f^N_{max}}{\sigma_N(x)}\right)\end{aligned} \tag{2.18}$$

In Equation (2.18) $\mu_N(x)$ and $\sigma_N(X)$ are the predictive mean and standard deviation of $Y(x)$. ϕ and Φ are the standard normal probability density and cumulative density functions. Because with a GP, there are analytic solutions for predictive mean and standard deviation, a GP can easily be used for efficient global optimization of a black box function. At each step, in practice, a candidate set of x is generated, predictive means and standard deviations are calculated using a GP at each x, EI is then calculated at each x, and the x with the largest expected improvement is chosen for testing. To generate a candidate set x, often a Latin Hypercube Sample (McKay, Beckman, and Conover 1979) is used (Jones, Schonlau, and Welch 1998).

Sequential design has advantages not only in optimization, but in experimental design for predictive modeling. One can sequentially choose to test the experimental location with the highest predictive variance to obtain better estimates of the model parameters (MacKay 1992). Such a design criteria, when used with GPs, is called Active Learning McKay (ALM) (Seo et al. 2000).

An alternative is the Active Learning Cohn (ALC) criteria (Seo et al. 2000), which seeks to find the testing location which maximizes the average reduction in predictive variance over the design space through the additional experiment. For a candidate set of x_{N+1} testing locations an integral over the test space can be approximated at each candidate in the set, selecting the the best one. For noisy experiments, the value for variance at the location can be removed in calculation so that only uncertainty of the GP fit is reduced, such as in the IMSPE criteria implemented in Binois, Gramacy, and Ludkovski (2018).

2.5.3 Model Calibration

In this context, *model calibration* refers to parameter estimation of an equilibrium model described in §2.4.1, which could include distribution ratios, a system of differential equations, or a stochastic model. Typically, when calibrating a model with real data the relationship in Equation (2.19) is used in inference (Gramacy 2020), where Y^F represents observed field data, y^R represents the *real* process with noise removed, and ϵ represents normally distributed error.

$$Y^F(x) = y^R(x) + \epsilon \tag{2.19}$$

If y^R is a system of differential equations modeling SX equilibria, it would have a set of tuning parameters u, which would include chemical kinetics constants. Varying u in $y^R(x, u)$ would allow u to be varied for calibration.

There are multiple ways to use a GP in a calibration context. A method, suggested Kennedy and O'Hagan (Kennedy and O'Hagan 2001), restates Equation (2.19) as Equation (2.20). $y^R(x)$ in Equation (2.19) is replaced with a model dependent on x and u along with a bias function dependent only on x, noted as $y^M(x, u)$ and $b(x)$ respectively in Equation (2.20).

$$Y^F(x) = y^M(x, u) + b(x) + \epsilon \tag{2.20}$$

Equation (2.20) can be further reworked to provide a vector of residuals modeled by the bias for a n_F set of number of field data points and a given value of u (Higdon et al. 2004).

$$Y_{n_F}^{b|u} \equiv b(x_{n_F}) = Y^F(x_{n_F}) - y^M(x_{n_F}, u) \tag{2.21}$$

From Equation (2.21) a model for inference for u can take different forms depending on how expensive evaluations of the computer simulations providing $y^M(x_{n_F}, u)$. If the simulation is cheap to evaluate, likelihood on $Y_{n_F}^{b|u}$ can be maximized directly (Higdon et al. 2004). If the simulation is expensive to evaluate a GP can be fit to y^M based on a selection of model runs and details for calibration are changed to account for uncertainty in $\hat{y}^M$. It is possible to place a prior distribution on u in order to constrain the values, and inference can be conducted via MLE or MCMC.

Kennedy and O'Hagan's original method is good for augmenting a simulator for good field data predictions, but the simultaneous inference of simulator and GP parameters leads to a high degree computational complexity (Gramacy 2020).

An update on the method, called modularization (Bayarri, Berger, and Liu 2009), fits a GP surrogate to a computer simulation independent of the field data, with inputs of x, u. Next, u is calibrated by finding the setting of u which maximizes the likelihood of the residuals between the field and model data.

2.5.4 Other GP Models

2.5.4.1 Multiple Outputs

GP models described previously assume a scalar output, meaning at location x a $y = f(x)$ is a single value. However, there are methods for producing vector valued outputs, which could be advantageous for predicting concentrations of multiple chemical elements simultaneously.

One method for producing multiple outputs at the same x location is augment x with u, a vector or matrix giving indication for a discrete input to $f(x, u)$. If u can indicate two discrete, or qualitative, factors, two different values can be predicted at the same testing location x. P. Z. G. Qian, Wu, and Wu (2008) provides a GP model structure and parameter estimation techniques for GP models with both continuous quantitative and a binary qualitative factor. The model essentially uses a latent variable which estimates the squared *distance* between the two factors in one dimension.

Y. Zhang et al. (2020) expands upon the methods in P. Z. G. Qian, Wu, and Wu (2008) by presenting a method for modeling a larger number of discrete factors, using latent variables to place the factors in a higher dimensional space for distance calculations. Authors suggest placing the latent variables in a 2D space for the advantages of allowing a more complex spatial relationship then placing factors on a line, but without drastically increasing computational complexity. For approximating integrals over an input space which contains both continuous and discrete variables, a sliced Latin Hypercube Sample can be used (P. Z. Qian 2012).

Methods in Y. Zhang et al. (2020) allow for a GP model to be used if data is not available for each discrete factor at each x. This would be advantageous if it was necessary to obtain data for each discrete factor individually. However, with chemical equilibrium experiments, ICP-MS data typically contains information on all elements of interest simultaneously. With little to gain for implementation of the model in Y. Zhang et al. (2020) for this application, the increase in the number of latent variables does not make such a model the best choice.

Cokriging (Ver Hoef and Barry 1998) takes a different approach and allows for vector valued output at a single input location. A covariance matrix between the outputs is calculated in closed form, and estimation of other GP parameters are similar to a scalar output GP. However, for some data generating mechanisms Cokriging may be an unrealistic model (Gramacy 2020). It is expected that there will be covariance between most of the elements in equilibrium. If the Cokriging model is not realistic, coregionalization may provide the necessary flexibility (Bourgault and Marcotte 1991).

2.5.4.2 Error Structure

When fitting a line to data, commonly variance in the response is assumed to be a constant for any x. Constant variance is called homoscedasticity. A model which allows for heteroskedasticity, allows for the variance to change along x. One heteroskedastic GP model is stochastic kriging (Ankenman, Nelson, and Staum 2008), which has good properties for separating signal from noise, but requires a minimum number of replications at each testing location and makes the inclusion of exploratory testing locations difficult. Treed Gaussian Processes (Gramacy 2005) segment the input space and fit a GP to each segment allow for heteroskedasticity but variance may not evolve smoothly as one would expect with real data.

Binois, Gramacy, and Ludkovski (2018) published a model which fits a GP to the variance parameter in a typical GP model, using latent variables to estimate variance at tested locations which allow for variance predictions at untested locations. The model in Binois, Gramacy, and Ludkovski (2018), and implemented in Binois and Gramacy (2021b), allows for IMSPE sequential design criteria.

2.5.4.3 Stationarity

While GPs may appear more flexible than linear regression, they are inflexible in other respects. Stationarity in this context is the assumption that the way points are weighted with regard to distance in each dimension is constant. If dynamics change drastically in one area of the input space, this assumption may be unrealistic.

Examples of nonstationary GP models include the selection of a subset of neighboring data points for prediction (Emery 2009), treed GPs (Gramacy

2005), and deep GPs (Sauer, Gramacy, and Higdon 2020).

2.5.5 GPs and Real Data

Gaussian Processes have been used on a wide array of real data outside of geostatistics. Applications involving real data include successful autonomous robot learning and control (Deisenroth, Fox, and Rasmussen 2013), accurate modeling of the growth of microbial populations (Tonner et al. 2017), calibration of mass and infrared spectroscopic tools (T. Chen, Morris, and Martin 2007), and for the prediction of wastewater effluent streams (Hvala and Kocijan 2020).

Chapter 3

The utility of Bayesian data reconciliation for separations

This chapter was accepted for publication in the international peer reviewed journal Minerals Engineering. Citation details are provided below:

Koermer S, Noble A (2021). “The utility of Bayesian data reconciliation, for separations.” *Minerals Engineering*, *169*, 106837.

Abstract

Data reconciliation methods for separation processes typically rely on classical statistical approaches to generate estimates of true mass flow rates from measurements. Knowledge regarding the uncertainty of these estimates has value in decision making, but is often not acquired. Bayesian approaches intrinsically quantify uncertainty; however, literature for Bayesian data reconciliation of separation processes is scarce. This publication outlines two Bayesian data reconciliation models and provides details for how the models were implemented for the `BayesMassBal` (V 1.0.0) software package written in `R`. To demonstrate the advantages of this approach for data reconciliation, the models were first applied to simulated data and then compared to a classical model through a Monte Carlo experiment. In this example, the Bayesian models were found to provide more accurate estimates of the simulated data, while also providing quantitative information on the estimate uncertainty. To demonstrate the use of the technique in a practical problem, the models were also applied to real data collected from a pilot-scale rare earth solvent extraction process. This publication provides a small window into how Bayesian methods can be used for data reconciliation, but findings suggest Bayesian data reconciliation models for separation processes have distinct advantages over classical alternatives.

3.1 Introduction

The interpretation and utilization of raw plant data from mineral processing operations are non-trivial tasks that requires both mathematical rigor and expert judgment. In most cases, the raw assay data are initially used to complete a steady-state flowsheet balance for all major constituents, such as total solids, water, target mineral species, and gangue minerals. This balanced flowsheet is in turn used to calculate metallurgical process performance indicators, such as throughput, recovery, and rejection. In cases where redundant data have been collected (e.g., the feed and both products of a simple separator have been assayed), the flowsheet balance almost certainly lacks internal consistency, as the steady-state mass balanced condition is not achieved for all components simultaneously. Use of this poorly balanced data can lead to nonsensical calculations of performance metrics and in turn lead to misguided or erroneous decisions on process improvement.

Poor sampling, unreliable assay procedure, and random process variation can be contributing factors to poorly balanced datasets, even in process operations with tight controls on representative sampling and assays. To mitigate these issues and produce an internally consistent dataset, engineers must employ a data reconciliation (also known as mass balancing) technique to filter noise so a consistent picture of the true behavior of the system can be obtained.

Data reconciliation leverages the conservation of mass and energy to aid in separating true values from noise and variability (Wills 2006; Romagnoli and Sanchez 1999). For a single node process at steady state, consisting of a feed (F), a concentrate (C), and a tailings (T) stream, the mass entering a unit operation must be equal for both the total mass (3.1), and the component mass (3.2), as given by:

$$F = C + T \tag{3.1}$$

$$Ff = Cc + Tt \tag{3.2}$$

Where f, c, and t are the grade of the feed, concentrate, and tailings respectively, in fractional form. Commonly, a sum of weighted least squares criteria is used to find the best point estimates for the true values of grade (Romagnoli and Sanchez 1999).

There are many data reconciliation methods. For a given data set, some are more useful than others. Model specification can influence the accuracy of results. For example, BILMAT, a popular algorithm for data reconciliation in mining applications, can handle time correlated data, but can produce poor results when a too large time window is chosen (Makni and Hodouin 1994).
Similarly, specifying covariance structure is important. Estimation of the covariance between measurement errors is crucial to trustworthy data reconciliation (Romagnoli and Sanchez 1999). Vasebi, Poulin, and Hodouin (2014) showed data reconciliation model covariance mis-specification produces sub optimal estimates.

Proper estimation of the covariance is important, but difficult. The publications J. Chen, Bandoni, and Romagnoli (1997) and Keller, Zasadzinski, and Darouach (1992), furnished and tested methods for estimation of a covariance matrix. Each method is tested using 1,000 observations, a testament to the large data sets required to produce good point estimates for each element of a covariance matrix. Obtaining similarly large data sets is not always feasible. Furthermore, some estimation procedures, such as maximum likelihood estimation (MLE), can have computational and convergence problems (Darouach et al. 1990; Keller, Zasadzinski, and Darouach 1992).

Regardless of the reconciliation method employed, the flow balance estimates produced from a limited dataset will never be completely free from error. Estimates produced from a model are dependent on the data observed, implying that the variability present in the data will produce variability in the estimate when different data sets from the same source are used. Nevertheless, the preponderance of uncertainty does not preclude the use of such data in downstream decision making. Reconciled data from process operations is used for simulation calibration (Reimers, Werther, and Gruhn 2008), process control (Bai, Thibault, and McLean 2006), and decisions with a scope ranging from daily operations to large capital investments and environmental sustainability. In an ideal case, the data reconciliation model would provide the information needed for uncertainty quantification so as to evaluate the magnitude and risk of the decision against the reliability of the source data, as well as aid in selecting the proper model for a data set.

Bayesian statistical methods (Hoff 2009) allow for such models. Bayesian inference results in probability distributions, instead of point estimates, of model parameters, given the observed data and a model structure. Algorithms are used to produce random draws from parameter distributions. The draws obtained can be used in other calculations and analyses to gain further insight into a process. A Bayes factor (Kass and Raftery 1995) can be used to aid in selecting which Bayesian model is the best representation of a data set. Model selection procedures and an understanding of the uncertainty of the results from a model can substantially improve estimation accuracy and the understanding of risk for decision making.

Prior studies in the technical literature show how Bayesian methods can be applied to the data reconciliation of chemical processes. Tamhane, Iordache, and Mah (1988) outlined a Bayesian model for gross errors in the observations of a chemical process. In Romagnoli and Sanchez (1999), a model incorporating Bayesian logic is used for data reconciliation of processes with gross and random errors with mixed results. Methods for reconciling process data between measured and unmeasured sampling locations, where error for process inputs and outputs is independently distributed have shown favorable results (Cencic and Frühwirth 2015). While the popularity of Bayesian methods is increasing, studies in the literature using Bayesian methods for mineral processing are still sparse. Of particular note are studies comparing classical methods and Bayesian methods, financial modeling using Bayesian methods, and Bayesian model selection related to data reconciliation models.

The goal of this publication is to illustrate some of the advantages and disadvantages of Bayesian methods, examine some of the ways Bayesian data reconciliation can aid in decision making, and guide the reader through using the `BayesMassBal` package (Koermer 2020a), written for the `R` programming language as a supplement to this publication. Methods (§3.2) outlines, model structure, Bayesian inference used for model parameters and model selection, approximation of the main effect of a variable independent of the process on a function dependent on process metrics. §3.2.4 describes how to execute the methods in §3.2 using the `BayesMassBal` package.

The Experimental section (§3.3) gives details of how simulated and real data were used to test the usefulness of the Bayesian models. This section includes details for the procedures used to apply the models to simulated data once, conduct a Monte-Carlo experiment comparing the average behavior of the Bayesian models and a point estimate model, and apply the Bayesian models to real data. Results from the Experimental section and the authors' interpretation is provided in the Results and Discussion section §3.5.

3.2 Methods

3.2.1 Model Derivation

Observed mass flow rates for a simple separation process (e.g. one input, two outputs) can be modeled as shown in Equation (3.3), equivalently Equation (3.4), where $\boldsymbol{\beta}$ is a vector of true mass flow rates, $\boldsymbol{\epsilon}$ is a vector of random noise, and $\boldsymbol{y}$ is a vector of observed mass flow rates.

$$\boldsymbol{y} = \boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{3.3}$$

$$\begin{bmatrix} y_{\mathrm{F}} \\ y_{\mathrm{C}} \\ y_{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} \beta_{\mathrm{F}} \\ \beta_{\mathrm{C}} \\ \beta_{\mathrm{T}} \end{bmatrix} + \begin{bmatrix} \epsilon_{\mathrm{F}} \\ \epsilon_{\mathrm{C}} \\ \epsilon_{\mathrm{T}} \end{bmatrix} \tag{3.4}$$

Due to the conservation of mass, this system must be constrained such that $\beta_{\mathrm{Feed}} = \beta_{\mathrm{Concentrate}} + \beta_{\mathrm{Tailings}}$. To constrain the system, the true masses should be represented using a vector $\boldsymbol{\beta}$ with two elements, as shown in Equations (3.5) and (3.6).

$$\begin{bmatrix} y_{\mathrm{F}} \\ y_{\mathrm{C}} \\ y_{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} \beta_{\mathrm{C}} + \beta_{\mathrm{T}} \\ \beta_{\mathrm{C}} \\ \beta_{\mathrm{T}} \end{bmatrix} + \begin{bmatrix} \epsilon_{\mathrm{F}} \\ \epsilon_{\mathrm{C}} \\ \epsilon_{\mathrm{T}} \end{bmatrix} \tag{3.5}$$

$$= \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \beta_{\mathrm{C}} \\ \beta_{\mathrm{T}} \end{bmatrix} + \begin{bmatrix} \epsilon_{\mathrm{F}} \\ \epsilon_{\mathrm{C}} \\ \epsilon_{\mathrm{T}} \end{bmatrix} \tag{3.6}$$

Equation (3.7) shows Equation (3.6) with general terms, where $\boldsymbol{X}_g$ maps values in $\boldsymbol{\beta}$ to values of $\boldsymbol{y}$.

$$\boldsymbol{y} = \boldsymbol{X}_g\boldsymbol{\beta} + \epsilon \tag{3.7}$$

Real plant operations have several nodes, often interconnected in complex configurations. Equation (3.7) can be used to model and constrain a more complex circuit by specifying linear constraints. Constraints for the two node process in Figure 3.1 are listed as Equations (3.8). This process has one input, three outputs and five sampling locations. To simplify notation, the remainder of this document will use y_j to stipulate the mass flow rate observed at sample location j of N total sample locations.

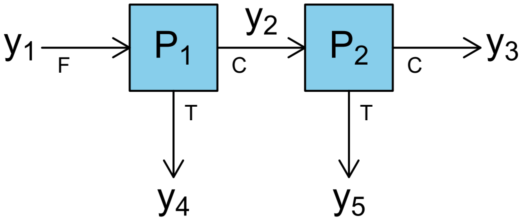

Figure 3.1: Example multi-node circuit

$$\begin{aligned}\beta_1 &= \beta_2 + \beta_4 \\ \beta_2 &= \beta_3 + \beta_5\end{aligned} \tag{3.8}$$

The constraints in Equations (3.8) constraints can be indicated in the five column, two row matrix $\boldsymbol{C}$, shown in Equation (3.9), where the columns index each $\boldsymbol{\beta}$, and the rows index each constraint. Note, $\boldsymbol{C\beta} = 0$. Gauss-Jordan elimination can be used to find the reduced row echelon form, $\boldsymbol{C}_R$ as in Equation (3.10).

$$\boldsymbol{C} = \begin{bmatrix} 1 & -1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 & -1 \end{bmatrix} \tag{3.9}$$

$$\boldsymbol{C}_R = \begin{bmatrix} 1 & 0 & -1 & -1 & -1 \\ 0 & 1 & -1 & 0 & -1 \end{bmatrix} \tag{3.10}$$

Inspecting the reduced row echelon form reveals which elements of $\boldsymbol{\beta}$ are free. Importantly, $\boldsymbol{C}_R\boldsymbol{\beta} = 0$ holds. For a row in $\boldsymbol{C}_R$ the $\boldsymbol{\beta}$ indexed by the location of a positive 1, can be substituted by the corresponding elements of $\boldsymbol{\beta}$ indexed by the presence of -1. The resulting constrained model, built from $\boldsymbol{C}_R$, is shown in Equation (3.11). Note, the remaining elements of $\boldsymbol{\beta}$ are the process outputs. More details on similar procedures can be found in Madron (1992).

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \beta_3 \\ \beta_4 \\ \beta_5 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \end{bmatrix}$$

$$\boldsymbol{y} = \boldsymbol{X}_g \boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{3.11}$$

Usually data reconciliation is required for multiple sample components. A component could be a mineral such as $CuFeS_2$, an element of interest, or a generic gangue material. To include multiple sample components in the model the matrix $\boldsymbol{X}$ is written as a block diagonal matrix equal to the Kronecker product $\boldsymbol{I}_M \otimes \boldsymbol{X}_g$, where M is the number of sample components and $\boldsymbol{I}_M$ is an $M \times M$ identity matrix. i indexes the M components of a sample, and Q is the dimension of the constrained $\boldsymbol{\beta}$. The structure of $\boldsymbol{X}$ for a two component model is shown in (3.12).

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_g & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{X}_g \end{bmatrix}_{(MN)\times(MQ)} \tag{3.12}$$

Lastly, the relationship between the elements of $\boldsymbol{\epsilon}$ must be considered. A constant variance for all sample components at all sampling locations is not realistic (Wills 2006). Removing a constant variance assumption requires multiple sample sets to be obtained. Let K be the number of sample sets. An individual sample set k is made up of samples taken at all N locations simultaneously. A subsequent sample set, $k+1$ is taken enough time apart from set k that collection of k does not interfere with the observation $k+1$. If variation in the process over time is not of concern, an alternate method for obtaining K sample sets would would be to take one sample set and make representative splits.

Let $\boldsymbol{\Omega}$ be the covariance between the observations in sample set $\boldsymbol{y}_k$. For a model where each $\boldsymbol{y}_{i,j,\cdot}$ has independent variance, $\boldsymbol{\Omega} = diag(\sigma^2_{1,1}, \ldots, \sigma^2_{i,j}, \ldots, \sigma^2_{M,N})$. This specification indicates no error correlation between sample locations or components. The resulting covariance matrix is shown in Equation (3.13). Bayesian models using the error structure in (3.13) will be referred to as the *independent variance model* for the remainder of this text.

$$\boldsymbol{\Omega} = \begin{bmatrix} \sigma^2_{1,1} & & & & \\ & \ddots & & 0 & \\ & & \sigma^2_{i,j} & & \\ & 0 & & \ddots & \\ & & & & \sigma^2_{M,N} \end{bmatrix}_{NM\times NM} \tag{3.13}$$

Error correlation is achieved by allowing off diagonal elements of $\boldsymbol{\Omega}$ to be non zero. A model allowing for correlated error may be a better fit for data, but requires more degrees of freedom for parameter estimation. Let the covariance matrix $\mathbf{\Sigma}_i$ be the covariance of the mass flow rates of a sample component between different locations. $\mathbf{\Sigma}_i$ is the covariance of $\boldsymbol{y}_i$. The covariance structure for observation $\boldsymbol{y}_k$ is shown in Equation (3.14).

$$\mathbf{\Omega} = \begin{bmatrix} \mathbf{\Sigma}_1 & & & & \\ & \ddots & & 0 & \\ & & \mathbf{\Sigma}_i & & \\ & 0 & & \ddots & \\ & & & & \mathbf{\Sigma}_M \end{bmatrix}_{NM \times NM} \tag{3.14}$$

It is possible to specify other covariance structures such as correlation between sample components at a given location. Complicated custom error structures can also be specified. For simplicity, only the correlation as specified in (3.14) is examined. For the remainder of this text, a Bayesian model allowing for error correlated between mass flow rates of an individual component will be referred to as the *covariance model*.

3.2.2 Bayesian Inference

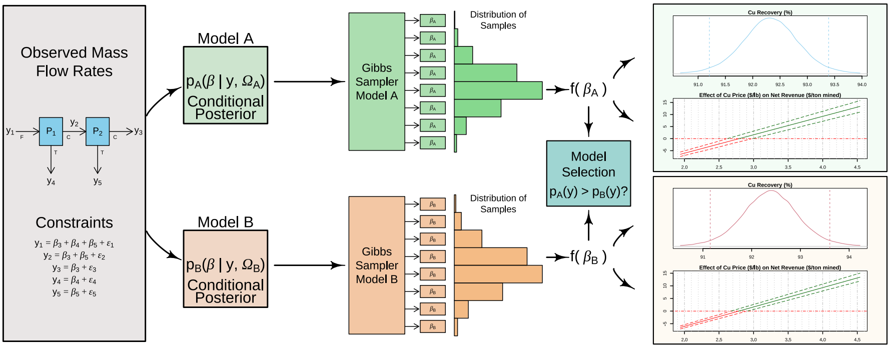
Observed Mass Flow Rates
y_1
F
P_1
y_2
C
P_2
C
y_3
T
T
y_4
y_5
Constraints
$y_1 = \beta_3 + \beta_4 + \beta_5 + \varepsilon_1$
$y_2 = \beta_3 + \beta_5 + \varepsilon_2$
$y_3 = \beta_3 + \varepsilon_3$
$y_4 = \beta_4 + \varepsilon_4$
$y_5 = \beta_5 + \varepsilon_5$
Model A
$p_A(\beta \mid y, \Omega_A)$ Conditional Posterior
Model B
$p_B(\beta \mid y, \Omega_B)$ Conditional Posterior
Gibbs Sampler Model A
Gibbs Sampler Model B
β_A
β_B
Distribution of Samples
Distribution of Samples
$f(\beta_A)$
$f(\beta_B)$
Model Selection
$p_A(y) > p_B(y)$?
Cu Recovery (%)
91.0
91.5
92.0
92.5
93.0
93.5
94.0
Effect of Cu Price ($/lb) on Net Revenue ($/ton mined)
-5
0
5
10
15
2.0
2.5
3.0
3.5
4.0
4.5
Cu Recovery (%)
91
92
93
94
Effect of Cu Price ($/lb) on Net Revenue ($/ton mined)
-5
0
5
10
15
2.0
2.5
3.0
3.5
4.0
4.5

Figure 3.2: Structural outline of applied Bayesian data reconciliation.

Bayesian inference of the model parameters is governed by Bayes' rule (3.15). However, $p(\boldsymbol{y})$ is a constant, so in Bayesian inference, Bayes' rule is often reduced to (3.16).

$$p(\boldsymbol{\beta}, \mathbf{\Omega}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\beta}, \mathbf{\Omega})p(\boldsymbol{\beta}, \mathbf{\Omega})}{p(\boldsymbol{y})} \tag{3.15}$$

$$p(\boldsymbol{\beta}, \mathbf{\Omega}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{\beta}, \mathbf{\Omega})p(\boldsymbol{\beta}, \mathbf{\Omega}) \tag{3.16}$$

Equation (3.16) can be read as, *the posterior distribution of beta and omega given the data observed, is proportional to the probability of observing the data,* *given omega and beta, times the prior probability of beta and omega.* $p(y|\boldsymbol{\beta}, \boldsymbol{\Omega})$ is also known as the likelihood function, and can be written as $\mathcal{L}(\boldsymbol{\beta}, \boldsymbol{\Omega}|y)$. Through Bayes rule, prior beliefs about model parameters are updated with observed data to produce the probability distribution of model parameters *a posteriori*, or after, observing data. More on the details of Bayesian inference can be found in Hoff (2009).

While Bayes' rule is fundamental to Bayesian inference, applications often require a mixture of analytical derivations and computational approximations of intractable integrals. Figure 3.2 gives a structural outline of how Bayesian inference is applied to data reconciliation for separations. After obtaining the data and specifying constraints, a conditional posterior distribution for each model parameter is derived using Bayes' rule (§3.2.2.1). The conditional posterior distributions are used with a Gibbs Sampler; an algorithm which iteratively obtains a large number of samples, or draws, from the marginal posterior distribution of each model parameter (§3.2.2.2). These draws can be simply binned into a histogram to visualize the form of the distributions. More intricate analysis can be conducted using the obtained samples, including applying a function to each draw obtaining draws from the function output, model selection (§3.2.2.3) and economic analysis applications (§3.2.3).

3.2.2.1 Conditional Posterior Distribution Derivation

To derive the conditional posterior distribution for model parameters $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}_i$ or $\sigma^2_{i,j}$, prior distributions first need to be specified. The prior distribution for a parameter is then multiplied by the likelihood function, conditioned on the other parameters. Working in proportionality allows for dropping constant terms, and the resulting expression can be recognized as the kernel of a conditional posterior distribution. The kernel of a distribution is the elements of a distribution function without its normalizing constants. More details can be found in Hoff (2009).

To use the methods in §3.2.2.3 the prior distributions are required to have a known distributional form. Conditionally conjugate priors are used, where a conjugate prior for a parameter has the same distributional form as its posterior distribution. Sometimes reference priors, containing little information, are desired. More on such priors can be found in Kass and Wasserman (1996) and Yang and Berger (1996), however these priors are not compatible with all the methods used in this text. Conjugate priors usually influence posterior inference more than reference priors, but prior distribution hyperparameter can be specified to give little influence over the posterior distribution.

The conditional conjugate prior for $\boldsymbol{\beta}$ is a truncated normal distribution with a mean of $\boldsymbol{\mu}_0$ and a covariance of $\boldsymbol{V}_0$ with a left truncation bound at 0. The prior belief that $p(\boldsymbol{\beta})$ is bounded at 0 is stipulated, because even before viewing any data, one can be sure negative mass is not relevant to this application. To minimize influence of the values chosen for $\boldsymbol{\mu}_0$ the diagonal elements of $\boldsymbol{V}_0$ can be specified to be large, flattening the prior distribution of $\boldsymbol{\beta}$.

For the independent variance model a prior distribution is placed on each $\sigma^2_{i,j}$. The conjugate prior used for the variance is an inverse Gamma distribution with hyper-parameters α_0 and β_0. α_0 and β_0.

In the covariance model, $p(\mathbf{\Sigma}_i)$ is specified to be an inverse Wishart distribution (Hoff 2009) with hyper-parameters of the scale matrix $\boldsymbol{S}_0$, and the degrees of freedom ν_0. Specifying the hyper-parameter values for $p(\mathbf{\Sigma}_i)$ can be confusing for this high dimensional distribution. It is necessary to specify $\nu_0 > N - 1$, specifying $\nu_0 = N$ will cause the prior distribution to have less influence over the posterior distribution compared to higher values. In Gelfand et al. (1990) an equivalent scale matrix was specified by a diagonal matrix containing rough estimates of the variance multiplied by the prior degrees of freedom. Using data to specify prior distribution hyper-parameters is known as Empirical Bayes, some discussion on the topic can be found in Kass and Steffey (1989).

Appendix A.1 gives details for the derivation of conditional posterior distributions. $p(\boldsymbol{\beta}|\mathbf{\Omega}, \boldsymbol{X}, \boldsymbol{y})$ is derived in Appendix A.1.2 as a truncated multivariate normal distribution $\mathcal{N}_0(\hat{\boldsymbol{\beta}}, \boldsymbol{V})$, where $\hat{\boldsymbol{\beta}} = \boldsymbol{V}(\boldsymbol{V}_0^{-1}\boldsymbol{\mu}_0 + K\boldsymbol{X}^T\mathbf{\Omega}^{-1}\bar{\boldsymbol{y}}_{i,j})$, $\boldsymbol{V} = (\boldsymbol{V}_0^{-1} + K\boldsymbol{X}^T\mathbf{\Omega}^{-1}\boldsymbol{X})^{-1}$, $\bar{\boldsymbol{y}}_{i,j}$ is the a vector containing the mean of each $\boldsymbol{y}_{i,j,\cdot}$, and $\boldsymbol{X}$ is $\boldsymbol{I}_M \otimes \boldsymbol{X}_g$, the same form used to create the matrix in (3.12).

For the independent variance model (3.13), the conditional posterior distribution for each $\sigma^2_{i,j}$ is invGamma $\left(\frac{K}{2} + \alpha_0, \frac{1}{2}\sum_{k=1}^{K}(y_{i,j,k} - x^T_{i,j}\boldsymbol{\beta})^2 + \beta_0\right)$, where $x^T_{i,j}$ is the row in $\boldsymbol{X}$ that maps $\boldsymbol{\beta}$ to $y_{i,j}$. Derivation is shown in Appendix A.1.3.

When using the covariance model, with the covariance structure of (3.14), the distribution $p(\mathbf{\Sigma}_i|\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{y})$ is $\mathcal{W}^{-1}(\boldsymbol{S}, \nu_0 + K)$. $\boldsymbol{S}$ is equal to $\left(\sum_{k=1}^{K}(\boldsymbol{y}_{i,k} - \boldsymbol{X}_i\boldsymbol{\beta})(\boldsymbol{y}_{i,k} - \boldsymbol{X}_i\boldsymbol{\beta})^T\right) + \boldsymbol{S}_0$. Derivation of this inverse Wishart conditional posterior distribution is shown in Appendix A.1.4.

3.2.2.2 The Gibbs Sampler

Algorithm 3.2.1: Gibbs Sampler for Error Structure in (3.14)

```
Result: Draws from the marginal distributions of $\boldsymbol{\beta}$, $\mathbf{\Sigma}_1, ..., \mathbf{\Sigma}_M$
initialization: $B$, $T$, $\boldsymbol{\beta}^{(1)}$, $\mathbf{\Sigma}_1^{(1)}, \dots, \mathbf{\Sigma}_M^{(1)}$;
for $t = 2, 3, ..., T$ do
    $\mathbf{\Omega} \leftarrow \text{BlockDiagonal}(\mathbf{\Sigma}_1^{(t-1)}, ..., \mathbf{\Sigma}_M^{(t-1)})$;
    $\boldsymbol{\beta}^{(t)} \sim \mathcal{N}_0((\boldsymbol{V}_0^{-1} + K\boldsymbol{X}^T\mathbf{\Omega}^{-1}\boldsymbol{X})^{-1}(\boldsymbol{V}_0^{-1}\mu_0 + K\boldsymbol{X}^T\mathbf{\Omega}^{-1}\bar{\boldsymbol{y}}_{i,j}), (\boldsymbol{V}_0^{-1} +$
      $K\boldsymbol{X}^T\mathbf{\Omega}^{-1}\boldsymbol{X})^{-1})$;
    for $i = 1, ..., M$ do
        $\mathbf{\Sigma}_i^{(t)} \sim$
          $\mathcal{W}^{-1}\left((\sum_{k=1}^{K}(\boldsymbol{y}_{i,k} - \boldsymbol{X}_i\boldsymbol{\beta}^{(t)})(\boldsymbol{y}_{i,k} - \boldsymbol{X}_i\boldsymbol{\beta}^{(t)})^T) + \boldsymbol{S}_0, \nu_0 + K\right)$;
    end
end
Save samples from $t = B + 1, B + 2, ..., T$
```

It is worth restating, the goal is to draw samples from the marginal distribution of $p(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{y})$ and each $p(\boldsymbol{\Sigma}_i|\boldsymbol{X}, \boldsymbol{y})$ or $p(\sigma^2_{i,j}|\boldsymbol{X}, \boldsymbol{y})$. A Gibbs sampler is a Markov-Chain Monte-Carlo (MCMC) method which generates random samples from the marginal posterior distributions.

To implement a Gibbs sampler, first a value for each model parameter must be initialized with a semi-arbitrary value. Initialization values can have an effect on convergence time. Then iterative draws of each parameter are taken conditional on the previous draw of the other parameters, using the conditional distributions specified §3.2.2.1. Iterating over the conditional distributions is analogous to averaging, or integrating, over the conditional model parameters. Some iterations after initialization, the Markov-Chain converges on the target distribution. The initial draws before convergence are discarded as *Burn-in* iterations. If the algorithm has converged properly, samples after $t = B$ are from the marginal target distributions. These random samples can be directly used to calculate recovery and grade, generating samples from the probability distribution for both of these metrics. Steps for the covariance model are listed in Algorithm 3.2.1, while the Gibbs sampler for the independent variance model replaces $\boldsymbol{\Sigma}_i$ with $\sigma^2_{i,j}$. More detail on Gibbs samplers can be found in Casella and George (1992) and Gelfand et al. (1990).

When using an MCMC tool such as a Gibbs sampler, there is some time before convergence and there will be some autocorrelation between sequential draws. Ideally, sequential draws from the marginal posterior distribution are independent, and have no correlation. Statistical tools have been developed for checking convergence and if autocorrelation is at an acceptable level. A calculation for effective sample size can be used to quantify the auto-correlation observed in the draws (Liu and Chen 1995). The closer the effective sample size is to the number of samples taken, the less auto-correlation there is between the samples.

The test for convergence used is the CD score (Geweke et al. 1991). This score blocks the samples taken and compares the mean of each block. If the Markov chain has converged, the difference in the means weighted by their standard error can be simulated from from an asymptotic standard normal distribution. The resulting score is interpreted similarly to a Z-score, where if values are observed greater than 1.95 standard deviations apart, they may not necessarily come from the same distribution. CD scores and effective sample size are computed using the `geweke.diag()` and `effectiveSize()` functions, respectively, from the `coda` V 0.19.4 package (Plummer et al. 2006).

A less quantitative method of observing the independence of subsequent draws is to plot the values of each draw for a parameter sequentially. A plot of this style, made from draws that have low auto-correlation and have converged, will look more like a fuzzy caterpillar than a snake.

3.2.2.3 Model Selection

"All models are wrong, but some are useful," is a commonly stated excerpt from Box (1976). Model selection aims to objectively determine which model, out of a group of models considered, is most useful, or best represents the data.

One method for Bayesian model selection is the calculation of a Bayes factor (Kass and Raftery 1995). For a given model M_l, the likelihood of observing the data collected, given the model is true, is called the marginal likelihood, $p(y|M_l)$. For two models $\{M_1, M_2\}$, a Bayes factor, shown in Equation (3.17), is the ratio of two marginal likelihoods, and is used to compare the models. To improve numerical stability, a logarithm of the Bayes Factor is used and reported (Equation (3.18)). This publication and the `BayesMassBal` package use $\log_e$.

$$BF = \frac{p(\boldsymbol{y}|M_1)}{p(\boldsymbol{y}|M_2)} \tag{3.17}$$

$$\log_e(BF) = \log_e(p(\boldsymbol{y}|M_1)) - \log_e(p(\boldsymbol{y}|M_2)) \tag{3.18}$$

There are a few methods to calculate or approximate the marginal likelihood. Since a Gibbs sampler and conditional conjugate priors are used, methods in Chib (1995) are appropriate. The approximation in Chib (1995) hinges on the marginal likelihood identity, or that Equation (3.19) holds for any value of θ, making Equation (3.20) true.

$$p(y|M_l) = \frac{p(y|\theta, M_l)p(\theta|M_l)}{p(\theta|y, M_l)} \tag{3.19}$$

$$= \frac{p(y|\bar{\theta}, M_l)p(\bar{\theta}|M_l)}{p(\bar{\theta}|y, M_l)} \tag{3.20}$$

For a one parameter model, it is possible to simply evaluate each of the densities on the right hand side of Equation (3.20). For a two parameter model, with parameters $\theta = \{\theta_1, \theta_2\}$, it is necessary to break down the denominator in (3.20) further, as shown in Equation (3.21).

$$\begin{aligned} &p(\bar{\theta}|y, M_l) \\ &= p(\bar{\theta}_2|\bar{\theta}_1, y, M_l)p(\bar{\theta}_1|y, M_l) \\ &= p(\bar{\theta}_2|\bar{\theta}_1, y, M_l)\int p(\bar{\theta}_1|\theta_2, y, M_l)p(\theta_2|y, M_l)d\theta_2 \end{aligned} \tag{3.21}$$

To solve the integral, the Gibbs sampler output is used along with Monte Carlo integration methods (Chib 1995; Metropolis and Ulam 1949) by the approximation shown in Equation (3.22). The rest of the densities are simply evaluated for values of $\bar{\theta}$.

$$p(\bar{\theta}_1|y, M_l) \approx \frac{1}{T-B} \sum_{t=B+1}^{T} p(\bar{\theta}_1|\theta_2^{(t)}, y, M_l) \tag{3.22}$$

When implementing the models in §3.2.1 most applications will have more than two model parameters. While Chib (1995) gives methods for computing the marginal likelihood for models with more than two parameters, the independence of some of the model parameters can be exploited to avoid this treatment and reduce computation time. For the covariance model, each $\boldsymbol{\Sigma}_i$ is independent. The joint posterior density can be broken down and then approximated as shown in (3.23).

$$\begin{aligned}
&p(\bar{\boldsymbol{\beta}}, \bar{\boldsymbol{\Sigma}}_1, \ldots, \bar{\boldsymbol{\Sigma}}_i, \ldots, \bar{\boldsymbol{\Sigma}}_M|\boldsymbol{y}) \\
&= p(\bar{\boldsymbol{\Sigma}}_M|\bar{\boldsymbol{\Sigma}}_{M-1}, \ldots, \bar{\boldsymbol{\Sigma}}_1, \bar{\boldsymbol{\beta}}, \boldsymbol{y}) \ldots p(\bar{\boldsymbol{\Sigma}}_1|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) p(\bar{\boldsymbol{\beta}}|\boldsymbol{y}) \\
&= p(\bar{\boldsymbol{\Sigma}}_M|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) \ldots p(\bar{\boldsymbol{\Sigma}}_1|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) p(\bar{\beta}|\boldsymbol{y}) \\
&= p(\bar{\boldsymbol{\Sigma}}_M|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) \ldots p(\bar{\boldsymbol{\Sigma}}_1|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) \\
&\qquad \times \int \cdots \int p(\bar{\beta}|\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_M, \boldsymbol{y}) d\boldsymbol{\Sigma}_1 \ldots d\boldsymbol{\Sigma}_M \\
&\approx p(\bar{\boldsymbol{\Sigma}}_M|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) \ldots p(\bar{\boldsymbol{\Sigma}}_1|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) \\
&\qquad \times \frac{1}{T-B} \sum_{t=B+1}^{T} p(\bar{\boldsymbol{\beta}}|\boldsymbol{\Sigma}_1^{(t)}, \ldots, \boldsymbol{\Sigma}_M^{(t)}, \boldsymbol{y})
\end{aligned} \tag{3.23}$$

For the independent variance model the approximation is shown as Equation (3.24).

$$\begin{aligned}
&p(\bar{\sigma}_{M,N}^2|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) \ldots p(\bar{\sigma}_{1,1}^2|\bar{\boldsymbol{\beta}}, \boldsymbol{y}) \\
&\qquad \times \frac{1}{T-B} \sum_{t=B+1}^{T} p(\bar{\boldsymbol{\beta}}|(\sigma_{1,1}^2)^{(t)} \ldots (\sigma_{M,N}^2)^{(t)}, \boldsymbol{y})
\end{aligned} \tag{3.24}$$

3.2.2.4 HPDI

The highest posterior density interval (M.-H. Chen and Shao 1999), or HPDI is used, to quantify the uncertainty of posterior distributions using the samples obtained from the Gibbs sampler. For a uni-modal distribution, the HPDI is the narrowest interval of the probability distribution containing 95% of the probability mass. If samples are taken from a population with an unknown mean and a known distributional form, the 95% HPDI of a posterior distribution from an ideal model would contain the true mean 95% of the time. The HPDI is a relatable measure of uncertainty of the metrics in question. It is used to aid in financial analysis and assess model validity in this publication.

3.2.3 Main Effects

For a function with independent inputs $f(x_1, x_2, \ldots, x_k)$, Saltelli (2002) gives the equation to estimate the sensitivity of $f(x)$ with respect to x_j as in (3.25). The function being studied could be something along the lines of the net revenue for the mine and, the independent input could be something like copper price.

$$S_j = \frac{\mathrm{ar}(\ \ [f(x)|x_j])}{\mathrm{ar}(f(x))} \tag{3.25}$$

To visualize how the value of a function will vary with respect to an independent input, the main effect can be plotted. The main effect, only part of (3.25), is $\ [f(x)|x_j]$. To find the main effect, the integral, given in Equation (3.26) (Saltelli 2002), must be solved.

$$\begin{aligned} &{}_x[f(x)|x_j] = \\ &\int\int\cdots\int f(x_1, \ldots, x_j = \tilde{x}_j, \ldots, x_k) \prod_{\substack{i=1 \\ i\neq j}}^{k} p(x_i)dx_i \end{aligned} \tag{3.26}$$

To observe the main effect of x_j, while taking into account process uncertainty, $f(\cdot)$ must be related to the output of a data reconciliation model. The balanced mass flow rates, $\boldsymbol{y}_{\mathrm{bal}}$, are included as $f(x, y)$. Then the expectation with respect to x is taken, conditioned on x_j and $\boldsymbol{y}_{\mathrm{bal}}$. The integral required to find this expectation is shown as as Equation (3.27). For Equation (3.27) to be valid, x and $\boldsymbol{y}_{\mathrm{bal}}$ must be independent.

$$\begin{aligned} &{}_x[f(x, \boldsymbol{y}_{\mathrm{bal}})|x_j, \boldsymbol{y}_{\mathrm{bal}}] = \\ &\quad \int\cdots\int f(x_1, \ldots, x_j = \tilde{x}_j, \ldots, \boldsymbol{y}_{\mathrm{bal}} = \tilde{\boldsymbol{y}}_{\mathrm{bal}}) \\ &\quad\quad \times \prod_{\substack{i=1 \\ i\neq j}}^{k} p(x_i)dx_i \end{aligned} \tag{3.27}$$

This integral can be approximated using Monte-Carlo integration methods (Metropolis and Ulam 1949), by supplying values of x_j and $\boldsymbol{y}_{\mathrm{bal}}$. Using draws from the marginal distribution of $\boldsymbol{y}_{\mathrm{bal}}$ it is possible to iteratively approximate ${}_{x|\boldsymbol{y}_{\mathrm{bal}}}[f(x, \boldsymbol{y}_{\mathrm{bal}})|x_j, \boldsymbol{y}_{\mathrm{bal}}]$ over each draw of $\boldsymbol{y}_{\mathrm{bal}}$, marginalizing over $\boldsymbol{y}_{\mathrm{bal}}$, and producing draws from the distribution of ${}_x[f(x, \boldsymbol{y}_{\mathrm{bal}})|x_j]$. These resulting samples show the uncertainty of the expected value of $f(x, \boldsymbol{y}_{\mathrm{bal}})$, specifically related to process uncertainty.

The implementation steps used are mostly adapted from the main effect algorithm for an unrelated application in Gramacy (2020). An implementation, where each x is independent and uniformly distributed, can take advantage of Latin hypercube sampling (LHS) (McKay, Beckman, and Conover 1979) which has favorable properties for approximating integrals.

3.2.4 `BayesMassBal` Package

The `BayesMassBal` Package (Koermer 2020a) was built to make Bayesian data reconciliation methods more accessible, as well as make the results of this paper easy to reproduce. Functions are available that implement many of the procedures outlined in Methods (§3.2) and performed in Experimental (§3.3). The `importObservations()` function can be used to import and organize observations for the models in §3.2.1. The function `constrainProcess()` takes a matrix of linear constraints, as specified in (3.9), and produces the matrix $\boldsymbol{X}_g$ as in Equation (3.11).

Methods in §3.2.2.2 and §3.2.2.3 are implemented in the `BMB()` function. The argument `BTE = c(B,T,E)` is a numeric vector specifying the number of **B**urn-in iterations, **T**otal iterations, and that **E**very E^{th} sample is saved. $\text{E} > 1$ cuts down on auto-correlation between consecutive samples at the expense of computation time.

Error structure is selected using the `cov.structure` argument. `cov.structure = "indep"` selects the independent variance model. The covariance model in this publication can be selected by setting the argument `cov.structure = "component"`.

The default prior hyperparameter settings specified by the `BMB()` function were used for all inference in this publication. The default mean of $p(\boldsymbol{\beta})$ is set to the ordinary least squares estimate $\hat{\boldsymbol{\beta}}_{\text{OLS}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\bar{\boldsymbol{y}}$. The prior variance of each β_q is equal to 10 to the power of the number of integer digits in $\hat{\beta}_{\text{OLS},q} + 6$. For the independent variance model a default prior distribution of $\text{invGamma}(0.000001, 0.000001)$ on each $\sigma^2_{i,j}$ is specified. This prior is fairly flat, but is still informative (Gelman et al. 2006). For the covariance model a prior distribution of $\mathcal{W}^{-1}(N, N \times S_{0,i})$ was used for each $\boldsymbol{\Sigma}_i$ where $S_{0,i}$ is a diagonal matrix with the sample variance of each $\boldsymbol{y}_{i,j}$ as each element. Allowing the prior distributions to be informed by the data allows for the default settings of `BMB()` function to better adapt to general use cases. However, a more pure Bayesian approach to Bayesian data reconciliation can be taken by specifying the `priors` argument for `BMB()` as a list of hyperparameter settings. When implementing these methods on a complex circuit with large amounts of correlation between nodes or sample components, a large data set may be required to reduce the influence of the prior distributions. If a small data set is used, for example a data set which adds a number degrees of freedom less than the dimension of $\boldsymbol{\Sigma}_i + 1$, the user should specify hyperparameters for the prior distributions, perhaps based on their intuition or old data. See the package documentation for the `priors` argument of `BMB()` (Koermer 2020a) for more information.

Setting `BMB(...,lml = TRUE,...)` indicates the log-marginal likelihood should be approximated.

The default argument setting, `diagnostics = TRUE`, returns the CD score and effective sample size for each element in $\boldsymbol{\beta}$, and $\boldsymbol{\Omega}$. The trace plots discussed at the end of §3.2.2.2 can be created by feeding the output from `BMB()` to the `plot()` function by specifying `plot(...,layout = "trace")`.

The main effect can be computed as outlined in §3.2.3 by passing the output of the `BMB()` function to the `mainEff()` function and supplying code for $f(x, \boldsymbol{y}_{\text{bal}})$. See Appendix A.5 or package documentation for details.

The simulated data used in §3.3.1 is obtained using the default arguments of the `twonodeSim()` function. This function simulates seven data sets from the two node process in Figure 3.1 with sample components of $CuFeS_2$ and gangue. Each data set is independent and identically distributed. There are three sources of stochasticity, variability in feed rate, variation in process performance, and independent assay noise. These sources induce both independent and correlated errors. See `help("twonodeSim")` and the package source code for more details.

The `pointmassbal()` function implements a point estimate mass balance, adapted from Wills (2006) and derived in Appendix A.2, for a two node, two component process. This function was used for the applications in §3.3.1.1 and 3.3.1.2 where the Bayesian models are compared to a point estimate model. To allow for easy inspection of the source code, the function was included with the `BMB` package.

3.3 Experimental

This section details experiments conducted to test the usefulness and accuracy the Bayesian data reconciliation models. Section 3.3.1 outlines a simulation study which explores the use cases for the output of the Bayesian models for plotting posterior distributions and main effects, as well as model selection (§3.3.1.1). An advantage of testing a statistical model on simulated data is the ability to check the statistical model against the known expected output of a simulation. Since the expected output of the model is known, simulated data was used to compare the accuracy of the Bayesian models to a point estimate model. Derivation for the point estimate model is shown in Appendix A.2. A Monte Carlo experiment where every model was fit to each of 1,000 data sets was completed for this comparison in §3.3.1.2. Section 3.4 tests the application of the Bayesian models on real data.

3.3.1 Application on Simulated Data

3.3.1.1 Model Implementation

A data set was generated and is listed in Table A.1 in Appendix A.3. Linear constraints were specified and `constrainProcess()` was used to find the `X` matrix required for `BMB()`. Then, two calls to `BMB()` were used, one for each error structure stipulated in §3.2.1. Default prior settings were used and `lml=TRUE` was specified. For each function call, 100,000 samples were collected after 10,000 burn in samples were removed and every other sample was thinned from an initial 210,000 samples.

Sample code for the data simulation, specifying process constraints, and running the Bayesian data reconciliation models is shown in Appendix A.5.

The output for both Bayesian models was checked for convergence using the CD score and for auto-correlation by calculating the effective sample size. A summary of the worst values observed is included as Table A.2 in Appendix A.4.

The point mass balance model derived in Appendix A.2 was fit to the same data. The mean observed mass feed rate into the plant for the sample set was used for calculations involving the point mass balance model.

Table 3.1: Values used in financial and sensitivity calculations.

Parameter	Current Value	Min	Max
Milling and Mining ($/ton Ore)	5	4	6
Processing Cost ($/hour)	1,500	1,125	1,875
Copper Price ($/ton Cu)	6,000	3,880	9,080
Treatment Cost ($/ton Concentrate)	40	20	60
Refining Cost ($/ton Cu)	160	96	208
Freight ($/ton Concentrate)	25	20	60

The usefulness of the output from the Bayesian models was explored by plotting posterior densities of outputs and metrics calculated using the obtained samples. Figure 3.4 shows these densities along with 95% HPDI bounds, outcomes from the point estimate mass balance, and calculations from the expected value of the simulation.

The posterior density of net return from smelter (NSR) (Wills 2006) is one of the densities plotted, illustrating the utility of uncertainty quantification in financial analysis. NSR calculates revenue per ton of feed ore, and requires many of the mass balanced flow rates. The calculation is related to commodity prices, freight, treatment, and refining. The formula for calculating NSR is shown in Equation (3.28), and the values used for Figure 3.4 and §3.3.1.2 can be found under the *Current Value* column in Table 3.1.

$$\text{NSR} = \frac{Cc \times 0.346}{F} \times (6000 - 160) - \frac{C}{F} \times 40 \tag{3.28}$$

Processing cost per ton of ore mined is related to the uncertainty regarding the true plant feed rate. In real life uncertainty in processing cost can be related to mis-calibrated belt scales and improperly quantified feed surges. This density was found by taking the *Current Value* hourly plant operating cost given in Table 3.1 and dividing it by the combined mass flows for total plant feed. The density for net revenue was found by setting the mining cost per ton to a constant and subtracting the processing and mining cost from NSR.

After using the `BMB()` function for both Bayesian models, a Bayes Factor was used for evidence that one model is a better fit to the data than another.

Samples from the covariance model were then used to plot the main effect of copper price on net revenue. This was implemented by passing the output from `BMB()` to the `mainEff()` function along with a user specified function that calculates net revenue using the reconciled data. The code for the function calculating net revenue is included in Appendix A.5. See the `BayesMassBal` package documentation for more information on implementation. The output from the `mainEff()` function was used to generate Figure 3.5, to illustrate how the output from a Bayesian model can be used to gain a better understanding of how process uncertainty effects the bottom line.

3.3.1.2 Model Validation

Figure 3.4 allows for the results of the data reconciliation models to be compared. However, these specific results are dependent on the single data set used to generate them. Since there is variation in the data obtained from a stochastic process simulator, there will be variation in the results from each model between data sets. Little can be concluded from fitting the models to one data set. If performance between models is to be compared, the comparison must be completed over a large number of data sets.

To observe average behavior, a Monte Carlo experiment was run. The `twonodeSim()` function with default arguments was used to generate 1,000 sets of 7 observations. For each data set, both Bayesian models and the point estimate model were used to reconcile data using the same methods as §3.3.1.1. Using the reconciled data from each model, net revenue per ton processed was estimated. Net revenue was chosen for the comparison because its calculation relies on multiple estimates from the data reconciliation models. For the point estimate model, the estimate for net revenue was recorded. For each Bayesian model, the posterior mean and 95% HPDI was recorded.

3.4 Application on Real Data

A disadvantage of using a simulation to test a model is that it is not always clear that the model can be used on real data. Sometimes models require assumptions that make use on real data unreasonable. For the Bayesian models in this paper, the steps used with real data are indistinguishable from the steps required for use with simulated data.

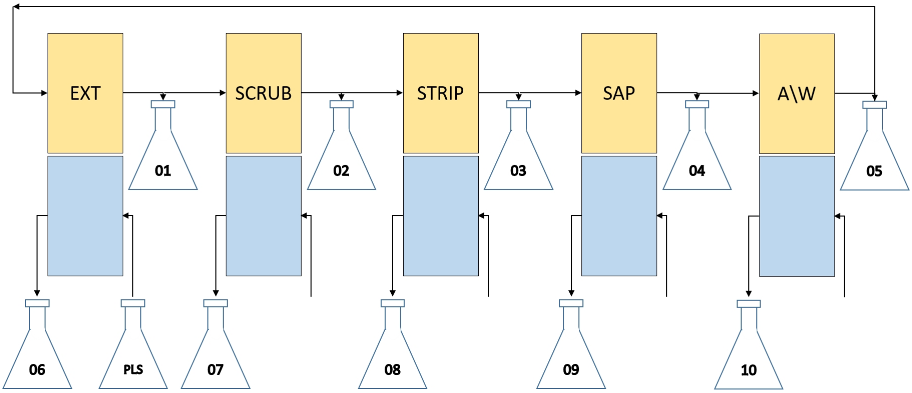

Figure 3.3: Flowchart of SX system used for data collection

Data was obtained from a pilot scale liquid-liquid extraction system for rare earth elements. The total rare earth element (TREE) mass flow rates needed to be reconciled, and could then be used to calculate concentrations. Figure 3.3 is an illustration of the pilot plant. The lighter yellow areas represent an organic solution, while the blue areas represent an aqueous solution. Unlabeled inputs are fresh chemicals fed to the system, assumed to have zero rare earth element content. Samples from the system were taken every hour for 44 hours at each of the 10 numbered locations. Only six samples of pregnant leachate solution (PLS) were taken over the course of the test, however the PLS samples were found to have little variability. It was deemed appropriate to use the bootstrap resampling technique (Efron 1979) to produce a PLS assay for each sampling interval from the 6 observations. In doing so, the 6 original data points were randomly sampled with replacement 44 times thereby expanding initial PLS sample set into a sample set containing 44 observations. The bootstrap technique operates under the assumption that all original data were obtained from the same distribution. After examination of the data for the remaining sampling locations, it was determined 26 of the samples were obtained after steady state was reached. These 26 samples would be used for data reconciliation.

Next, linear constraints specified in Equation (3.29) were passed to the `constrainProcess()` function, and the required `X` argument of the `BMB()` function was generated. The default prior hyperparameter settings for the `BMB()` function were used. Posterior draws from both Bayesian models were generated along with approximations for the log-marginal likelihood. 100,000 iterations of the Gibbs sampler were run after a burn in of 10,000 iterations by setting the argument `BTE = c(10000,110000,1)`.

$$\begin{aligned}
0 &= y_{\mathrm{PLS}} + y_{05} - y_{01} - y_{06} \\
0 &= y_{01} - y_{02} - y_{07} \\
0 &= y_{02} - y_{03} - y_{08} \\
0 &= y_{03} - y_{04} - y_{09} \\
0 &= y_{04} - y_{05} - y_{10}
\end{aligned} \tag{3.29}$$

Upon inspection of the MCMC diagnostics, effective sample size was relatively low. In particular this was true for the covariance model, with the minimum value at 16,850. Checking auto-correlation plots for the samples revealed a lag in correlation of about 10 iterations for the chain. The Gibbs sampler was rerun for 1,000,000 iterations, and after 10,000 samples were removed as burn in, every 10^{th} sample was saved as one of the independent draws from the marginal posterior distribution. A summary of the resulting diagnostics is shown in Table A.3 in Appendix A.4. Performance was improved at the expense of computation time.

Lastly, both of the Bayesian models were compared by computing the $\log(BF)$ from the output in `BMB()$lml`.

3.5 Results and Discussion

Part of the motivation for the Bayesian Mass balance is in part to provide a useful model that has some advantages over a point estimate model. The posterior density plots described in Section 3.3.1.1 are shown in Figure 3.4. For the single data set used to generate Figure 3.4, each model shows a case where it provides the best estimate. More importantly, these plots show how Bayesian models provide more information about process uncertainty, and therefore more insight into the process. When viewing the Bayesian densities, one gets an idea of the likely range of outcomes, instead of just a single value with no uncertainty quantification. The plotted HPDI intervals allow for visualization of the uncertainty quantification. Shorter intervals indicate less uncertainty. Figure 3.4 also gives insight into how both Bayesian models are related. The independent variance model generally shows a tighter HPDI than the covariance model.

Selecting a model via Bayes factor in Section 3.3.1.1 is achieved using the output of `BMB()`. Letting M_1 be the covariance model, and M_2 be the independent variance model, a $\log(BF) = \log(p(\boldsymbol{y}|M_1)) - \log(p(\boldsymbol{y}|M_2)) = 126.9$ was calculated. A $\log(BF)$ of 126.9 shows the covariance model is certainly a better fit to the data than the independent variance model (Kass and Raftery 1995). The result is not surprising. Because feed rate varies stochastically, observed flow rates will be correlated. When there is a feed surge, all flow rates for a given component should increase accordingly. Correlation between locations is also induced via the stochasticity in process performance. Knowing this simulation

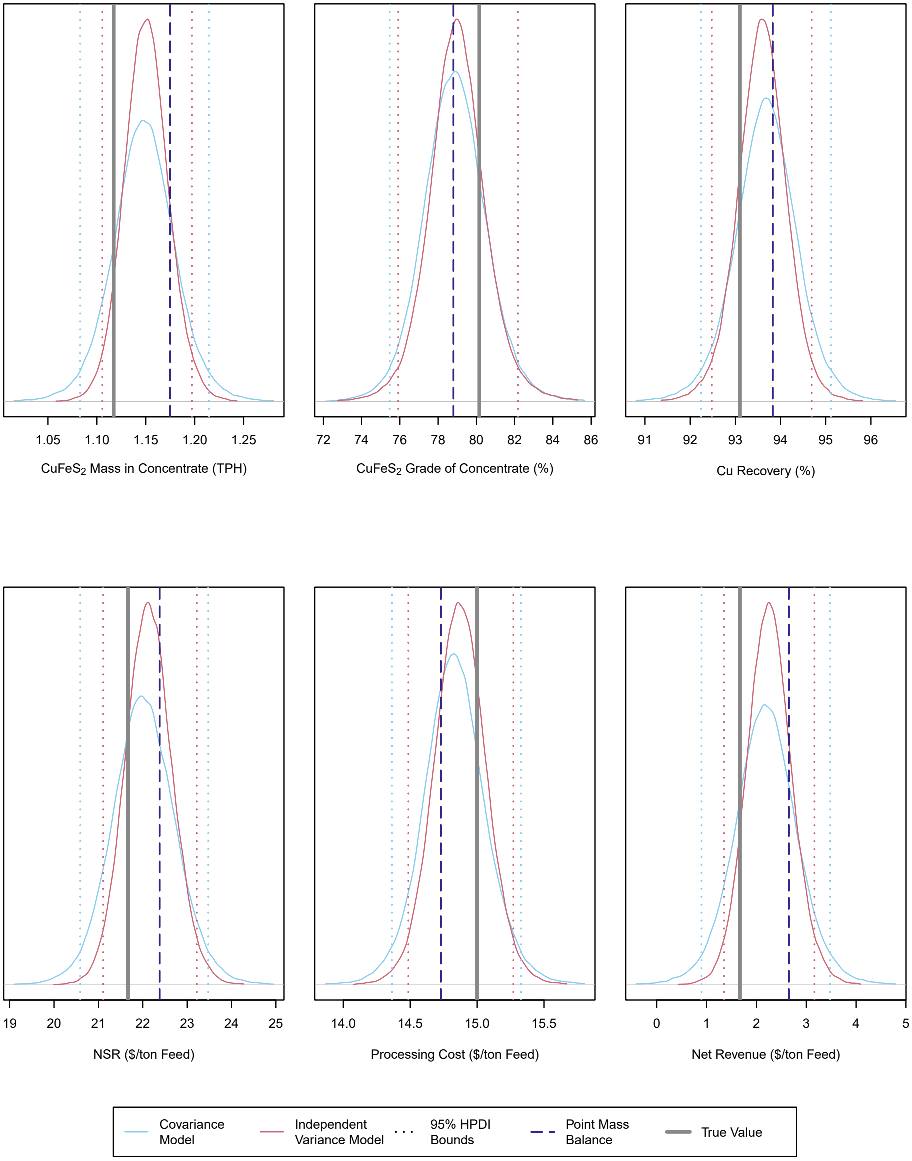

Figure 3.4: Results from reconsciliation of simulated data, including posterior densities with 95% HPDIs, point mass balance estimates, and true values.

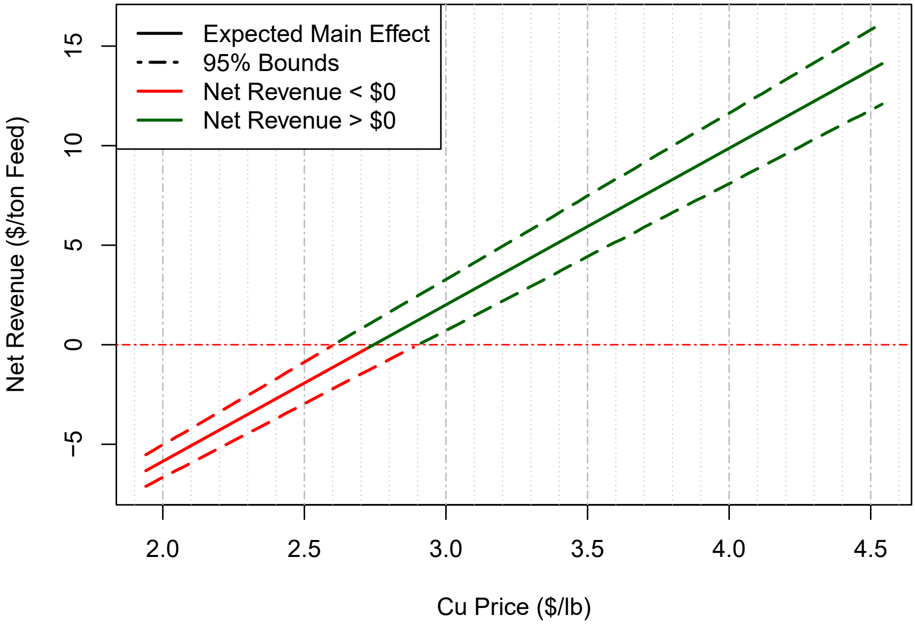

Figure 3.5: Main Effect of Cu price on Net Revenue.

structure, support for the correlated error model indicated by the Bayes factor is sensible.

The main effect plot in Figure 3.5 displays how the output of a Bayesian model can be used for further insight into applications related to process uncertainty. The distribution of $\ _{x}[f(x, \boldsymbol{y}_{\mathrm{bal}})|x_j]$, where $f(x, \boldsymbol{y}_{\mathrm{bal}})$ is net revenue per ton processed and x_j is copper price, is plotted for a sequence of copper prices. This plot gives an engineer insight into how the uncertainty of process performance effects net revenue per ton mined. It is shown that when copper price is low, the interval for $\ _{x}[f(x, \boldsymbol{y}_{\mathrm{bal}})|x_j]$ is more narrow, indicating lower uncertainty of the expected net revenue. When copper prices are higher, the interval is wider, showing higher uncertainty in the expected net revenue. This is an interesting relationship as it indicates there is more uncertainty in net revenue per ton processed when commodity prices are higher. Examining Equation (3.28) this is a sensible relationship. There is a one dimensional distribution for $\frac{Cc}{F}$ which is scaled by the change in copper price, effectively broadening the distribution for the slope of the line as copper price increases. Variability in other calculations shift the intercept of the line, and are unaffected by a change in copper price. This graphical method allows for easier visualization and communication of such concepts, and can be used to understand more complicated relationships.

Figures 3.4 and 3.5, as well as the model selection application show there are uses for the output of Bayesian data reconciliation models. However, if the model is wildly inaccurate, these applications are useless. The results of the Monte Carlo experiment (§3.3.1.2) in Figure 3.6 show the Bayesian models are

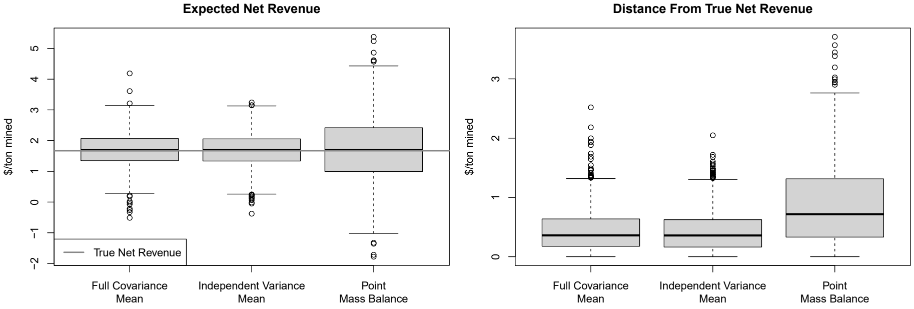

Figure 3.6: Variability of expected net revenue, taken from Monte Carlo simulation.

in fact useful. The left box plot in Figure 3.6 shows that the median of result from all three models is close to the truth. The small discrepancies are likely due to Monte Carlo variance. The script used to generate the data for Figure 3.6, as well as the data obtained for this publication, are available on an online repository [dataset](Koermer 2020b).

More important is the variability in the results. Fifty percent of the observations lie within the box, the whiskers represent the 1.5 interquartile range, and the points represent outliers. Both of the Bayesian models have more narrowly bounded boxes, whiskers, and outliers. For this data generating mechanism, a point estimate mass balance is more likely to produce results that show the process is unprofitable, or embellish the profitability giving decision makers a false sense of confidence.

The right box plot in Figure 3.6 displays the disparity in variability using the distance of each estimate from the true value. The median for both Bayesian models is clearly lower than the median for the point estimate model, implying the Bayesian models are frequently closer to the true value. The upper whiskers for the point mass balance extend to almost \$1.50/ton higher than the Bayesian models. When the true net revenue is 1.669 the level of inaccuracy is concerning.

With these results, an important question to ask is, *Why did the point estimate model perform so poorly?* One possibility is the error structure of the point estimate model, as the lack of error correlation is a mis-match for the data generating mechanism. However, the error structure is practically the same for the independent variance Bayesian model, so there must be other, more important, factors.

Looking at the model derivation in Appendix A.2 gives some more clues. First, the error for the reconciled assay values, $\hat{\boldsymbol{a}}$, in Equation (A.5) is normally distributed. $\hat{\boldsymbol{a}}$ is a percentage on $[0, 100]$, while a normal distribution has infinite support, making the least squares criteria in (A.5) a mis-specification. In the beginning of this investigation, the point mass balance was observed to perform poorly for assays near 0% or 100%. The default parameters in `twonodeSim()` are specified to give assay values that are not too close to 0 or

100, so that the comparison between models is fair. The point estimate model in Appendix A.2 was chosen as it was expected to work well with a low number of observations, relative to methods which estimate a full covariance matrix.

Another hypothesis for why there may be a difference between the accuracy of the point estimate model and the Bayesian models is simple; for this application there are some advantages in using Bayesian inference. When sample variance is estimated for the point estimate model, there is no account of the uncertainty of that estimate. For the Bayesian models, the marginal distribution of reconciled mass flow rates is obtained after integration over all possible values for the model variance or covariance parameters, likely having some positive effect on model accuracy.

Moving on to a comparison between the two Bayesian models, Figure 3.6 shows little difference between the posterior mean accuracy of both models. The accuracy of uncertainty quantification is another issue. A perfect model would place the true net revenue within the bounds of a 95% HPDI, 95% of the time. In the Monte Carlo experiment, the more narrow 95% HPDI bounds of the independent variance model contained the expected net revenue in 93.3% of the experiments, underestimating the uncertainty. The comparatively wider 95% HPDI bounds of the covariance model captured the expected net revenue 97% of the time, a slight over estimate. While the posterior mean values from both Bayesian models were accurate, the covariance model was observed to have superior uncertainty quantification. This result is corroborates the results found from the Bayes factor, which strongly suggested the covariance model was a better fit for the data.

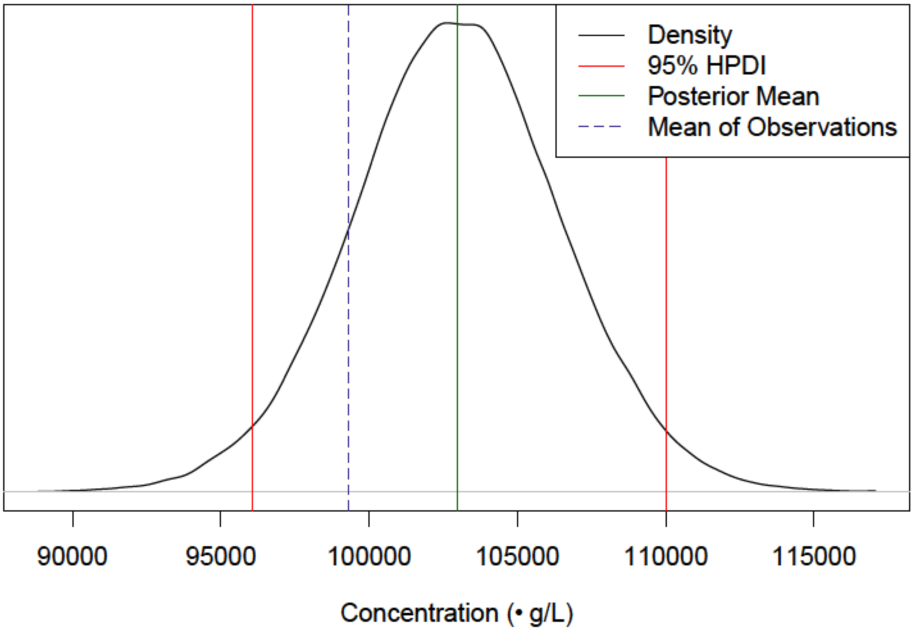

Figure 3.7: Mass balanced concentration of stripping solution.

Bayesian mass balance methods are not relegated to use only with simulations, nor do they require assumptions that are impossible to determine for real data. Section 3.4 shows Bayesian data reconciliation works well with real data. When using the real data set, there were some issues with auto-correlation. Initially the minimum effective sample size of a parameter for either model was 16.9% of the total collected samples. Auto-correlation issues were resolved by running the Gibbs sampler for more iterations and saving less samples. This practice of thinning the samples improved the minimum effective sample size for a parameter to 93.1% of the total. None of the CD scores are high enough where convergence is of concern.

Using the output from calling the `BMB()` function for the independent variance and covariance error structures, a log Bayes factor of 126.52 was approximated. A Bayes factor with this value shows strong support for the covariance model being a better explanation of the data. The resulting Bayes factor is sensible, as one would expect there to be some error correlation for this complicated liquid-liquid extraction process.

After selecting the covariance model, the posterior draws were used to plot a density for concentration of the stripping solution as well as the posterior mean, and 95% HPDI in figure 3.7. This plot shows the best estimate of of the concentration is 103,000 $\mu g/L$, and it is likely the true concentration is between 96,060 and 110,000. The arithmetic mean of the observations falls within the bounds of the 95% HPDI. It is quite possible this value is the true concentration. However, simply taking the mean of the observations does not take into account the conservation of mass or error correlation which may occur as concentrations fluctuate. While using the mean value can give an ok estimate and is easy to do, it can give less reliable results.

One downside of Bayesian data reconciliation is an increase in computation time. The average time for data reconciliation and approximation of the the log-marginal likelihood using the simulated data was 13 minutes on an laptop with a 2.8 GHz Intel i7 processor and using Intel's math kernel library. This time is reasonable, but it is much longer than the perceptually instantaneous results produced form the point estimate model. A much longer computation time was observed when real data was used, due to the computation time required to produce the 1,000,000 draws before burn in was removed and samples were thinned. Average time for the two models was 43.9 minutes, which is longer but still not unreasonable. However, for a process with more sample components or many nodes, computation time for a Bayesian mass balance can start to become an issue.

With some additional code, perhaps using the `doParallel` and `foreach` packages, it is possible to run the `BMB()` function in parallel to reduce the computation time. Parallel processing was not implemented in `BayesMassBal` V. 1.0.0 to ensure ease of use. While computation time is the most obvious downfall of some applied Bayesian methods, spending some money on a computer upgrade, renting time on a cloud computing service, or waiting a little extra time for the results, is likely less costly than making a less informed, hasty decision.

A second downside is the added complexity in implementation. Bayesian inference is often reserved for graduate level statistics coursework. However, the tools in the `BayesMassBal` package remove many of the barriers of implementation for individuals unfamiliar with the intricacies of Bayesian inference.

3.6 Conclusion

Bayesian data reconciliation methods are useful. Most importantly, the posterior mean can be more accurate than estimates produced from a well established point estimate model. Model selection methods for Bayesian models are already established, and can be used to find which model best fits a data set, allowing for more reliable data reconciliation.

The uncertainty quantification inherent to Bayesian methods allows for the generation of posterior distribution plots which can aid in the understanding of the precision of performance metrics. Applications such as plotting a main effect can also take advantage of the uncertainty quantification.

Bayesian data reconciliation is not just a thought experiment or only useful with simulations. These methods can be used on real data. The `BayesMassBal` package was designed with use on real data in mind. Models not specified in this publication can further improve accuracy. While the `BayesMassBal` package includes models that can work well in a general sense, some applications may benefit from custom built models.

The Bayesian models were shown to be superior to the point mass model in this examination. However, each unique data set presents its own series of challenges. Even with the sound theoretical reasoning and results presented indicating Bayesian data reconciliation is superior, the models presented are not a catch-all best choice for every scenario. Through the `BayesMassBal` package, published as a companion to this article, the reader can easily see if Bayesian data reconciliation is useful for their unique process.

CRediT authorship contribution statement

- **Scott Koermer:** Conceptualization, Methodology, Software, Validation, Formal analysis, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization
- **Aaron Noble:** Resources, Writing - Review & Editing, Supervision, Project administration, Funding acquisition

Role of Funding Source

Funding for this work has been provided by the US Department of Energy under award No. DE-FE0026927. The funding source(s) had no involvement in study design; in the collection, analysis, and interpretation of data; in writing the report; and in the decision to submit this article for publication.

Declaration of Competing Interest

The authors declare that there is no conflict of interest.

Acknowledgements

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Federal Award Number: DE-FE0026927

Chapter 4

Analysis of Steady State

Abstract

To improve efficiency, separations engineers will typically design process circuits containing recirculating streams, which mix one or more of the process outputs with the feed material. Doing so can improve efficiency, but will cause a delay in the system reaching steady state conditions until the recirculating load mass flows stabilize. In testing separation circuits, often engineers will test a variety of factors and complete an analysis from sample results. Knowledge of if a process is at steady state, as well as the steady state conditions of a process, is essential for a valid techno-economic analysis. However, the definition of process steady state is often poorly defined, or does not include uncertainty quantification. If the performance of a process operating under two different sets of conditions are compared, an engineer who does not test for steady state or quantify steady state conditions risks producing a faulty analysis. In this chapter, a Bayesian statistical method for testing if all streams are at steady state is further motivated and then derived. Then after testing for steady state, the same model is used with a prior distribution which enforces a steady state assumption to estimate steady state conditions. The resulting models are then used with observed conditions of a solvent extraction plant to infer steady state conditions which can be used as a part of further analysis.

4.1 Introduction

A process at steady state, is one where properties of interest do not change over time Paula (2006). A typical property of interest for the steady state of a separation process is mass. For example Figure 4.1 shows a simple separation process, which could be considered to be operating at steady state while the mass flow constraint $F = C + T$ is valid. Often process models for data reconciliation (see Koermer and Noble 2021) require a data to be collected from a process at steady state. Calculations of the percent of a feed component exiting the process in a concentrate are invalid prior to achieving steady state. The amount of time for a solvent extraction (Lo, Baird, and Hanson 1983) circuit to reach steady state is non-trivial, and estimating when a circuit reaches steady state without a mathematical framework can lead to inconsistent analysis.

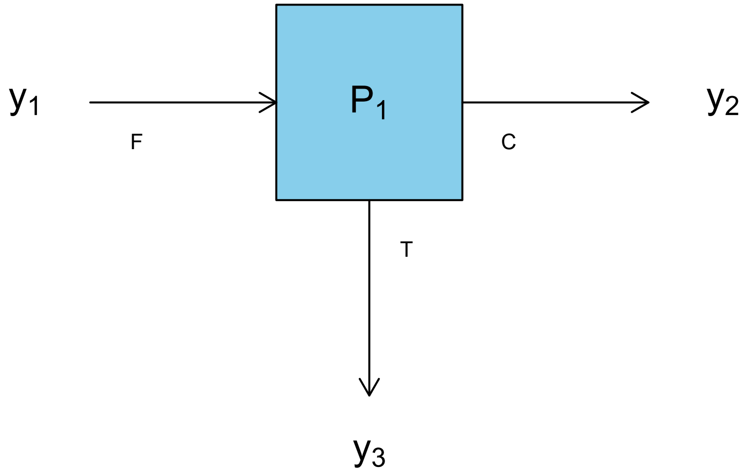

Figure 4.1: A simple separation process showing the input feed flow F, the concentrate flow C, and the tailings flow T.

There are numerous statistical methods in the literature for estimating steady state. A simple method is to take a collection of data observed within a specified time window, and calculate if the range of this data is less than some previously stipulated tolerance (Bethea and Rhinehart 1991). A slightly more complicated method is to use linear regression with the same data set and test if the slope of the data over time is equal to zero (Bethea and Rhinehart 1991). Similar to an F-test, Von Neumann (1941) proposed using a ratio between the mean standard squared deviation from the sample mean and the mean squared deviation between subsequent data points. Using this *R statistic* method the statistic for a process at steady state has an expectation equal to 1, assuming there is no autocorrelation between the data.

Filtering methods, described in Cao and Rhinehart (1995), compute a similar ratio of variances using an exponentially weighted moving average. While this method is able to filter data and test for steady state, it is not suitable for data with autocorrelation and an arbitrary tuning parameter must be selected. Uncertainty quantification is mostly absent from these methods.

As an alternative, a Bayesian tool for quantifying and estimating steady state conditions is proposed. First, a linear time series model is used to check evidence of *stationarity* of the data observations. If there is sufficient evidence the observations are from a stationary distribution, the expected value of the

mean, given the condition of stationarity, can be computed and utilized for further analysis. Although these tools are imperfect, having a rigorous definition of steady state is advantageous for comparing multiple processes and process parameters within an analysis.

4.2 Methods

4.2.1 Bayesian Inference

Bayesian inference differs from classical statistical inference, in that model parameters are inferred through Bayes' rule (4.1).

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \tag{4.1}$$

Importantly, a prior distribution $p(\theta)$ on model parameters θ can be specified before any data is observed. One can incorporate knowledge about the model parameters into the prior distribution to ensure reasonable inference of the posterior distribution after data is observed $p(\theta|y)$. As a relevant example, if one wanted to assume that the value of an inferred model parameter was bound between -1 and 1, they could specify a normal prior truncated at -1 and 1.

Often, analytical inference of the marginal posterior distribution of a model parameter is intractable. Numerical Markov-Chain Monte-Carlo methods, particularly the Gibbs sampler (Gelfand et al. 1990), are often used in Bayesian inference to obtain samples from the marginal posterior distribution of model parameters. See the textbooks of Hoff (2009) for further details on Bayesian inference and Robert and Casella (2013) for further information on Monte-Carlo methods for statistical simulation.

4.2.2 The Autoregressive Time Series Model

An autoregressive (AR) time series process is a model where a future state is dependent on the past state (Shumway and Stoffer 2000). An example, which includes the addition of a constant value (μ), and random normally distributed perturbations (ϵ) is shown in Equation (4.2). In the model shown, the state at time step t, y_t, is dependent on the previously measured value y_{t-1}. y_t is separated temporally from y_{t-1} by a consistent and discrete time interval. Because y_t is only dependent on a single lagged value of the observation, the model is referred to as an Autoregressive 1, or AR(1), process.

$$y_t = \alpha y_{t-1} + \mu + \epsilon \tag{4.2}$$

A weakly stationary time series process implies that the variance of the process is finite, the mean is constant in time, and that the covariance between two observations is only dependent on their difference in time and not the value of some t itself (Shumway and Stoffer 2000). For a weakly stationary process, from here on referred to simply as *stationary processes*, it is possible to predict a future y_{t+h} from y_t using a linear model. For the AR(1) process (4.2) is stationary only if $|\alpha| < 1$ (pages 87-90 of Shumway and Stoffer 2000) mean that the process generating y_t from y_{t-1} is stationary, and causal.

If $\alpha = 1$, the process is considered a white noise process, or a random walk and considered non-stationary because of the non constant mean. If $|\alpha| > 1$ the process is considered stationary, however the observation y_t is dependent on future observations y_{t+h} (page 80 Shumway and Stoffer 2000), making the model not useful. A stationary process and exploding process are shown in Figure 4.2. One can gain intuition as to why an exploding process is problematic by iterating through Equation (4.2) with $\alpha > 1$.

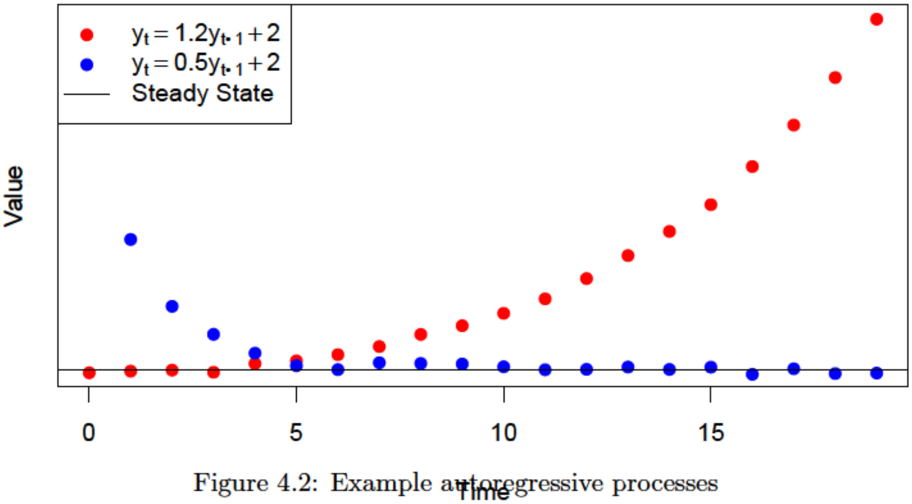

Figure 4.2: Example autoregressive processes

For a stationary process, the constant mean required for stationarity is not given by μ in Equation (4.2), and instead implies $\mathbb{E}[y_t] = \mathbb{E}[y_{t-1}]$. Given this condition, the mean function of the AR(1), or in the case of this application the process steady state, can be calculated as shown in Equation (4.3).

$$\begin{aligned}\mathbb{E}[y_t] &= \mathbb{E}[\mu + \alpha y_{t-1} + \epsilon] \\ &= \frac{\mu}{1-\alpha}\end{aligned} \tag{4.3}$$

For a set of time series observations in discrete time from time y_1 to y_T, if one were to estimate the parameters μ and α, and $|\alpha| < 1$, then it would be possible to estimate the expected value of the time series as (4.3).

4.2.3 Inference

Given a vector of T time series observations y modeled as Equation (4.2) with $\epsilon \sim \mathcal{N}(0, \sigma^2)$ the joint likelihood function can be written as Equation (4.4).

$$\mathcal{L}(\mu, \alpha, \sigma^2|y) \propto (\sigma^2)^{-\frac{T-1}{2}} e^{-\frac{\frac{1}{2}(y_{-1}-X\beta)^T(y_{-1}-X\beta)}{\sigma^2}} \tag{4.4}$$

Where:

$$y_{-1} = \begin{bmatrix} y_2 \\ y_3 \\ \vdots \\ y_T \end{bmatrix}$$

$$X = \begin{bmatrix} 1 & y_1 \\ 1 & y_2 \\ \vdots & \vdots \\ 1 & y_{T-1} \end{bmatrix}$$

$$\beta = \begin{bmatrix} \mu \\ \alpha \end{bmatrix}$$

Without enforcing stationarity, the Jeffreys priors (Jeffreys 1946; Kass and Wasserman 1996; Yang and Berger 1996) of $p(\beta) \propto 1$ and $p(\sigma^2) \propto 1/\sigma^2$ can be stipulated for parameter inference. Using the prior $p(\beta, \sigma^2) \propto \frac{1}{\sigma^2}$ produces the conditional distributions for β as (4.5) and σ^2 as (4.6).

$$p(\beta|y, X, \sigma^2) \propto e^{-\frac{\frac{1}{2}(\beta-(X^TX)^{-1}X^Ty_{-1})^TX^TX(\beta-(X^TX)^{-1}X^Ty_{-1})}{\sigma^2}} \tag{4.5}$$

$$p(\sigma^2|y, X, \beta) \propto (\sigma^2)^{-\frac{T-1}{2}-1} e^{-\frac{\frac{1}{2}(y_{-1}-X\beta)^T(y_{-1}-X\beta)}{\sigma^2}} \tag{4.6}$$

Equation (4.5) can be recognized as a bi-variate normal distribution with a mean of $(X^TX)^{-1}X^Ty_{-1}$, and a covariance matrix of $\sigma^2(X^TX)^{-1}$. Equation (4.6) can be recognized as an inverse gamma distribution with a shape parameter of $\frac{T-1}{2}$ and a scale of $\frac{1}{2}(y_{-1} - X\beta)^T(y_{-1} - X\beta)$.

Inference of μ, α, and σ^2 while enforcing a stationarity assumption can be conducted by specifying a truncated multivariate normal distribution for $p(\beta)$. The goal of utilizing such a prior is to bound inference of α to between -1 and 1, but not to constrict the values of μ. Allowing for prior independence of μ and α as $p(\mu, \alpha) = p(\mu)p(\alpha)$, means that one can set $p(\mu) = \mathcal{N}(\mu_0, v_{0,\mu})$ and $p(\alpha) = \mathcal{N}_{\alpha\in(-1,1)}(0, v_{0,\alpha})$. The parameters for $p(\alpha)$ can be set with a mean of 0 and a large variance in order to limit the influence on the posterior.

Utilizing the truncated normal prior in conjunction with $p(\sigma^2) = 1/\sigma^2$ leads to the conditional posterior derivations in (4.7) and (4.8), where $V_0 = \text{diag}(v_{0,\mu}, v_{0,\alpha})$, β_0 is the vector $[\mu_0, 0]^T$, $\tilde{\beta} = \left(\frac{1}{\sigma^2}X^TX + V_0^{-1}\right)^{-1}\left(\frac{1}{\sigma^2}X^Ty_{-1} + V_0^{-1}\beta_0\right)$, $\tilde{V} = \left(\frac{1}{\sigma^2}X^TX + V_0^{-1}\right)^{-1}$, and $I[|\alpha| < 1]$ is an indicator function equal to 1 when $|\alpha| < 1$.

$$p(\beta|X,y,\sigma^2) \propto e^{-\frac{1}{2}(\beta-\tilde{\beta})^T\tilde{V}^{-1}(\beta-\tilde{\beta})} I[|\alpha|<1] \tag{4.7}$$

$$p(\sigma^2|y,X,\beta) \propto (\sigma^2)^{-\frac{T-1}{2}-1} e^{-\frac{\frac{1}{2}(y_{-1}-X\beta)^T(y_{-1}-X\beta)}{\sigma^2}} \tag{4.8}$$

Equation (4.7) can be recognized as a bi-variate normal distribution with a mean of $\tilde{\beta}$ and a covariance matrix of $\tilde{V}$. However, the distribution for the second element of $\tilde{\beta}$, corresponding to α, is truncated at $(-1,1)$. Note that the conditional posterior distribution of σ^2 written as (4.8) is equivalent to (4.6).

Inference for both the unconstrained and stationary models are carried out using a Gibbs sampler, iteratively drawing the parameters from each conditional posterior distribution. A function, named `ssest` and written in `R`, is provided in Appendix B.1 for drawing samples from the marginal posterior distributions using a Gibbs sampler. `ssest` can provide inference for bot the constrained and unconstrained models. The `y` argument to the `ssest` function is a vector of sequentially observed mass flows or concentrations. `BTE` is a three element vector containing the number of Markov-Chain Monte-Carlo iterations for `B`urn in, `T`otal, and `E`very, with the same meaning as the methods in §3. The last argument `stationary` is a logical where if `stationary = FALSE` the unconstrained model is fit, and if `stationary = TRUE` the model is constrained so that $|\alpha| < 1$. To make the `ssest` function easier to use for a general audience, prior distribution hyperparameters for the constrained model are loosely informed by the data. The function sets $\beta_0 = [\texttt{mean(y)}, 0]^T$, and $V_0 = \text{diag}(\texttt{100 * var(y)}, 1000)$ with the intention of providing little information.

4.2.4 Application

To test the method, total rare earth element (TREE) concentrations were observed from a solvent extraction (SX) processing plant at the locations observed in 4.3. Flows entering or exiting a blue part of the diagram are aqueous solutions and flows entering or exiting yellow parts of the diagram are organic solutions. Unlabeled flows are assumed to have no TREE concentrations, as they are fresh acid solutions used for stripping elements of interest from the organic phase. Flow y_6 is a constant pregnant leachate solution (PLS) feed to the system and did not need analysis. Flow y_1 is the feed of the organic phase and not constant because of its use in other processes.

Samples were collected from each location hourly for 44 hours of run time. At each location in Firgure 4.3 an unconstrained AR(1) model was fit to TREE concentrations. For flow y_2 the 12th sample was an outlier, almost twice as large as any other measured concentration. Likely the unusually high value was caused by sampling or laboratory errors, and only data after $t = 12$ was used for evaluation. For inference, the burn in was set to 500 samples and the total number of MCMC iterations was set to 20,000.

Statistical samples of α were collected for each physical sampling location. Researchers decided before fitting the models, that if 95% of the samples of

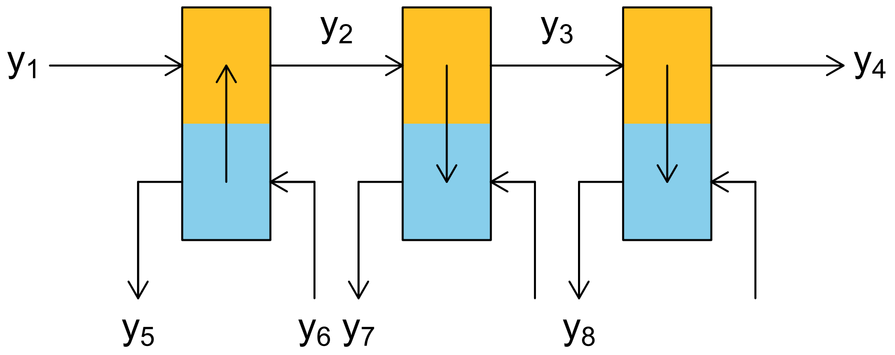

Figure 4.3: Process used for data collection.

alpha were in between -1 and 1 the flow would be considered stationary. If all flows were considered stationary, the process could be considered stationary and at steady state. If after inspecting samples of α the process is deemed stationary, then the model could be refit for each physical sampling location with a prior on α which enforces stationarity. For any physical sampling location where samples satisfying $|\alpha| \geq 1$ occurred with the unconstrained model, the equation for the mean of the process (4.3) is no longer valid. After refitting the constrained model at each location using `ssest(y, BTE =  c(500, 20000,1), stationary = TRUE)` for each location. Then the expected steady state values can be computed by evaluating (4.3) with each pair of samples μ, α to produce the distribution of process steady state with uncertainty quantification. The mean or samples of the distribution of $[y|\mu, |\alpha| < 1]$ can be used for further technoeconomic analysis of the process.

4.3 Results and Discussion

After organizing the data and fitting the unconstrained AR(1) process, the percent of samples of α satisfying $|\alpha| < 1$ were computed and are tabulated in Table 4.1. Inspection of Table 4.1 provides strong evidence of a stationary process, given the requirement of 95% of samples satisfy $|\alpha| < 1$, with the lowest percent of samples at a given location satisfying the constraint is y_3 with 97.74%.

Each data set was plotted with the information available from the parameter samples. Similar plots to what is shown here are available by loading the `BayesMassBal` package with `library(BayesMassBal)` and feeding the output of the `ssest` function included in Appendix B.1 into the `plot(...)` function. Plots for the results from location y_3 and y_7 are shown here as examples for examining the results.

Figure 4.4 shows the data along with the posterior distribution of α for y_3. Inspecting the plots, one can see that there are some samples which satisfy $|\alpha| \geq 1$, although the fraction of samples is minimal. The uncertainty quantification

Table 4.1: Samples of α providing evidence of independently stationary process flows.

Sample Location	Samples Where $\|\alpha\| < 1$
y_1	99.3%
y_2	99.1%
y_3	97.7%
y_4	99.8%
y_5	100%
y_7	100%
y_8	100%

of α aids engineers in understanding the evidence that a stationary process generated the data. Looking at the data alone, as the bottom plot in 4.4, making an assumption without analysis that the data was observed from a steady state process is not unreasonable.

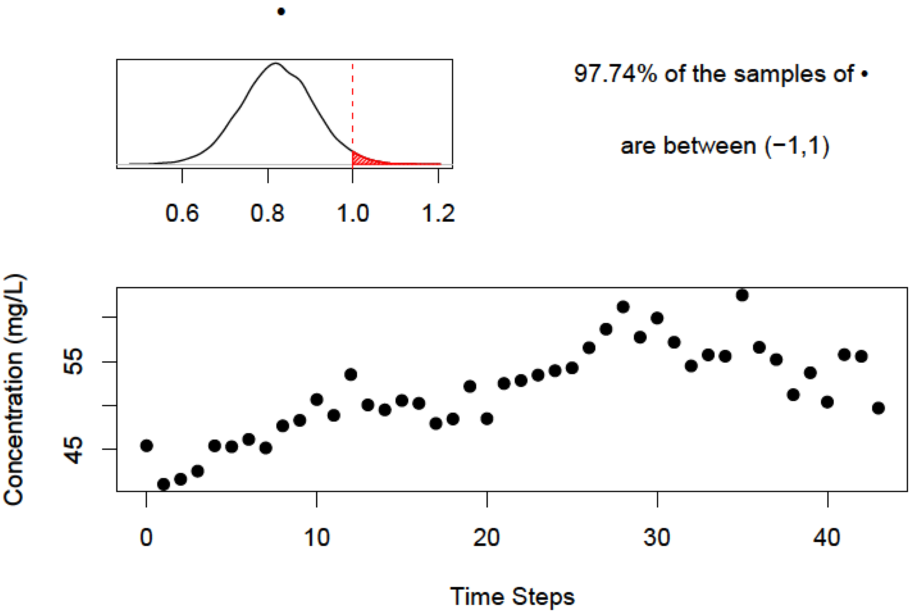

Figure 4.4: Plot of results from fitting unconstrained AR(1) model to y_3.

Figure 4.5 shows the equivalent plots to those produced for figure 4.4, but for y_7 where all samples of α satisfied $|\alpha| < 1$. Because all samples of α satisfied the stationarity constraint, the mean of the expected value of the process steady state conditions can be computed, and are plotted in the top right and bottom plots of Figure 4.5 for examination, along with the 95% Credible Interval (M.-H. Chen and Shao 1999). The results in Figure 4.5 appear sensible as there is not

Table 4.2: Expected value of steady state concentration, given the process is stationary.

Sample Location	$\mathbb{E}[y]$ (mg/L)
y_1	41.17
y_2	81.64
y_3	51.34
y_4	41.45
y_5	0.02
y_7	20.27
y_8	100.43

much variation after the first few data points. Just looking at the data plot one would likely conclude that the TREE concentration at y_7 had reached steady state, similar to the results of the statistical analysis.

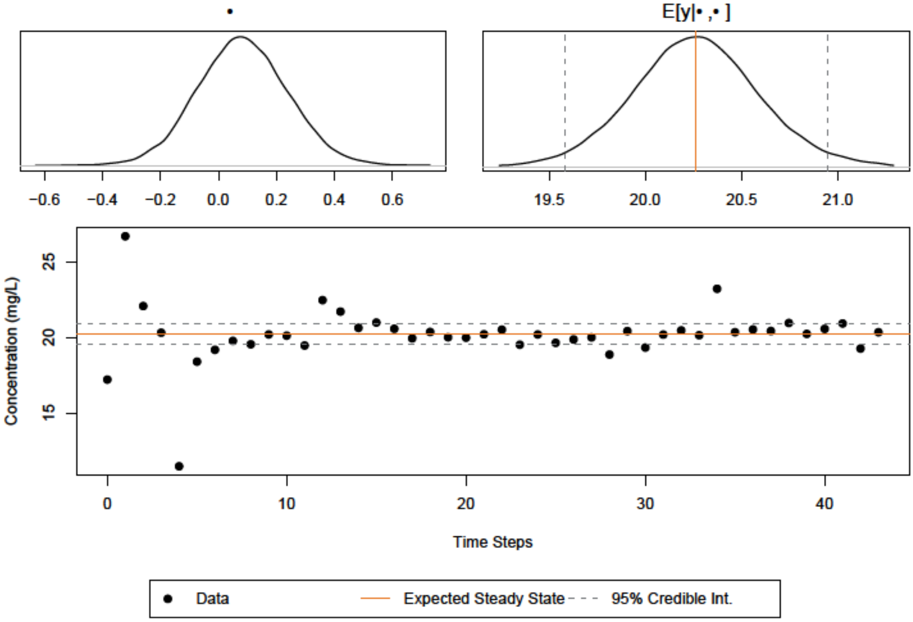

Figure 4.5: Plot of results from fitting unconstrained AR(1) model to y_7.

Because all physical sampling locations were deemed stationary, the constrained AR(1) model was refit to the locations where any sample of α satisfied $|\alpha| \geq 1$, in order to estimate steady state concentrations with Equation (4.3). The mean calculated $\mathbb{E}[y|\mu, |\alpha| < 1]$ values are shown in Table 4.2.

Results for data fit to a constrained model was plotted again for examination, and the results to fitting the constrained AR(1) process to y_3 is shown in Figure

4.6. On the top right plot the distribution of the steady state concentration $\mathbb{E}[y|\mu, |\alpha| < 1]$ is shown, along with the expected value of the concentration as the orange line and the 95% credible interval as the gray dotted lines. The same information about the distribution of $\mathbb{E}[y|\mu, |\alpha| < 1]$ is shown on the bottom plot of the data. The estimates with the real data appear to be a sensible evaluation of steady state conditions, given that the process is stationary, with uncertainty quantification.

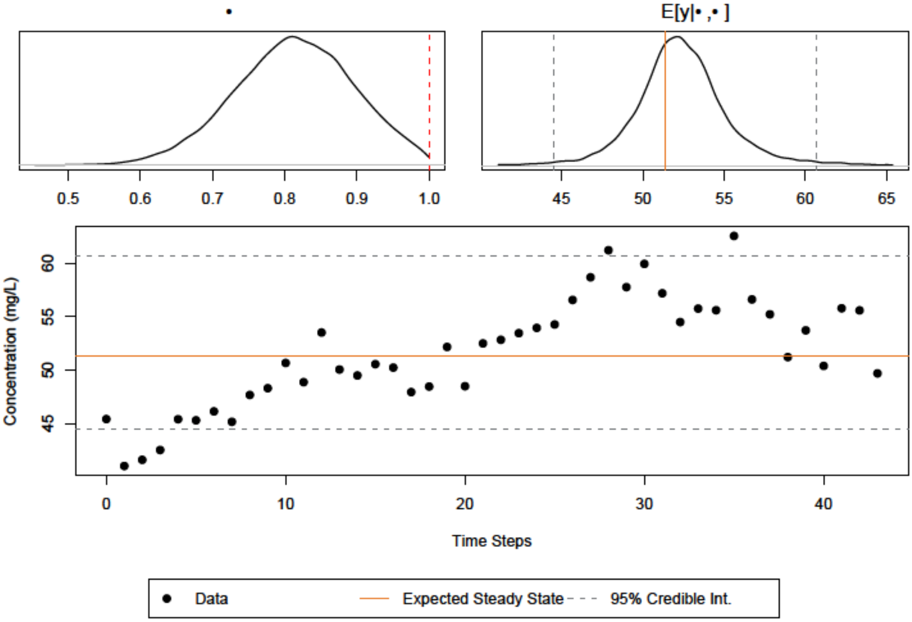

Figure 4.6: Plots showing the results of fitting a stationary AR(1) process to y_3 with uncertainty quantification of steady state conditions.

4.4 Conclusion

The methods presented in this chapter are not applied with the intention of perfection. Using the methods outlined for steady state determination the conservation of moss or correlation of observations is not taken into account. One would not be able to utilize the methods to estimate steady state conditions for a process where all mass flows are not approaching steady state.

Instead, what is provided is a method to provide engineers with a consistent metric joined with uncertainty quantification. Tools are provided to first determine if one can utilize the data to fit an AR(1) process and predict a steady state condition. Then, if the process is deemed predictable, one can estimate steady state conditions and the variance of the estimate.

An engineer operating without similar tools may obtain a point estimate of steady state conditions without uncertainty quantification. Alternatively, one may pick an arbitrary set of points and take an average. When utilizing arbitrary methods, comparisons between engineers and between processes become closer to being based in opinion rather than fact. Additionally, methods without uncertainty quantification do not provide an engineer with uncertainty of their analysis derived from the results of the statistical method.

This last statement brings to light the question, *how would one use the results from fitting an AR(1) model to their data?* Such an answer is likely application dependent and left to the reader. One possibility is using the mean $[y|\mu, |\alpha| < 1]$ with a typical point mass balance. A second would be to either use samples of $[y|\mu, |\alpha| < 1]$ or the data points collected at times when all sampling locations were found to be within the 95% credible interval with a Bayesian Mass Balance (§3). Furthermore, one could use the samples of α, μ, and σ^2 and the model $y_{T+1} = \alpha y_T + \mu + \epsilon$ to generate *new data* for analysis. All of these options, and more, could be considered appropriate for particular applications. However, the key to such an analysis is *consistency* by deciding on some consistent application of the statistical methods before data is observed, to ensure a consistent and fair comparison.

Chapter 5

Expected Improvement Optimization of an Uncertain Process

Abstract

Integrated laboratory testing and process modeling are often used to optimize metallurgical process operations. While many metallurgical models are available for commercial processes, observations often differ from model predictions by some amount of *bias*. Bias can be related to factors unaccounted for in the model including mineral speciation, and complex chemical dynamics. Polynomial based response surface methodology can be used to optimize an unknown funciton, but suffer from local convergence problems that can make use with design of laboratory experiments prohibitive. Optimization using Gaussian Process (GP) regression and the Expected Improvement (EI) criteria has been shown to find the global optimum of black box functions with relatively few tests. The goal of this chapter is to explore the utility of using GP regression to find and test the optimum of a typical laboratory experiment, when process dynamics are unknown. After overviewing the methods and a biased leaching simulation based on the shrinking core model, the EI algorithm is repeatedly tested on randomly generated data sets in a Monte-Carlo experiment. The mean behavior of this concept shows that EI has the potential to be used as a systematic method for finding and the optimum of a real process.

5.1 Introduction

Finding the optimum of a complex process is difficult, particularly when functions for adequately modeling such a process are unknown. Response surface methodology (Box and Draper 2007) is commonly used to model unknown processes using polynomials, where the response is modeled as some $y = X\beta + \epsilon$. When using response surface methodology to find and test the optimum of a process, one would use a gradient based technique, approximating derivatives using finite differences in each dimension of X. Gradient based methods may be infeasible when data is expensive. Due to convergence to local optima, large numbers of tests with restarts at random locations are required to ensure convergence on the global optimum (Gramacy 2020).

With the explosion in popularity of machine learning and probabilistic modeling, one might ask *is there a better way?* Active learning and Expected Improvement (EI) (Schonlau 1997) has gained popularity in the computer experiments literature (Gramacy 2020). Additionally EI has been used to optimize real processes for robotics (Tesch, Schneider, and Choset 2011), polymer synthesis (Li et al. 2017), and genetics (Gonzalez et al. 2015). This chapter aims to illustrate how EI can be used to efficiently optimize a real separation process with few experiments, by showing results on a simulation based on the shrinking core model (Yagi and Kunii 1955; Gbor and Jia 2004). An overview of previous optimization work in mineral processing in §5.2. Then, Gaussian process modeling and expected improvement methods utilized for optimizing the leaching process simulation are provided in §5.3. Results from optimization and average behavior in a Monte-Carlo experiment are shown in §5.4.

5.2 Literature Review

There are numerous articles on the use of response surface methodology for modeling and optimization of mineral processes. Aslan (2008) uses response surface methodology and a central composite design for real gravity separator data, collecting 20 tests to fit a polynomial. Veglio and Ubaldini (2001) provides analysis of variance for a real leaching process with data collected using a full factorial design with 9 experiments. A. Chen et al. (2015) fit a polynomial to a process recovering platinum group metals from automobile catalysts using 20 real experiments. Notably, quite a few of these papers have a different definition of optimization than what is used in this chapter. In these publications, generating a response surface is seen as part of the engineering progress for making an improvement. Any optimum found is limited by the complexity of the polynomial used. As written in this chapter, optimization means finding some specific values of factors which yield a minimum or maximum value of a function, see §5.3.3.

There are other, more complicated, methods for fitting a statistical model to real data for optimization available in the literature. Al-Thyabat (2008) uses 40 real world flotation experiments, varying 4 factors, for training and validation of an artificial neural network (ANN). Notably, the experimental design used in Al-Thyabat (2008) is a *one-shot* design. The surrogate ANN is then optimized by evaluating the ANN at 100 points which fall along a 4-D line, and picking the candidate point with the best performance. This technique may have been chosen to reduce computation time, but fails to fully explore the range of the input space for an optimum.

Bu et al. (2016) uses 30 sets of flotation experiments to first select a flotation model which best fits the total data set where four factors are varied. The theoretical flotation models are fit using a least squares criteria. Then, the authors use response surface methodology to predict kinetic constants for the selected model, given a set of input variables, and optimize the process. Essentially the authors build a polynomial surrogate for the kinetic constants, and then optimize the full model utilizing values predicted from the surrogates. This approach takes into account model uncertainty, employs surrogate modeling, and finds an optimal input, but requires quite a large data set, and cannot account for any bias between all the models and observations.

5.3 Methods

5.3.1 Gaussian Processes

One can understand a Gaussian Process by first studying the normal distribution, as shown in Equation (5.1).

$$y \sim \frac{1}{\sqrt{2\pi\nu}} e^{-\frac{\frac{1}{2}(y-\mu)}{\nu}} \tag{5.1}$$

A random variable y can be simulated from a normal distribution with a mean value of μ and a variance of ν. A random draw from a normal distribution with $\mu = 0$ and $\nu = 1$ is shown in Figure 5.1.

Next, examine the multivariate normal distribution (5.2). In the multivariate normal distribution a $N \times 1$ vector Y_N is simulated from the distribution function. In this case each element has the same mean, denoted by the multiplication of the scalar μ times the $N \times 1$ column vector 1_N, and the variance ν.

$$Y_N \sim (2\pi\nu)^{-\frac{N}{2}} e^{-\frac{\frac{1}{2}(Y_N - 1_N\mu)^T (Y_N - 1_N\mu)}{\nu}} \tag{5.2}$$

Figure 5.2 shows a draw from a multivariate normal distribution with $\mu = 0$ and $\nu = 1$. Importantly, the particular form of the multivariate normal distribution in (5.2) does not provide correlation between the elements of a single draw of Y_N. Elements of Y_N are independent and a draw of Y_N is equivalent to N draws of y from (5.1).

Equation (5.3) specifies the covariance matrix K_N, providing a way to model correlation between the elements of Y_N.

$$Y_N \sim (2\pi\nu)^{-\frac{N}{2}} |K_N|^{-\frac{1}{2}} e^{-\frac{\frac{1}{2}(Y_N - 1_N\mu)^T K_N^{-1} (Y_N - 1_N\mu)}{\nu}} \tag{5.3}$$

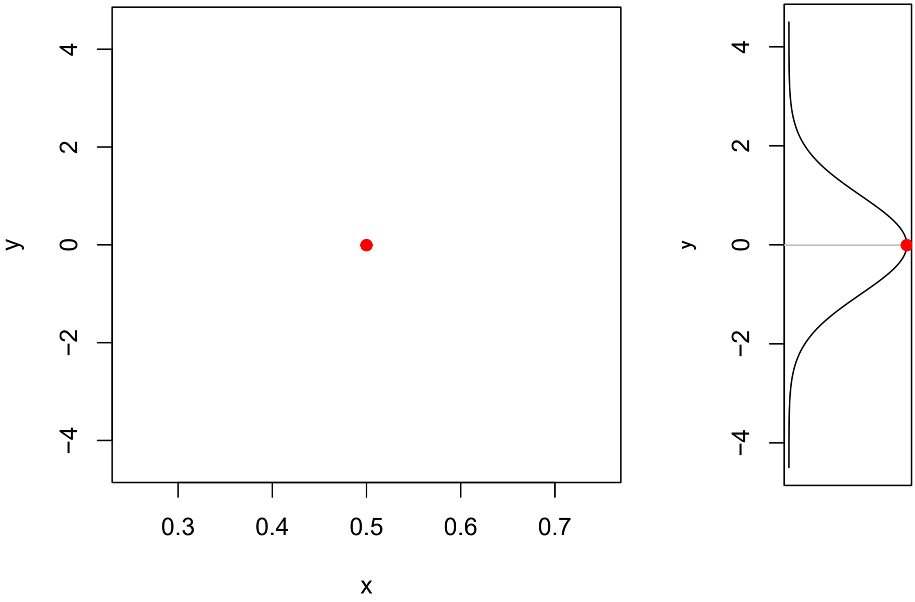

Figure 5.1: Point generated from a univariate normal distribution.

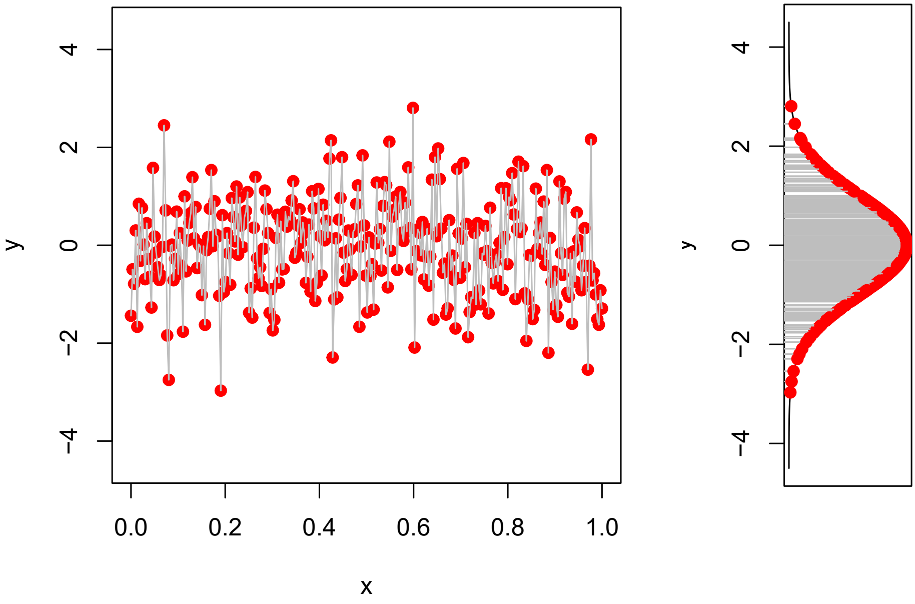

Figure 5.2: Random vector generated from a multivariate normal distribution with no correlation between points within a single vector valued draw.

Figure 5.3 illustrates a draw of Y_N from (5.3) with $\mu = 0$, $\nu = 1$, and a specified covarience matrix K_N.

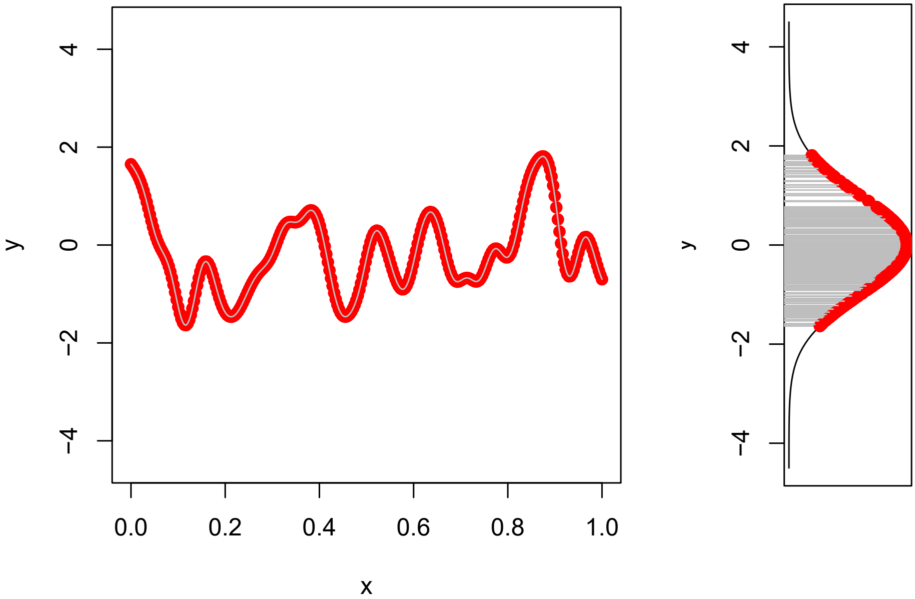

Figure 5.3: Vector valued draw from a multivariate normal distribution, with correlations between elements of a vector dependent on distance between the location of each point on X.

Interestingly, in Figure 5.3, the points that are close to each other on X have similar, possibly correlated, values of y. In GP regression the elements of the covariance matrix K_N are calculated as a function of location X. Often, pariwise distances on X between two elements in Y_N are used for evaluation. The Gaussian covariance kernel, shown in Equation (5.4), is a common choice. The element of K_N related to locations x and x' is the exponentiation of the negative sum of the squared distance between each of the s dimensions of x and x', scaled by the *lengthscale* parameter θ. Because there are s elements of θ the specific covariance function in (5.4) is called a *seperable* covariance function, as opposed to each of the s distances scaled by a single scalar θ.

$$k(x, x') = \exp\left(-\sum_{l=1}^{s} \frac{(x_l - x_l')^2}{\theta_l}\right) \tag{5.4}$$

The plots and equations are interesting, but not yet useful. For utility, a GP has to utilize real data to make predictions. First, break down the vector Y_N into two column vectors, $Y_1(x)$ at location x, and $Y_2(X)$ at location X. Both $Y_1(x)$ and $Y_2(X)$ can be said to be drawn from a zero mean GP prior with covariance structure shown in (5.5).

$$\begin{bmatrix} Y_1(x) \\ Y_2(X) \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \nu \begin{bmatrix} k_{(x,x)} & k_{(x,X)} \\ k_{(X,x)} & K_{(X,X)} \end{bmatrix} \right) \tag{5.5}$$

Using the Kriging equations (Matheron 1963), given a set of observations $Y_2(X)$ the predictive mean (5.6) and the predictive variance (5.7) at location x can be derived in closed form.

$$\hat{Y}(x) = k_{(x,X)}^T K_{(X,X)}^{-1} Y_2(X) \tag{5.6}$$

$$\hat{K}(x) = \nu k_{(x,x)} - \nu k_{(x,X)} K_{(X,X)}^{-1} k_{(X,x)} \tag{5.7}$$

Combining everything in this section so far, given some observations at location X, and values θ, (5.3) can be rewritten to include the predictive mean and predictive variance as (5.8).

$$Y_N \sim (2\pi\nu)^{-\frac{N}{2}} |\hat{K}_N|^{-\frac{1}{2}} e^{-\frac{\frac{1}{2}(Y_N - \hat{Y}(x))^T \hat{K}_N^{-1} (Y_N - \hat{Y}(x))}{\nu}} \tag{5.8}$$

Draws of Y_N conditioned on previously observed data and estimates of θ and ν, are shown in 5.4. The mean function shown is the same as the predictive mean (5.6), and the 90% intervals are computed from the normal quantile function using the predictive variance (5.7).

The take away from Figure 5.4 is that the results from GP regression include a closed form solution for uncertainty of the fit along with a typical expected value from a regression technique. The ability to quantify uncertainty in closed form has allowed for the use of active learning experimental designs for improved prediction (Seo et al. 2000).

For the previous examples in this section, it was assumed that the data generating mechanism is deterministic. Doing so provides more clear visuals. When Y_N is observed with noise, it can be said that $Y_N = f(x) + \epsilon$. To model ϵ, where noise normally distributed distributed with a mean of zero and constant across X, a *nugget* parameter is added to the diagonal entries of K_N.

This section is meant to be an overview of Gaussian processes, with just enough details for the reader to understand the Expected Improvement algorithm in §5.3.3. Much information is left out of this section. The methods in this chapter utilize maximum likelihood estimation (MLE) of model parameters and the Gaussian covariance function. For further details on model parameter estimation and alternate covariance functions see the excellent texts of C. K. Williams and Rasmussen (2006) and Gramacy (2020).

5.3.2 Shrinking Core Model

The shrinking core model (Yagi and Kunii 1955; Gbor and Jia 2004) can be used to model a leaching process. The version used to simulate data is shown

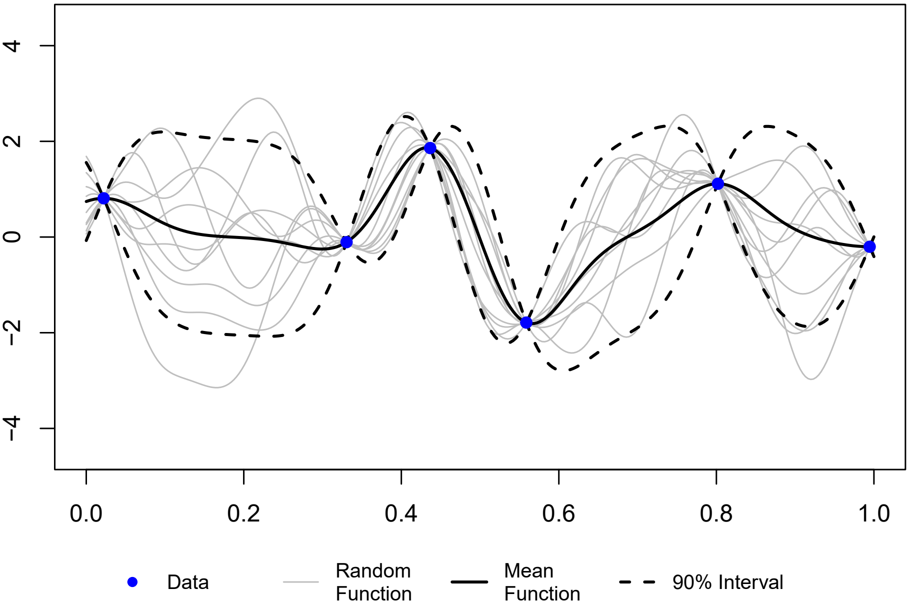

Figure 5.4: A set of random functions generated from a GP conditioned on data observations, with the mean function and uncertainty quantification via the 90% interval for the predictive variance shown.

in Equation (5.9), where X_{Ce} is the recovery of Cerium as a fraction, C_{Ab} and C_{zb} are the concentrations of chemicals A and z respectively, b_A and b_z are the moles of solid consumed per mole of A and z reacted, k_A and k_z are chemical reaction rates, ρ is the molar density of the solid, and r is particle radius.

$$X_{\mathrm{Ce}} = \left(\frac{b_A k_A C_{Ab} + b_z k_z C_{zb}}{\rho r} \right)^3 \tag{5.9}$$

Parameter values used to simulate data are shown in Table 5.1, and the resulting response surface is shown in Figure 5.5. Notably, the response surface in Figure 5.5 monotonically increasing with increasing acid and additive concentrations, and therefore optimization is trivial. Furthermore, most models the goal of this chapter is to illustrate the optimization of a process where a model does not fully capture the dynamics.

To complicate the optimization a bias function is added to the response surface generated using the shrinking core model. Responses become $y_f = y_m(x) + b(x)$, where y_f are field data observations, $y_m(x)$ is the shrinking core model, and $b(x)$ is the bias function. The bias function is shown in Equation (5.10), and the response surface is shown in Figure 5.6.

Table 5.1: Parameters of the shrinking core model.

Symbol	Value	Units
b_A	3	mol/mol
k_A	0.0008	m/s
C_{Ab}	variable acid concentration	mol/m^3
b_z	3	mol/mol
k_z	0.00001	m/s
C_{zb}	variable additive concentration	mol/m^3
ρ	48317	mol/m^3
r	0.0001	m^3

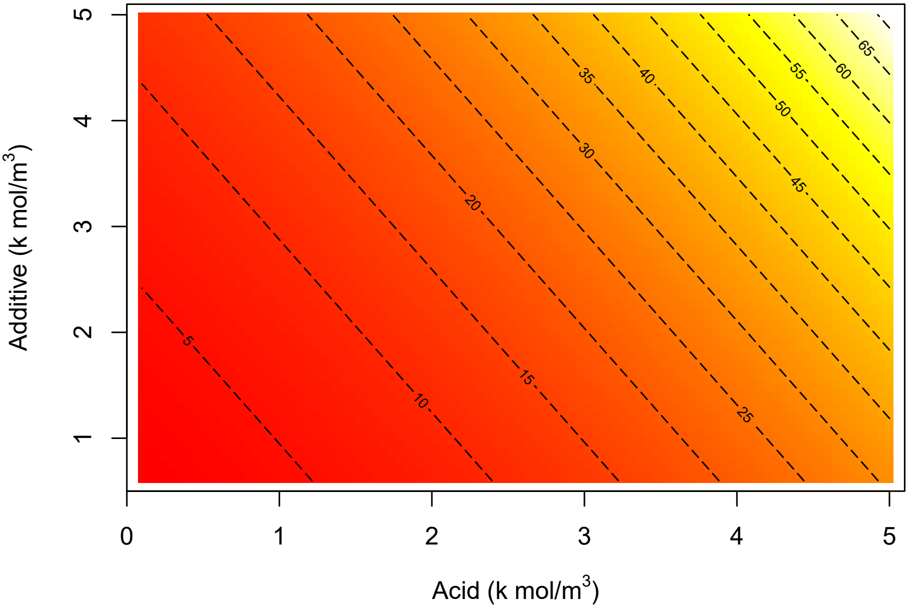

Figure 5.5: Shrinking core response surface.

$$\begin{aligned}
\eta_{Ab} &= \frac{C_{Ab} - 100}{5000 - 100} \\
\eta_{zb} &= \frac{C_{zb} - 600}{5000 - 600} \\
b(x) &= 0.088\phi\left(\frac{\eta_{Ab} - 0.3}{\sqrt{0.02}}\right)\phi\left(\frac{\eta_{zb} - 0.5}{\sqrt{0.02}}\right) + \\
&\qquad 0.015\phi\left(\frac{\eta_{Ab} - 0.8}{0.1}\right)\phi\left(\frac{\eta_{zb} - 0.7}{0.1}\right)
\end{aligned} \tag{5.10}$$

Lastly, to more accurately resemble a real process, random independent normally

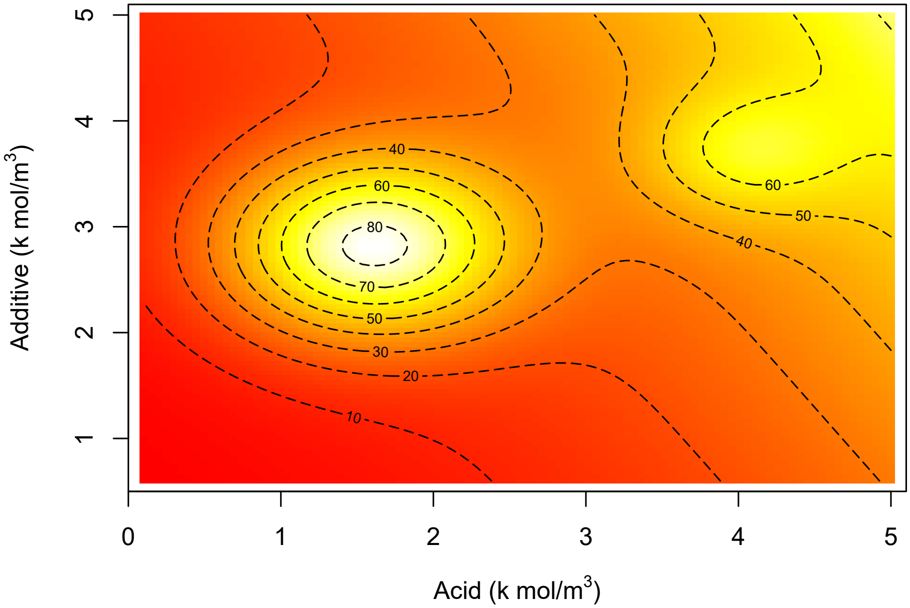

Figure 5.6: Response surface with bias function.

distributed noise with mean zero and standard deviation of 0.05 or 5% is added to y_f. A noisy realization of the response surface is shown in Figure 5.7.

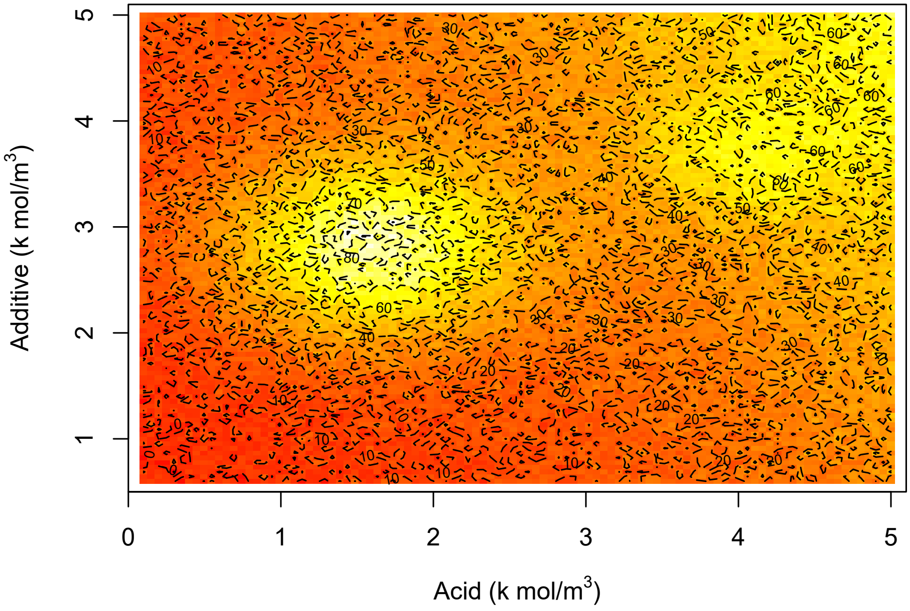

Figure 5.7: Response surface with noise added.

5.3.3 Expected Improvement

Stating the problem, when one wants to find the optimum as a maximum, they want to find some x that maximizes a function as in (5.11).

$$x = \underset{x}{\text{argmax}} f(x) \tag{5.11}$$

Utilizing the predictive mean of a GP surrogate to find the optimum as $f(x) = \hat{Y}(x)$ would be fairly simple, but would not include any of the available information about the uncertainty of the fit. Secondly, if one was to maximize the predictive mean of a GP fit $\hat{Y}(x)$, they would not be maximizing the true underlying function $f(x)$, but instead maximizing the expected value of $f(x)$ given a model structure, estimated hyperparameters, and most importantly previously observed data. If after finding the x in (5.11) for a given data set, data is then collected at x, it would be possible to see how accurate the model was as well as refit the surrogate model to include the new data point. An ideal optimization algorithm to use in conjunction with a GP surrogate would utilize the GP predictive variance and be able to adapt to newly collected data.

Expected Improvement (EI) (Schonlau 1997) juggles both the predictive mean and predictive variance in order to find the global optimum of a function. First improvement at location x is defined as $I(x)$ in (5.12).

$$I(x) = \max\{0, Y(x) - f^{\text{BOV}}\} \tag{5.12}$$

Where $Y(x)$ is a random draw from a GP surrogate conditioned on a data set and f^{BOV} is the best observed value (BOV) from a function. If optimizing a leaching experiment, after running 10 tests, f^{BOV} would be the best observed value from the set of 10 experiments. After formally defining improvement as (5.12), it is possible to define probability of improvement at location x (PI(x)), given a set of data D as Equation (5.13).

$$PI(x) = P(I(x) > 0|D) = P(Y(x) > f^{\text{BOV}}|D) \tag{5.13}$$

The statistical expectation of any function is $\quad [f(x)] = \int_{-\infty}^{\infty} f(x)p(x)dx$. Setting $f(x)$ equal to (5.12) it is possible to approximate the expectation of improvement using numerical techniques. However, when a GP surrogate is used, the expectation of improvement, or EI, can be solved for in closed form. EI for maximization is shown in Equation (5.14), where $\hat{Y}(x)$ is the predictive mean (5.6) and $\hat{k}(x)$ is the predictive variance (5.7) at location x.

$$\text{EI}(x) = (\hat{Y}(x) - f^{\text{BOV}})\Phi\left(\frac{\hat{Y}(x) - f^{\text{BOV}}}{\hat{k}(x)}\right) + \hat{k}(x)\phi\left(\frac{\hat{Y}(x) - f^{\text{BOV}}}{\hat{k}(x)}\right) \tag{5.14}$$

Figure 5.8 illustrates how EI can be used with a GP surrogate and a data set. After fitting the GP surrogate to the data, the EI criteria is evaluated along X. In this case a set of X candidates drawn as a latin hypercube sample (McKay, Beckman, and Conover 1979) are each evaluated with the EI function (5.14). The X that shows the maximum expected improvement, is selected for evaluation or laboratory testing.

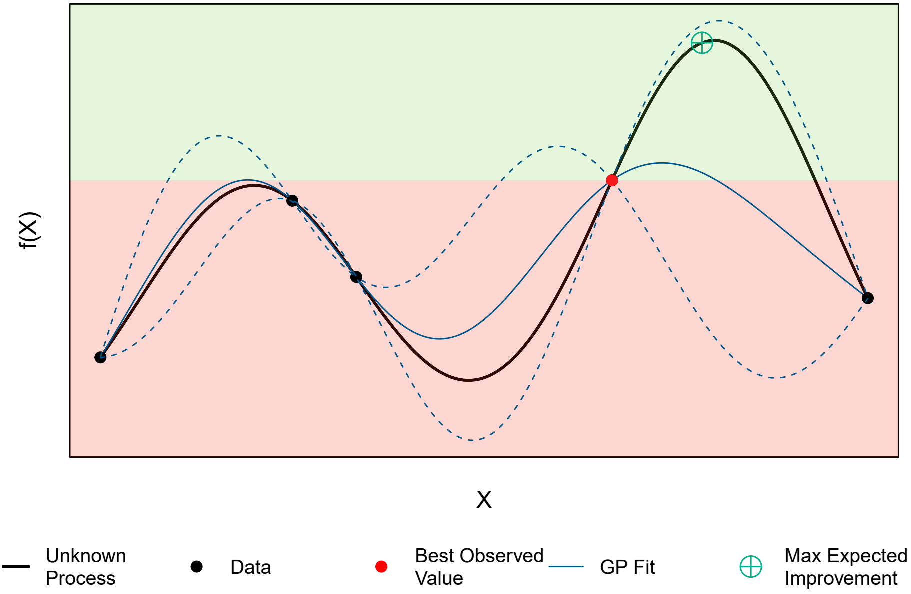

Figure 5.8: Illustration of GP fit highlighting the best observed value, and the maximum expected improvement as a trade off between probability of improvement and improvement of the predictive mean over the best observed value.

EI is able to leverage both the predictive mean and the probability of improvement to find a global optimum without the use of gradients. Additionally, EI tends to perform better maximizing $\hat{Y}(x)$ or probability of improvement alone (Gramacy 2020). In order to make the EI algorithm adaptable to newly acquired data, one would take a sequential design, or active learning, approach. First, a GP is fit to data set Y_N. Then EI is evaluated on a set of candidate locations, and the x which maximized EI is selected. Data y_{N+1} is collected at location x_{N+1}, and the original GP surrogate is updated to include this new data point. The process then repeats, sequentially adding data to the GP surrogate until x_{N+1} converges on an optimum.

To sequentially add EI points, one must start with an initial set of points. An eight point maximum entropy design (Shewry and Wynn 1987) is used as the initial design criteria, with the hopes that a substantial amount of information can be gained before sequential maximization via EI.

The evaluation of EI for the optimization of a noisy mineral process necessitates observing the performance repeatedly. Because each data set is different, performance will vary for repeated testing. Therefore, a Monte-Carlo experiment was run after initial observations.

5.4 Results

5.4.1 Single Experiment

To best observe the results of a single optimization experiment, noise is removed from the response surface. The maximum entropy initial design is shown in Figure 5.9. The purpose of the points, labeled 1 through 8, is to estimate GP hyperparameters, set the best observed value, and obtain conditional estimates of predictive variance and mean across the input space.

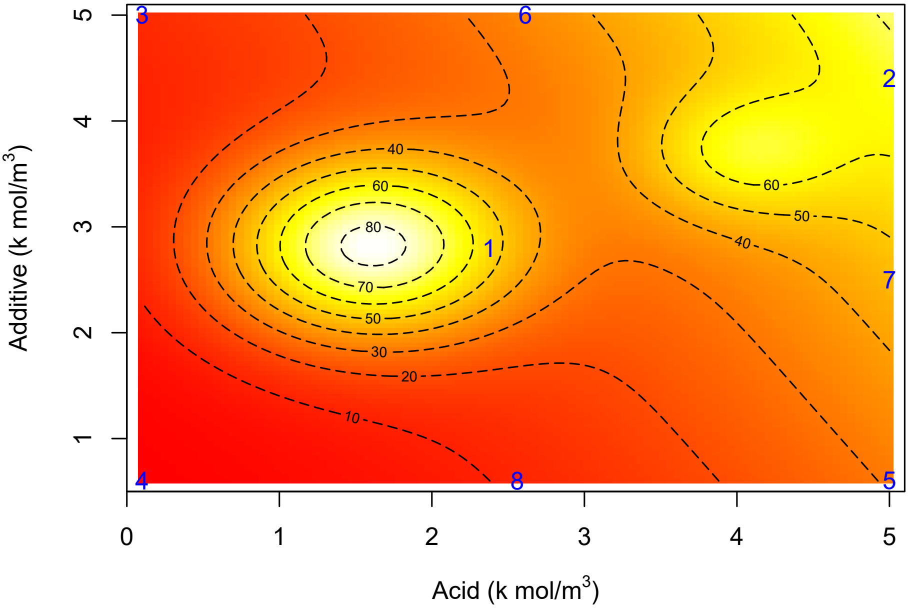

Figure 5.9: Maximum Entropy initial experimental design before EI maximization.

Then, the first EI point is added, shown as location 9 in Figure 5.10. The additional point is a balance between exploration and exploitation, but is not at the optimum.

Figure 5.11 shows the 15th point added to the set. The EI algorithm has converged on the global optimum of a complicated multi-modal process.

5.4.2 Monte-Carlo Experiment

As stated previously, results will vary with variable data. To observe the mean behavior of the EI algorithm a Monte-Carlo experiment was run, by repeating

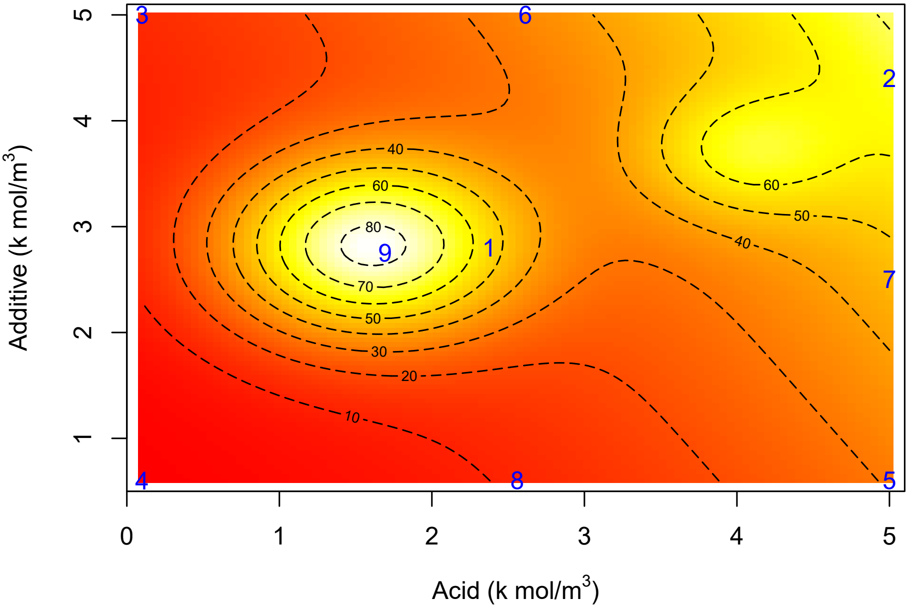

Figure 5.10: First point added to data set using EI.

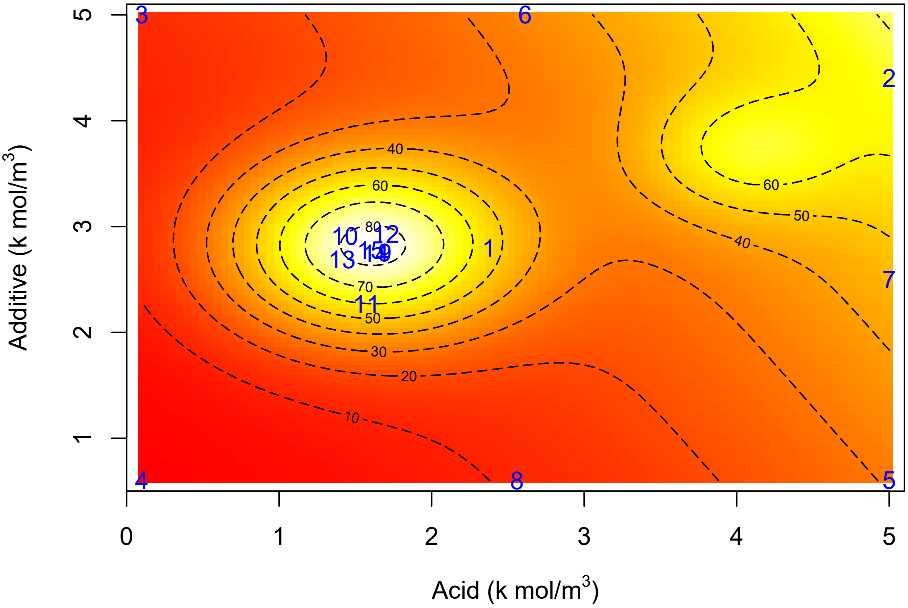

Figure 5.11: EI convergence on optimum.

the exercise in §5.4.1 100 times. For each experiment a new initial design and set of simulated data was generated, and EI was used to sequentially find and test the optimum. Because the standard deviation of the noise was 5%, a 2.5σ allowance was utilized to consider the best observed value to be the optimum. The true optimum is approximately 83%, and the process was considered optimized for any observed value above 71%.

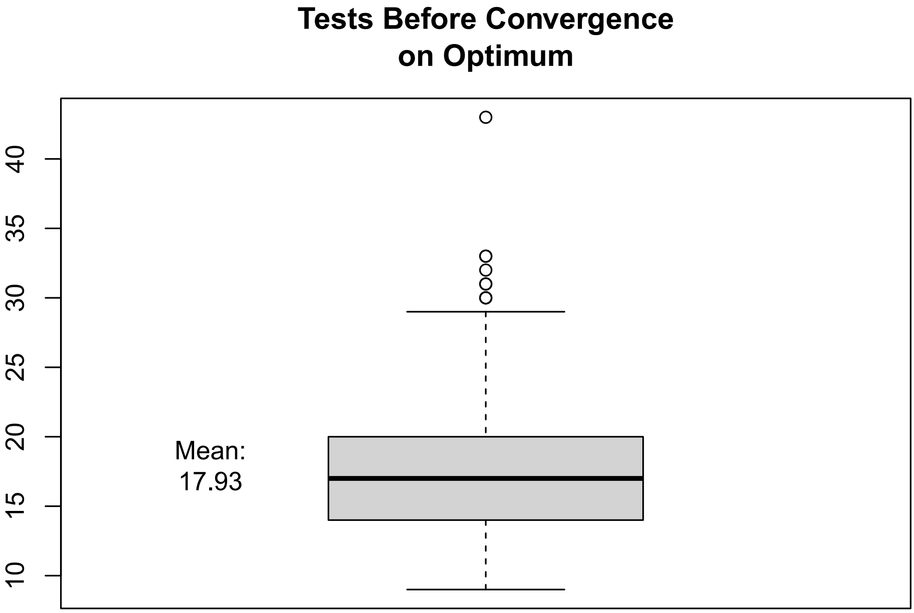

Figure 5.12: Results from Expected Improvement Monte-Carlo experiment.

Figure 5.12 shows the results of the Monte Carlo experiment. The minimum number of experiments EI required to test near the optimum was 9, and the maximum was 43. On average, EI found and tested the optimum in 17.93 tests.

5.5 Discussion and Conclusion

In this chapter, EI was utilized to maximize a noisy simulated chemical process. The simulation has a basis in chemistry theory, but did not capture the full relationship between the chemical concentrations and the *observations*. The flexibility of GP regression and the trade off between exploration and exploitation provided within the EI optimization criteria allowed for quick global optimization.

The use of EI is trivial for this simulation with negligible computation time. There is certainly utility in using EI to optimize more expensive simulators used in mineral processing. Moreover, this chapter aims to paint a picture of how EI can be used to optimize a real chemical process in a laboratory or pilot plant setting. EI was able to find and test the optimum of a complex and multi-modal response surface in relatively few *experiments*. Laboratory tests can be expensive and time consuming. When multiple factors are tested, a full factorial design can quickly become infeasible. Utilizing EI to sequentially and strategically test the optimum of an unknown process has the potential to significantly reduce laboratory costs.

While EI has clear advantages for optimizing multi-dimensional complex processes, some issues not yet touched upon can be problematic. One issue is taking a guess at the size of the initial design to use. One wants to use an initial experimental design large enough to obtain decent estimates of GP hyperparameters, but not so large as to needlessly expend experimental budget. Second, while these methods are able to optimize an *unknown* process, the choice of a stationary separable GP model assumes that the data will vary accordingly. For example, in a setting where there were step discontinuities, a non-stationary GP structure would be more appropriate. Last, but not least, the stopping criteria used in the set of experiments within this chapter is not realistic. This example necessitated a comparison to the known true optimum, while the truth is unknown in application. Before starting the set of experiments, setting some ϵ, where if $|y_n^{\mathrm{BOV}} - y_{n+1}^{\mathrm{BOV}}| < \epsilon$ researchers would assume convergence is necessary.

In this example, the EI algorithm was able to optimize a complex surface, on average using much less data than the methods in Al-Thyabat (2008) and Bu et al. (2016). Additionally, when using EI data is collected *at the optimum*, while the methods in Al-Thyabat (2008) and Bu et al. (2016) rely on the variable surrogate fit to find the optimum. Collecting data at the optimum location can inspire more confidence in the results, however a downside of EI is that the resulting model has less of a potential for interpretability. For example, the kinetic constant estimates found in Bu et al. (2016) may be of use to engineers for future projects. Depending on the purpose of the set of experiments, one may choose a method that simply finds the optimum, or another that allows for more flexibility and utility for prediction.

Overall, EI has the potential to perform better than classical response surface methods for the optimization of an unknown process. The example in this chapter is not indicative of the the limits for the EI algorithm. Additional work needs to be completed for optimization of a real process, although simulation results are promising. Future work in applying EI for the optimization of a mineral process, can incorporate heteroskedastic GP models (Binois, Gramacy, and Ludkovski 2018), and data from calibrated process simulators (see §6 for details on calibration).

Chapter 6

Active Learning for Bayesian Calibration of SX Simulators

Abstract

The Kennedy and O'Hagan (KOH) calibration framework uses coupled Gaussian processes (GPs) to simultaneously meta-model an expensive simulator (first GP), tune it's "knobs" (calibration inputs) to best match observations from a real physical/field experiment and correct for any modeling bias (second GP) when predicting under novel conditions (design inputs) in the field. Considering meta-models, or surrogates, for expensive computer simulation experiments in isolation, there are well-established methods for placement of design inputs for data-efficient planning of a simulation campaign. Examples include space-filling geometric principles or predictive optimality criteria like minimum integrated mean-squared prediction error (IMSPE). However, analogues for use within the coupled GP KOH framework, for both calibration and design inputs, are mostly absent from the literature. Intuitively, space-filling is inefficient because the computer model is most useful to KOH nearby promising settings of calibration inputs. Here we derive a novel, closed form IMSPE criteria for sequentially acquiring new simulator data in an active learning setting for KOH. We then illustrate that this intuition is correct: acquisitions space-fill in design space, but concentrate in calibration space. A closed form IMSPE precipitates a closed-form gradient, which we also provide, for efficient numerical optimization for acquisition of new runs. We then explore how how this new criteria leads to more a efficient simulation campaign compared to purely space-filling alternatives in benchmark problems. We conclude with a showcase of this new method on a motivating problem involving prediction of equilibrium concentrations of rare earth elements.

6.1 Introduction

Computer simulation experiments calibrated to real world observations can can assist in the understanding of complex systems. Examples include biofilm formation (L. R. Johnson 2008), radiative shock hydrodynamics (Goh et al. 2013), and the design of turbines (J. Huang et al. 2020). The canonical apparatus in this setting is due to Kennedy and O'Hagan (KOH, Kennedy and O'Hagan 2001). KOH models field-data from a physical system as the function of a computer simulation model plus an additional bias correction (see our review §6.2.3). Computer models are biased because they idealize physical dynamics and often have more dials or knobs, so-called *calibration parameters* or inputs, than can be controlled in the field. So KOH must juggle competing aims: furnish accurate, bias-corrected prediction for the real process in novel experimental conditions (i.e., design inputs, shared by both the physical/field apparatus and the computer simulation) while at the same time tuning good settings of calibration parameters. Moreover, limited simulation and field data necessitate meta-modeling. Toward this end, coupled Gaussian processes (GPs, C. K. Williams and Rasmussen 2006) are used as a surrogate (Gramacy 2020) for novel simulation, and to learn an appropriate bias correction.

This is hard to do, and in fact there are many recent papers that suggest that confounding between GP bias correction, GP surrogate, and tuning parameters creates an identification hazard (Bayarri, Berger, and Liu 2009; Higdon et al. 2004; Brynjarsdottir and O'Hagan 2014; Plumlee 2017, 2019; Gu 2019; Tuo and Wu 2015, 2016; Wong, Storlie, and Lee 2017). Nevertheless, the apparatus has proved highly useful for prediction. We thus take the framework as it is and focus our efforts here on data collection for efficient learning. Both experiments, field and simulated, must be carefully designed and modeled to make the most of limited resources.

Taken in isolation, the design for GP surrogates has a rich literature. Recipes range from purely random to geometric space-fillingness, such as via Latin-Hypercube Samples (LHS, McKay, Beckman, and Conover 2000) and minimax designs (M. E. Johnson, Moore, and Ylvisaker 1990). Closed form analytics from GP posterior quantities (again see §6.2.3) may leveraged to design optimality criteria, such as via maximum entropy or minimum integrated mean-squared prediction error (IMSPE) to develop designs (Sacks et al. 1989). These ideas may be applied as one-shot, allocating runs all at once, or sequentially via active learning (Seo et al. 2000), which can offer an efficient approximation to the one-shot approach due to submodularity (Wei, Iyer, and Bilmes 2015) properties while hedging against parametric specification of any (re-) estimated or fixed quantities. This active/sequential approach is generally preferred when possible. Ultimately the result is space-filling when variance/information criteria are measured globally in the input space. For a more thorough review see, e.g., Gramacy (2020) §4–§6.

Specifically within the coupled-GP KOH calibration framework, literature on simulation design for improved field prediction within the KOH framework is more limited. Most are one-shot or are focused on field design rather than computer model acquisition. Leatherman, Dean, and Santner (2017) built minimum IMSPE designs for combined field and simulation data. Arendt, Apley, and Chen (2016) used pre-posterior analysis to improve identification via space-filling criteria. Krishna et al. (2021) proposed one-shot designs for physical experimentation robust to modeling choices for bias correction. B. J. Williams et al. (2011) explored entropy and distance-based criteria in an active learning setting for the field experiment. Morris (2015) similarly study the selection of new field data sites, but in support of computer model development. None of these address a scenario where (new) field measurement is difficult/impossible, but new simulations can be run.

Ranjan et al. (2011) provide some insight along those lines, comparing reduction in field data IMSPE for surrogate-only designs. They found that new batches of simulations should involve design inputs closely aligned with the field data, paired with random calibration input settings. They stopped short of offering a recipe for choosing new acquisitions for simulation across both spaces simultaneously. We suspect that this may be because they did not have a closed form criteria that could easily be searched for new acquisitions. One such criterion is our main deliverable in this paper.

We study the coupled GP setup of KOH, define an IMSPE criteria for field-level predictions as a function of novel computer model runs, and show how the integral (the I in IMSPE) can be evaluated in closed form. Although similar analytic expressions for IMSPE have been developed in related contexts (e.g., Leatherman, Dean, and Santner 2017; Binois et al. 2019; Wycoff, Binois, and Wild 2021), we are unaware of any targeting computer model runs for improved KOH prediction. One advantage of having a closed form, as opposed to using quadrature or Monte Carlo integration, is that gradients can assist in optimization for new acquisitions. We additionally provide those in closed form so that finite differencing is not required.

Using our new KOH-IMSPE criterion, we reveal novel insights about which additional simulations lead to improved prediction. Rather than "matching" field data design inputs and being "random" on calibration parameters (Ranjan et al. 2011), we show that the criterion actually prefers being far previous simulations, but not too far from promising calibration parameters. In other words, it prefers to space-fill, modulo not entertaining calibration settings that are unlikely given current KOH model fits. We argue that this makes more sense than putting novel runs near field data, at least for prediction purposes. In any case, one quickly runs out of "matching" locations in the typical setup where simulation sizes dwarf field data observation, and "random" is an easy straw man to improve upon.

Although our contribution is largely methodological, we were motivated by an industrial application involving the extraction of rare Earth elements (REE), a significant portion of which are allocated to *high growth* green technologies, such as battery alloys (Goonan 2011; Balaram 2019). REEs include elements from the lanthanide series, Yttrium, and Scandium (Gosen et al. 2014). Liquid-liquid extraction, also known as solvent extraction (SX)m processes are often used to concentrate rare earth elements (C. K. Gupta and Krishnamurthy 1992) from natural and recycled sources. SX leverages the differing solubilities of various elements in organic (oil) and aqueous (water) solutions to make a separation.

Testing SX plants can be expensive due to the time required for the process to reach a steady state, and the difficulty of directly controlling some explanatory variables in a cost effective manner. These constraints render an active learning design for a SX plant at best difficult and at worst infeasible. Gathering data on elemental concentrations across the organic and aqueous phases is much easier than in greater generality. Prediction of SX equilibria clearly can benefit from the additional information provided from a simulator within KOH calibration. However, the high dimensionality of the simulator parameter space and necessity to numerically solve systems of differential equations prohibits exhaustive evaluation. Active learning to seek out promising runs for accurate real/field prediction with a limited simulation campaign is essential. We believe that our KOH-IMSPE is a perfect match.

With the ultimate aim of providing evidence in that real-data/simulation setting, the remainder of the paper is organized as follows. In §6.2, we review the elements in play: GPs, KOH, and sequential design. Our KOH-IMSPE criteria is developed and explored in §6.3. §6.4 provides implementation details and an empirical analysis of KOH-IMSPE in a sequential design/active learning context. §6.5 details our application for an experiment studying extraction of REEs. We conclude in §6.6 with a brief discussion.

6.2 Review of basic elements

KOH calibration couples GP surrogates with GP bias estimation, and our contribution involves active learning via IMSPE. Our review of these elements begins with GP regression, design for GPs via IMSPE, and the KOH apparatus with an eye toward their integration in §6.3.

6.2.1 Gaussian Process Regression

Generically, GP modeling means that a random variable of interest, like vector an $N \times 1$ vector of univariate responses $Y_N = Y(X)$ at a $N \times p$ design of inputs X, follows a multivariate normal (MVN) distribution $Y_N \sim \mathcal{N}_N(\mu, \Sigma)$. In a regression context, where we may apply a GP as a surrogate for computer model simulations $Y(X)$, it is common to take $\mu = 0$ and move all of the modeling "action" into the covariance structure Σ, which is defined by inverse distances between rows of X. For example,

$$\begin{gathered} Y_N \sim \mathcal{N}_N\left(0, \nu K(X)\right) \\ \text{where} \\ K(X)_{ij} = k(x_i, x_j) = \exp\left(-\sum_{l=1}^{s} \frac{(x_{il} - x'_{il})^2}{\theta_l}\right) + \delta_{(i=j)} g. \end{gathered} \tag{6.1}$$

Our specific choice of kernel $k(\cdot,\cdot)$ and the so-called "hyperparameterization" (via ν, g and θ) is meant as an example only. There are many variations, and our contributions are largely agnostic to these choices. When viewing (Y_N, X) as training data, the MVN in Eq. (6.1) defines a likelihood that can be used for inference for any unknowns. Textbooks cover many more of the details than are needed here (C. K. Williams and Rasmussen 2006; T. Santner, Williams, and Notz 2018; Gramacy 2020). Often, computer model simulations are deterministic, in which case the so-called a *nugget* parameter g is taken as zero (or small $g = \varepsilon > 0$ for better conditioned $K(X)$).

Regression, i.e., deriving a surrogate for new runs x, is facilitated by extending the MVN relationship in Eq. (6.1) for Y_N to $Y(x)$. Below, x is a $N' \times p$ matrix, but we may take $N' = 1$ for simplicity in many cases.

$$\begin{bmatrix} Y(x) \\ Y_N \end{bmatrix} \sim \mathcal{N}_{N'+N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \nu \begin{bmatrix} k(x,x) & k(x,X) \\ k(X,x) & K(X) \end{bmatrix} \right) \tag{6.2}$$

Observe that we are using $k(x, X)$ now, for cross-kernel evaluations between rows of x and rows of X such that $K(X) \equiv K(X, X)$. Then, standard MVN conditioning reveals $Y(x) \mid Y_N \sim \mathcal{N}_{N'}(\mu_N(x), \Sigma_N(x)))$, where

$$\mu_N(x) = k(x,X)^T K(X)^{-1} Y_N \qquad \hat{\Sigma}_N(x) = \nu(k_{(x,x)} - k(x,X)K(X)^{-1}k(X,x)). \tag{6.3}$$

These are known as the Kriging equations (Matheron 1963) in the geo-spatial literature, and they can be shown to provide the best linear unbiased estimator (T. Santner, Williams, and Notz 2018), among other attractive properties.

The top panel of Figure 6.1 shows a GP fit regressed onto a data set collected from an unknown function. The draws from the GP shown in grey. The analytical mean and variance can be simply evaluated as (6.3), and are included as the predictive mean and 95% confidence interval (CI). The bottom panel of Figure 6.1 shows two metrics for GP uncertainty quantification, and how these metrics can align with data locations. The value for predictive variance is plotted, analogous to the 95% CI shown in the top panel. For this deterministic example, predictive uncertainty is 0 for a location where data has already been observed. Integrated mean squared prediction error (IMSPE) is shown on the same plot, calculated as a function of x where x is augmented to the design matrix. Further details and comparison ensue in §6.2.2.

6.2.2 Integrated Mean Squared Prediction Error

One can imagine using those predictive equations (6.3) towards many ends: simply predicting at novel x not coinciding with training runs X; deducing where $Y(x)$ might be minimized, or so-called Bayesian optimization (BO, Jones, Schonlau, and Welch 1998); exploring which coordinates of x most influence $Y(x)$ (Marrel et al. 2009); to name just a few. Here we are interested in sequential experimental design, or active learning, to select new runs for an improved fit/prediction.

The simplest variation on this theme is to choose $x^\star_{N+1} = \text{argmax}_{x\in\mathcal{X}} \hat{\Sigma}_N(x)$, i.e., to maximize the predictive variance. Here x and $x^\star_{N+1}$ represent a single

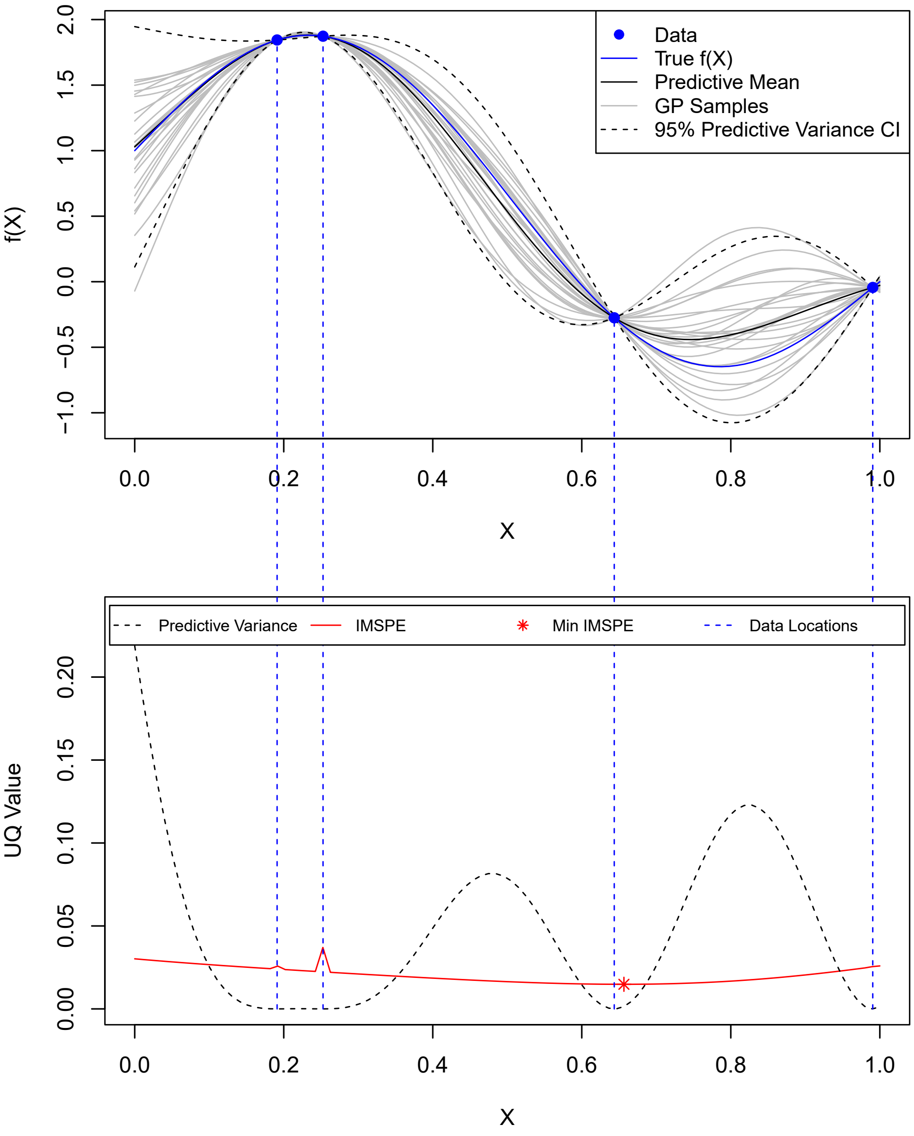

Figure 6.1: *top*: GP regression predictive mean, and predictive variance quantified as 95% CI, and GP draws, given a set of data. *bottom*: Gaussian process uncertainty quantification as related to data locations. For the same GP fit and data set as the top pannel, the quantity of predictive variance is plotted as a function of x. IMSPE values are shown as a function of x, where x is augmented to the initial data set and IMSPE is calcluated conditional on current hyperparameter estimates.

$N' = 1 \times p$ coordinate vector in the input space $\mathcal{X}$, but the idea is easily generalized to larger batches. MacKay (1992) showed that such acquisitions lead to so-called (approximate) maximum entropy designs in the context of neural network surrogates, and Seo et al. (2000) extended these to GPs naming the idea ALM. Despite their simplicity, convenient closed form and analytic gradients (not shown) for efficient library-based numerical optimization, ALM-based sequential designs are aesthetically limiting as they tend to concentrate new runs on the boundary of the input space $\mathcal{X}$. This is inefficient for prediction, the boundary is very important to the application at hand.

If prediction accuracy over the entirety of the input space $\mathcal{X}$, is desired then it may help to have a criteria that more squarely targets that objective. Sacks et al. (1989) proposed the integrated mean-squared prediction error (IMSPE) criterion, encapsulated generically as follows.

$$\mathrm{IMSPE}(X) = \int_{\mathcal{X}} \hat{\Sigma}_N(x)\; dx \tag{6.4}$$

Above, x is $1 \times p$ so that the integral is p-dimensional. One could use this criteria to chose an entire $N \times p$ design X in one shot by optimizing over all Np coordinates as in $X^\star = \mathrm{argmin}_{X \in \mathcal{X}^N} \mathrm{IMSPE}(X)$, or to simply augment X with a new row x_{n+1} sequentially, in an active learning setting. Independetly, Cohn (1994) developed a similar criteria neural network surrogates and one-at-a-time acquisition, approximating the integral as a sum; Seo et al. (2000) extended their work to GPs, dubbing this ALC. Several other authors have considered variations in-between and specific surrogate modeling settings (see §1).

Here we follow the mathematics laid out by Binois et al. (2019), who provided a closed form for IMSPE and gradient when $\mathcal{X} = [0,1]^p$. Their interest, like ours, was in active learning (i.e., acquisition of x_{N+1}) although their development is equally applicable to one-shot and batch design. The approach is at once elegant and practical in implementation, and consequently has spurred a cottage industry of variations (Wycoff, Binois, and Wild 2021; Cole, Christianson, and Gramacy 2021; Sauer, Gramacy, and Higdon 2021) of which our main contribution can be viewed as yet another. The derivation first relies on the trace equality $\mathrm{tr}(ABC) = \mathrm{tr}(BCA)$ to reorder the matrices in the predictive variance equation. The integral can be moved inside and outside of the matrix trace as both are linear operators. Because for this setting the elements of x are independent and occupy a uniformly rectangular space, we can let $W(X) = W(X,X) = \int_{[0,1]^d} k(X,x)k(X,x)^\top\; dx$ be an $N \times N$ matrix which has a closed form for several choices of kernel $k(\cdot,\cdot)$. A proof treating x as a random variable, in turn treating IMSPE as the expectation with respect to x of the predictive variance can be found in Binois et al. (2019). Given the integral

$W(X)$, it may be shown that

$$\begin{aligned}\text{IMSPE}(X) &= \int_{[0,1]^d} \nu k(x,x) - \nu k(x,X)^\top K(X)^{-1} k(X,x)\, dx \\ &= \nu - \int_{[0,1]^d} \text{tr}\left(\nu K(X)^{-1} k(X,x) k(x,X)\right)\, dx \\ &= \nu - \nu 1^\top \left(K(X)^{-1} \circ W(X)\right) 1\end{aligned} \tag{6.5}$$

To provide easier intuition for further derivations and maintain consistency with Binois et al. (2019) we simplify the expression using the trace identity $\text{tr}(AB) = 1^T(A \circ B)1$, where 1 is a vector of ones with a length equal to the number of rows of X, and $\circ$ is the Hadamard product, or element wise, product. Although it may be the case that estimates for ν involve (X, Y), the typical development presumes that we don't have Y-values yet, or at least not y_{N+1} in the active learning context. Consequently, it is equivalent to choose

$$x^\star_{N+1} = \underset{x \in [0,1]^d}{\text{argmin}} \sum_{i,j} \left(K(X)^{-1} \circ W(X)\right) \qquad \text{where } X \equiv [X; x^\top], \tag{6.6}$$

i.e., where X is augmented with the new row $x^\top$. In this way, the acquisition explicitly target predictive accuracy by minimizing mean-squared error. One consequence of this is that acquisitions avoid boundaries of $\mathcal{X} = [0,1]^d$ because that set is a mere manifold relative to the interior of the region being integrated over.

Returning to the bottom plot in Figure 6.1, a comparison between the points with the maximum predictive variance and how an additional data point will change the average predictive variance can be made. A point that is added at the left side of the plot, where the predictive variance is highest, will not necessarily minimize the integrated predictive variance if augmented to the current data set. A maximum predictive variance active learning criteria tends to add points at the border of the input space (Gramacy 2020). Adding a point at the location of highest variance, improves the predictive accuracy at that location, but not the average predictive variance across the input space. In fact, the point which will minimize IMSPE, shown on the plot, can at times be near other observations.

6.2.3 Simulator Calibration

A simple example of a simulator, taken from Gramacy (2020), would be a physics based model that predicts how long a ball would take to drop from a given height, neglecting wind resistance. The model, $t = \sqrt{2h/g}$, can be freely evaluated for positive values of both h, and g. However, in the real world h is a variable that can be easily controlled, while acceleration due to gravity g is a *tuning parameter* for the model and is not something that would change between experiments. Calibration would be the act of estimating g using a data set containing the time for a ball to drop to the ground from various heights. We want to estimate some constant g, to predict values for t. To learn about the simulator for calibration, and how the output varies with changes in g, it is necessary to evaluate the model for differing values of g.

In thinking of a simulator, or computer model, in this way $t = f(h, g)$, we can think of the inputs to the model as some subsets of X, equivalently $f(h, g) \equiv f(X)$. However, for a general case black box computer model we choose the notation $Y_m(x) = f(z, u)$, where $Y_m()$ is the output from a computer model, and $x \equiv [z, u]$ where z is the subset of x pertaining to the observable factor, equivalent to h in the simple example, and u is the subset of x pertaining to the calibration parameters, equivalent to g in the simple example. To simplify notation, future references to x and X include both the z and u subsets, while for the case when specificially z or u are specified, the derivation only is related to the relevant subset of x.

The Kennedy and O'Hagan calibration framework (KOH) (Kennedy and O'Hagan 2001) assumes real field observations at location z, $Y_f(z)$ are simulated from a computer model with $u = u^\star$, the *true* value of u, plus some unknown bias $b(z)$ function of z, and Gausian error ϵ (6.7). Given the computer model $Y_m(z, u^\star)$, $Y_f(z)$, and assuming normal error with a zero mean, $b(z)$ can be written as Equation (6.8).

$$Y_f(z) = Y_m(z, u^\star) + b(z) + \epsilon \tag{6.7}$$
$$\Longrightarrow$$
$$b(z) = Y_f(z) - Y_m(z, u^\star) + \epsilon \tag{6.8}$$

A GP prior on (6.8) can be used to estimate the unknown functional form $\hat{b}(z)$, with the covariance matrix K^B, given a set of field observations Y_f at data locations $Z_{N_f \times r}$ and the output of a simulator at the same locations $Z_{N_f \times r}$ at calibrated value $u = \hat{u}$.

$$b(z) \sim (2\pi\nu_B)^{\frac{-N_f}{2}} |K^B|^{-\frac{1}{2}} \exp\left(-\frac{1}{2\nu_B}(Y_f - Y_m(Z, \hat{u}))^T (K^B)^{-1} (Y_f - Y_m(Z, \hat{u}))\right) \tag{6.9}$$

If the computer model is expensive to evaluate, a GP emulator can be fit to a $N_m \times 1$ vector of observations Y_m at a set of locations $[Z_{N_m \times r}, U_{N_m \times p}]$. The observation Y_f can then be treated as a sum of Gaussians. The covariance matrix for all observations, Y_f and Y_m, can be written as (6.10).

$$\text{ov}\left(\begin{bmatrix} Y_f \\ Y_m \end{bmatrix}\right) = \begin{bmatrix} \nu_M K^M_{f,f} + \nu_B K^B & \nu_M K^M_{f,m} \\ \nu_M K^M_{m,f} & \nu_M K^M_{m,m} \end{bmatrix} \tag{6.10}$$

ν_M and $K^M_{\cdot,\cdot}$ are the GP scale parameter and covariance matrix for the computer model process. ν_B and K^B are the GP scale parameter and covariance matrix for the bias process. Here we take the computer simulation to be deterministic, meaning the nugget in $K^M_{\cdot,\cdot}$, $g_M = 0$, and use $b(x)$ to model homoskedastic noise by adding g_B to the diagonal of K^B. The subscript m indicates computer model data at locations $[Z_{N_m \times r}, U_{N_m \times p}$, while the subscript f indicates field data at locations $[Z_{N_f \times r}, 1_{N_f} \otimes \hat{u}_{1\times p}]$, where 1_{N_f} is a column vector of ones with length N_f and $\otimes$ is the Kronecker product. Important to note, only data locations in Z are utilized to evaluate K^B.

Kennedy and O'Hagan (2001) utilized fully Bayesian inference for joint estimation of computer model GP hyperparameters θ_M, bias hyperparameters θ_B, g_B, and the distribution of $\hat{u}$. However, such a flexible approach can cause identifiability issues with $\hat{u}$, and has motivated the use of the modularization approach to KOH (Tuo and Wu 2015). Modularization first trains the computer model GP on only Y_m. Then, the bias can be modeled as illustrated in Equation (6.11), allowing the bias GP hyperparameters and $\hat{u}$ to be jointly estimated conditional on $\hat{Y}_m(x)$.

$$b(z) = Y_f(z) - \hat{Y}_m(z, \hat{u}) + \epsilon \tag{6.11}$$

6.3 KOH-IMSPE

The statistical contribution of this work is to derive a closed form IMSPE sequential design criteria for acquiring additional computer simulation data for the purposes of minimizing IMSPE of field predictions. We term this IMSPE sequential design method for use within the KOH framework as KOH-IMSPE. First, augmenting a field prediction to the KOH covariance (6.10) provides (6.12), where we choose the notation such that x_f is the combined z, u space; $x_f = [z_{1\times r}, \hat{u}_{1\times p}]$.

$$\text{ov}\left(\begin{bmatrix} y_f(x_f) \\ Y_f \\ Y_m \end{bmatrix}\right) = \begin{bmatrix} \nu_M k^M_{(x_f,x_f)} + \nu_B k^B_{(z_f,z_f)} & \nu_M k^M_{(x_f,f)} + \nu_B k^B_{(z_f,f)} & \nu_M k^M_{(x_f,m)} \\ \nu_M k^M_{(f,x_f)} + \nu_B k^B_{(f,z_f)} & \nu_M K^M_{(f,f)} + \nu_B K^B & \nu_M K^M_{(f,m)} \\ \nu_M k^M_{(m,x_f)} & \nu_M K^M_{(m,f)} & \nu_M K^M_{(m,m)} \end{bmatrix} \tag{6.12}$$

Next, the predictive variance for the field data conditioned on both the field and simulator observations (6.16) can be derived in a similar fashion to Equation (6.3). To derive IMSPE in a comparable fashion to (6.5), first let:

$$\nu_M k_M = \nu_M \begin{bmatrix} k^M_{(f,x_f)} \\ k^M_{(m,x_f)} \end{bmatrix} \tag{6.13}$$

$$\nu_B k_B = \nu_B \begin{bmatrix} k^B_{(f,z_f)} \\ 0_{n_m} \end{bmatrix} \tag{6.14}$$

$$K_{M,B} = \begin{bmatrix} \nu_M K^M_{(f,f)} + \nu_B K^B & \nu_M K^M_{(f,m)} \\ \nu_M K^M_{(m,f)} & \nu_M K^M_{(m,m)} \end{bmatrix} \tag{6.15}$$

$$W_{\alpha,\beta} = \int k_\alpha k_\beta^T dx_f$$

Which produces the predictive variance from for field data form the covariance matrix (6.12), shown as (6.16).

$$\begin{aligned} \sigma^2_f(x_f) = \nu_M k^M_{(x_f,x_f)} + \nu_B k^B_{(z_f,z_f)} - \\ \left[\nu_M k_M + \nu_B k_B\right]^T \left[K_{M,B}\right]^{-1} \left[\nu_M k_M + \nu_B k_B\right] \end{aligned} \tag{6.16}$$

Then, the quadratic form in $\sigma^2_f(x_f)$ is expanded and then the trace identity $\mathrm{tr}(ABC) = \mathrm{tr}(BCA)$ is used to rearrange terms (6.17). Then we integrate over x_f, which is only contained within the $k.k^T_.$ matricies and IMSPE is calculated as a sum of these integrals weighted by the scales ν_M, ν_B and the elements of the covariance matrix $K_{M,B}$ as in Equation (6.18).

$$\begin{aligned} \text{IMSPE} &= \int \sigma^2_f(x_f) dx_f \\ &= \nu_M + \nu_B - \int \mathrm{tr}\Big(K^{-1}_{M,B}\left(\nu_M^2 k_M k_M^T\right. \\ &\qquad \left. + \nu_M \nu_B k_M k_B^T + \nu_M \nu_B k_B k_M^T + \nu_B^2 k_B k_B^T\right)\Big) dx_f \qquad (6.17) \\ &= \nu_M + \nu_B - 1^T K^{-1}_{M,B} \circ \left(\nu_M^2 W_{M,M} + 2\nu_M \nu_B W_{M,B} + \nu_B^2 W_{B,B}\right) 1 \qquad (6.18) \end{aligned}$$

For augmenting the existing computer model design with a new proposed computer model point $\tilde{x}_m = [\tilde{z}_m, \tilde{u}_m]$, the additional elements in (6.12) related to the computer model design are updated to include $\tilde{x}_m$, similar to (6.6). Just like implementation in Binois et al. (2019), derivatives can be provided for a gradient based search of the $\tilde{x}_m$ which minimizes IMSPE of the field data. The derivative of IMSPE with respect to element l of $\tilde{x}_m$ is shown in (6.19). The additional computer model point $\tilde{x}_m$ has no effect on $W_{B,B}$, and therefore $\frac{\partial W_{B,B}}{\partial(\tilde{x}_m)_l} = 0$.

$$\frac{\partial \text{IMSPE}}{\partial \tilde{x}_l} = -1^T \left(\frac{\partial K^{-1}_{M+1,B}}{\partial \tilde{x}_l} \circ (\nu_M^2 W_{M,M} + 2\nu_M \nu_B W_{M,B} + \nu_B^2 W_{B,B}) + \right.$$
$$\left. K^{-1}_{M+1,B} \circ \left(\nu_M^2 \frac{\partial W_{M,M}}{\partial \tilde{x}_l} + 2\nu_M \nu_B \frac{\partial W_{M,B}}{\partial \tilde{x}_l} \right) \right) 1 \tag{6.19}$$

6.3.1 Illustration

KOH-IMSPE is first demonstrated on a toy 1-D Z and 1-D U space with a sinusoidal computer model (6.20), the polynomial bias function (6.21), and field data generated as (6.22) where $\epsilon \sim \mathcal{N}(0, 0.1^2)$. The left plot of Figure 6.2 displays the computer model evaluated at $u = u^\star$ as well as $\mathbb{E}[Y_f(z)]$. The righthand side of Figure 6.2 shows the computer model evaluated at various settings for u.

$$y_m(z, u) = \sin(10z, u) \tag{6.20}$$
$$b(z) = 1 - \frac{1}{3}z - \frac{2}{3}z^2 \tag{6.21}$$
$$y_f = y_m\left(z, u^\star = \frac{\pi}{5}\right) + b(z) + \epsilon \tag{6.22}$$

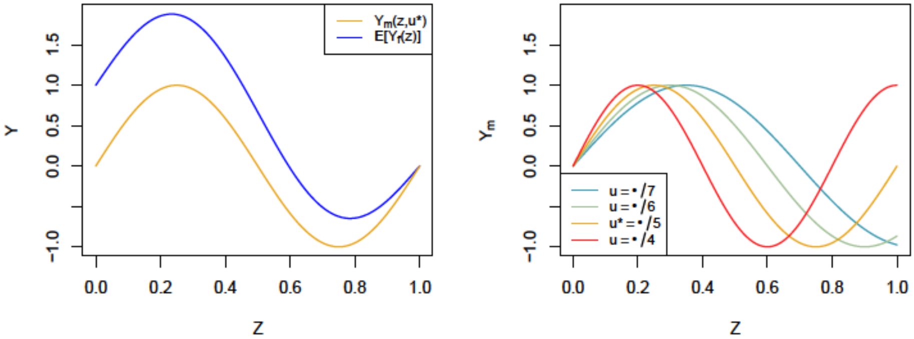

Figure 6.2: True surface, model surface with $u^\star$, and effect of varying u on the model surface.

Next an experiment was run to test KOH-IMSPE on the toy data generating mechanism. A 10 point, 2-D latin hypercube sample (LHS) (McKay, Beckman, and Conover 2000) was used for the initial computer model design in $[Z, U]$ space. Field data was collected as two replicates of five unique locations on an equally spaced grid on Z. Then, the modularization approach to KOH was used to find the maximum a-posteriori (MAP) estimate all hyperparameters and $\hat{u}$, where each $p(\theta_M) = \text{Gamma}(\frac{3}{2}, 2)$, $p(\theta_B) = \text{Gamma}(\frac{3}{2}, 5)$, and $p(u^\star) = \text{Beta}(2, 2)$. In total, 21 computer model points were added to the initial design.

Figure 6.3 shows KOH-IMSPE surface plots for $N_m = 10, 11, 12, 15, 20, 30$ as well as the initial model design, the points previously added via KOH-IMSPE, the location of $u^\star$ as a grey line, the location of field data given an estimate of $\hat{u}$, and the minimum KOH-IMSPE point for the provided N_m.

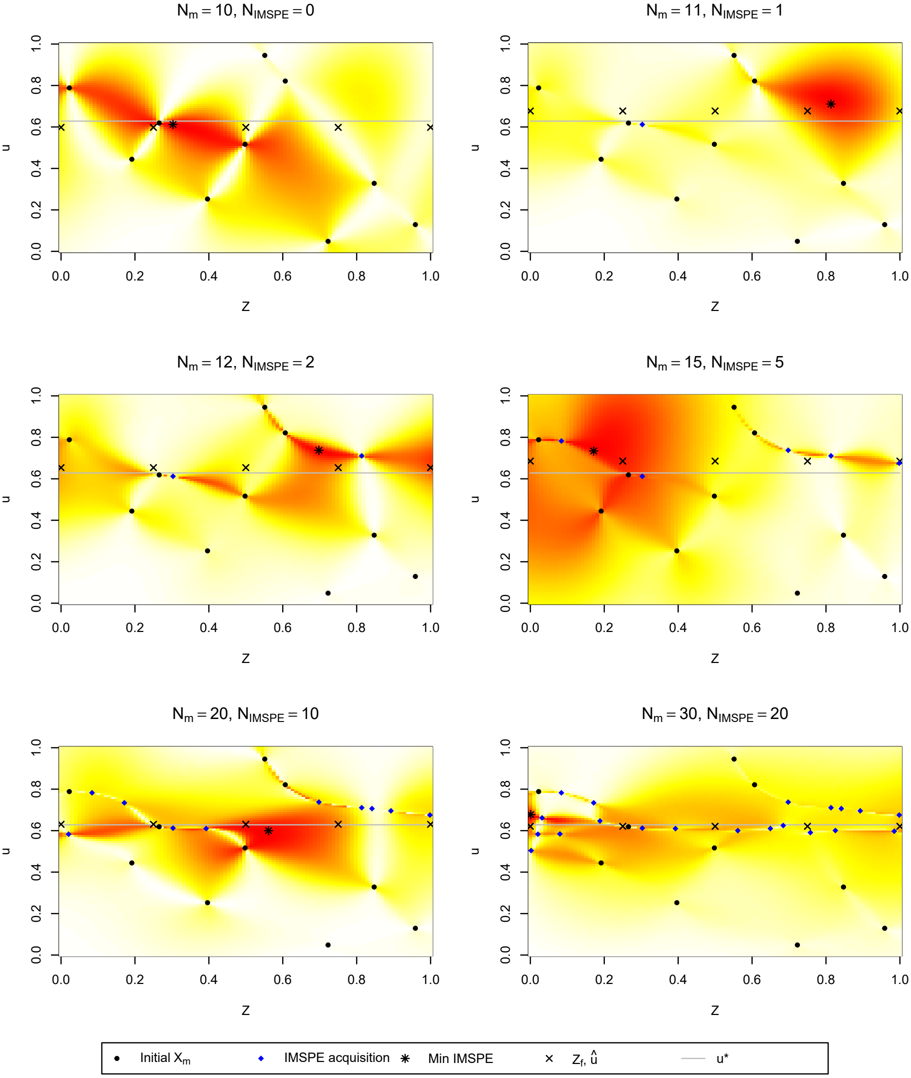

Figure 6.3: KOH-IMSPE surface in Z, U space as points are sequentially added to an inittial computer model design. Red indicates lower values and white/yellow indicates larger values.

Notably, Figure 6.3 shows computer model points often added near $\hat{u}$, although time to time KOH-IMSPE points can be more exploratory. Intuitively, this makes sense. If one wanted to better understand the response of a simulator for real world prediction, it would be sensible to acquire data points when the simulator is calibrated. However, the reason for this behavior is not immediately clear.

Digging into the derivatives of KOH-IMSPE provides some insight. First, we will use the identity $\frac{\partial U^{-1}}{\partial x} = -U^{-1}\frac{\partial U}{\partial x}U^{-1}$. Then the first term in Equation (6.19) can be rewritten as Equation (6.23).

$$\sum_{i,j} \frac{\partial K_{M+1,B}}{\partial \tilde{x}} \circ K_{M+1,B}^{-1} \left(\nu_M^2 W_{M+1,M+1} + 2\nu_M \nu_B W_{M+1,B} + \nu_B^2 W_{B,B}\right) K_{M+1,B}^{-1} \tag{6.23}$$

$K_{M+1,B}$ can be written as (6.24).

$$K_{M+1,B} = \nu_M \begin{bmatrix} K^M_{(f,f)} & K^M_{(f,m)} & k^M_{(f,\tilde{x})} \\ K^M_{(m,f)} & K^M_{(m,m)} & k^M_{(m,\tilde{x})} \\ k^M_{(\tilde{x},f)} & k^M_{(\tilde{x},m)} & k^M_{(\tilde{x},\tilde{x})} \end{bmatrix} + \nu_B \begin{bmatrix} K^B & 0_{N_f \times N_m} & 0_{N_f \times 1} \\ 0_{N_m \times N_f} & 0_{N_m \times N_m} & 0_{N_m \times 1} \\ 0_{1 \times N_f} & 0_{1 \times N_m} & 0_{1 \times 1} \end{bmatrix} \tag{6.24}$$

It is easy to see that differentiating (6.24) with respect to u produces a matrix of mostly zeros. Furthermore, for the Gaussian kernel, $\tilde{u} = \hat{u}$, $\frac{\partial K_{M+1,B}}{\partial \tilde{u}}$ can be written as (6.25). Specifically, the vector in the first row and third column of the block matrix layout, its transpose in the thrid row and first column are equal to zero when $\tilde{u} = \hat{u}$.

$$\frac{\partial K_{M+1,B}}{\partial \tilde{u}} : (\tilde{u} = \hat{u}) = \nu_M \begin{bmatrix} 0_{N_f \times N_f} & 0_{N_f \times N_m} & 0_{N_f \times 1} \\ 0_{N_m \times N_f} & 0_{N_m \times N_m} & \left(\frac{\partial k^M_{(m,\tilde{x}_f)}}{\partial \tilde{u}}\right)_{N_m \times 1} \\ 0_{1 \times N_f} & \left(\frac{\partial k^M_{(\tilde{x}_f,m)}}{\partial \tilde{u}}\right)_{1 \times N_m} & 0_{1 \times 1} \end{bmatrix} \tag{6.25}$$

Diving deeper, $\frac{\partial k^M_{(f,\tilde{x})}}{\partial \tilde{u}} = -\frac{2(\tilde{u}-\hat{u})}{\theta_M} \exp\left(-(\hat{u}-\tilde{u})^2/\theta_M\right)$. Therefore, in Equation (6.23), many terms in the sum are zero when $\tilde{u} = \hat{u}$. However, the fact that this portion of the $\frac{\partial \mathrm{KOH-IMSPE}}{\partial \tilde{u}} : (\tilde{u} = \hat{u})$ derivative is not equal to zero allows for some exploratory behavior.

The second term in Equation (6.19) differentiates each $W_{\cdot,\cdot}$. Similar to $\frac{\partial K_{M+1,B}}{\partial \tilde{u}}$ elements of $\frac{\partial W_{M+1,M+1}}{\partial \tilde{u}}$ and $\frac{\partial W_{M+1,B}}{\partial \tilde{u}}$ unrelated to $\tilde{u}$ are zero. Furthermore, for the Gaussian kernel, when $\tilde{u} = \hat{u}$ all elements of each $\frac{\partial W_{\cdot,\cdot}}{\partial \tilde{u}}$ are equal to zero. The location of minimum IMSPE regarding U space is therefore a trade off between $\tilde{u} - \hat{u}$ and the correlation between $\tilde{u}$ and existing model data locations.

6.4 Implementation and benchmarking

6.4.1 Details

Code was written in `R` to implement KOH-IMSPE for all examples. Functions from both the `laGP` (Gramacy 2016) and `hetGP` (Binois and Gramacy 2021a) packages were used to find MAP estimates of hyperparameters, to build covariance matrices, and evaluate integrals when appropriate. The modularization approach to KOH (Bayarri, Berger, and Liu 2009) was used in an effort to reduce identifiability issues. Independent inverse gamma priors were used for all lengthscale parameters as well as the nugget for the bias model. A Beta(2,2) prior was used for each element of u. To reduce computation time and simplify implementation relative to fully Bayesian Markov Chain Monte-Carlo parameter estimation, the MAP of of all hyperparameters and u were found using `optim(...)` routines.

Integration to find each $W_{\cdot,\cdot}$ can at times be tricky due to the use hyperparameters from two different kernels and the differing treatment between when integrating over X and U. When covariance functions are of the same form and utilize the same hyperparameters, integrating with respect to the dimensions of Z_f is equivalent to the methods provided in Binois et al. (2019). Because we utilize the point estimate $\hat{u}$ for both predictive and observed field data and a distance based kernel, integrating over U_f simplifies into a value of 1 or the multiplication of covariance functions. Integration details are provided in Appendix C.1.1.

To reduce computation time in the search of $\tilde{x}$ which satisfies (6.6), gradients were found analytically and supplied to `optim(..., method = "L-BFGS-B")`. Gradients of all $W_{\alpha,\beta}$ matricies are also provided in C.1.1. Additionally block matrix inversion (Bernstein 2009) of $K_{M+1,B}$ was utilized to remove the necessity of inverting the full covariance matrix for every KOH-IMSPE evaluation. Details for block matrix inversion and $\frac{\partial K^{-1}_{M+1,B}}{\partial \tilde{x}}$ are provided in Appendix C.1.2.

In implementation, we encountered numerical problems, particularly related to finding $K^{-1}_{M+1,B}$, which led to computation of negative KOH-IMSPE values as well as errors reported from the output of `optim()`. These problems were solved by using a few numerical tricks. First we took the average of the computed symmetric matrix $K^{-1}_{M,B}$ as $K^{-1}_{M,B} = \left(K^{-1}_{M,B} + \left(K^{-1}_{M,B}\right)^T\right)/2$. Second, when computing the inverse of matrix A, multiplied by vector b, instead of computing `solve(A) %*% b` we calculate `solve(A,b)`. The latter function has the advantage of solving for n^2 terms instead of n^3 terms. Big data GPs may further benefit from the similar, although more complex, methods in (Gardner et al. 2018). Lastly, to improve the stability of the `optim()` function we elect to minimize log(KOH – IMSPE) with an appropriate update to the gradient and set `optim(...,  control = list(pgtol = 0.5))`.

Figure 6.3 illustrates how the IMSPE surface can be fairly flat. To make the best of a random start optimization scheme, we first evaluate KOH-IMSPE for a number of candidate points generated via LHS. We then use `optim()` as a local optimizer on the minimum 7.5% of KOH-IMSPE evaluations. The choice of 7.5% is an arbitrarily chosen compromise between computation time and thoroughness of the search.

6.4.2 Sinusoid

To validate KOH-IMSPE, the method was tested and compared to a space filling LHS design and a design where data was acquired randomly using a Monte-Carlo (MC) experiment. We first compared performance using the sinusoid data generating mechanism introduced in §6.3.1 with $u^\star = \frac{\pi}{5}$ and $\epsilon \sim \mathcal{N}(0, 0.2^2)$. 1,000 MC repetitions were completed during this evaluation.

The constants between each MC repetition included the priors on the model lengthscale elements $p(\theta_M) = \text{Gamma}(3/2, 2)$, and the bias hyperparameters $p(\theta_B) = \text{Gamma}(3/2, 5)$, $p(g_B) = \text{Gamma}(3/2, 7)$. Two replicates of 10 field data points, evenly spaced on a grid, were collected, providing 20 field observations in total. For each design method in each experiment the initially $N_m = 10$ and the final computer model design size was $N_m = 50$.

Between each MC repetition values of field observations varied due to the random noise added to the expected response. Additionally, the initial design for computer model varied. A 10 point random subset from a 50 point LHS was used as the initial design for each method to ensure all three methods had the same starting point within a MC repetition. For the LHS design method the remaining 40 points from the initial 50 point LHS were sequentially added to the computer model design. The KOH-IMSPE method added points to the initial 10 point computer model design based on the KOH-IMSPE criteria. The random design method sequentially added uniformly random points on X to the computer model design. In each method, after a additional computer model point was acquired, estimates for GP hyperparameters and $\hat{u}$ were updated and root mean squared error (RMSE) on an $n_t = 100$ point LHS testing set sans noise. RMSE was calculated as shown in Equation (6.26), where $\quad [y_F(x_i)]$ was obtained from the data generating mechanism and $\hat{y}_{\text{GP}}(x_i)$ is the GP predictive mean at location x_i. To improve convergence of GP hyperparameters, the search for the MAP estimates of θ_M, θ_B and g_B were initialized at the previously estimated values when available.

$$\text{RMSE} = \sqrt{\frac{1}{n_t}\sum_{i=1}^{n_t}(\quad [y_F(x_i)] - \hat{y}_{\text{GP}}(x_i))^2} \tag{6.26}$$

The results are shown in Figure 6.4 with the left plot containing the mean RMSE and 90% quantiles over the 500 MC experiments for each method. The box-plot in Figure 6.4 shows a box-plot for $N_m = 22$, chosen because when $N_m = 22$ there is the largest difference between the the mean best performance and worst performance over the three methods.

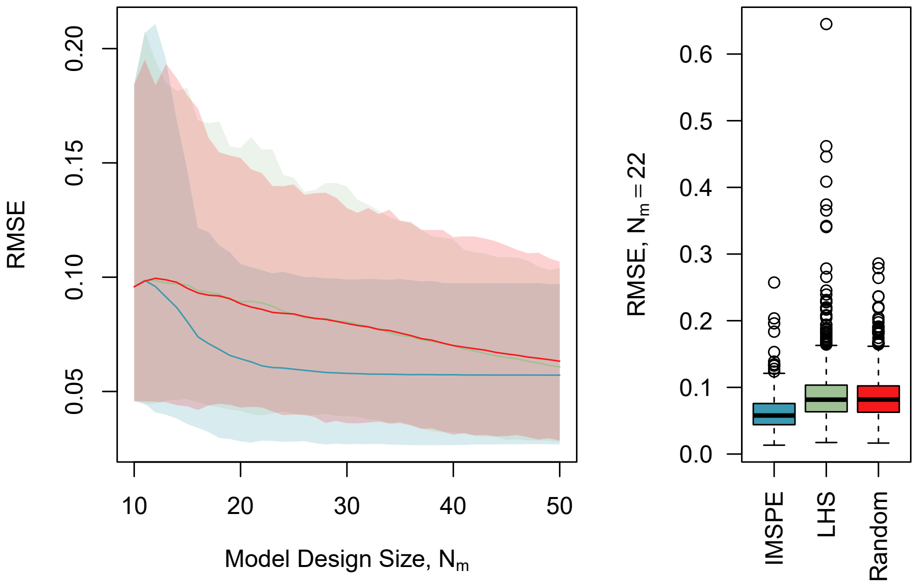

Figure 6.4: Mean RMSE and 90% quantiles for sinusoid data generating mechanism using IMSPE design, LHS deisgn, and random design.

Figure 6.4 shows high variability in the results, possibly due to the large amount of noise relative to the signal. However, on average KOH-IMSPE performs better than the competitors with less variability in the results.

6.4.3 Four Dimensional Problem

As a second test of the methodology, a similar MC test scheme is repeated on a larger toy problem with a 2-D X and 2-D U space. The problem is taken as a simplified version of the example found in Section 3.1 of Goh et al. (2013), and originates from (Bastos and O'Hagan 2009). The computer model (6.27), bias function (6.28), and field data generating mechanism (6.29) are shown below. Field data is generated using $u_1^\star = 0.2$, $u_2^\star = 0.1$, and $\epsilon \sim \mathcal{N}(0, 0.25^2)$. A plot of the response surface at $u = u^\star$ is shown in Figure 6.5.

$$y_m(z,u) = \left(1 - \exp\left(-\frac{1}{2z_2}\right)\right) \frac{1000u_1z_1^3 + 1900z_1^2 + 2092z_1 + 60}{100u_2z_1^3 + 500z_1^2 + 4z_1 + 20} \tag{6.27}$$

$$b(z) = \frac{10z_1^2 + 4z_2^2}{50z_1z_2 + 10} \tag{6.28}$$

$$y_f = y_m\left(z, u^\star = [0.2, 0.1]\right) + b(z) + \epsilon \tag{6.29}$$

The MC experiment was run in a similar fashion to the methods in 6.4.2. 100 MC repetitions were used for the experiment comparing KOH-IMSPE, LHS, and random computer model designs. For each method, an initial $N_m = 30$,

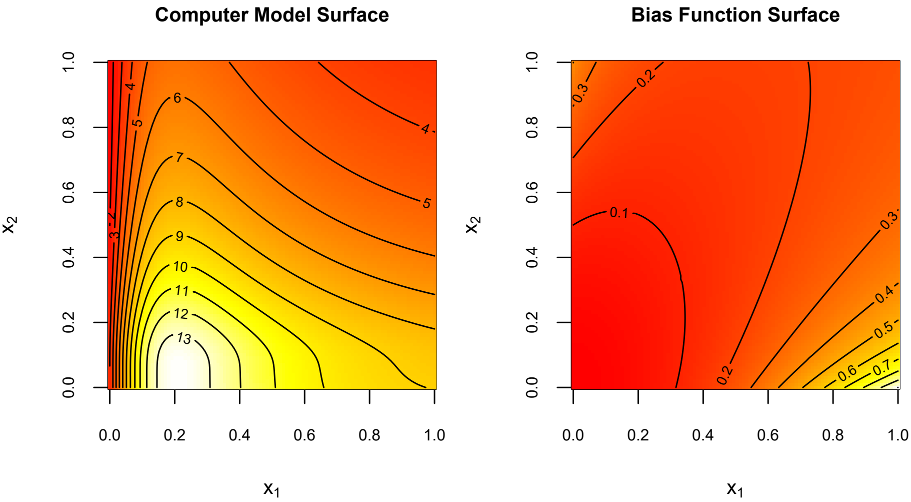

Figure 6.5: Response surface of the computer simulator for $u^\star = [0.2, 0.1]$ (left) and the bias function (right)

and 100 computer model points were added until $N_m = 130$. New field data was collected every MC repetition. Field data was collected at 25 unique locations on an evenly spaced grid in $X \in [0, 1]^2$. Two replicates at each field data location were used making $N_f = 50$.

The initial computer model design for all three methods is a 30 point subset of a 130 point LHS, unique to each MC repetition. The LHS design method sequentially added each of the remaining 100 points of the initial 130 point LHS, while the random design method added uniformly random points in $[0, 1]^4$.

Priors for hyperparameter elements were held at a constant and chosen to be $p(\theta_M) = \text{Gamma}(3/2, 5/4)$, $p(\theta_B) = \text{Gamma}(3/2, 5/2)$, $p(g_B) = \text{Gamma}(3/2, 1/20)$. After each computer model data point was acquired, hyperparameter estimates and $\hat{u}$ were updated, and RMSE was calculated using the expectation of 1000 field data points chosen using a LHS unique to each MC repetition. The hyperparameter initialization strategy used in 6.4.2 was used again to improve convergence.

Figure 6.6 shows the results over the 100 MC repetitions. The left plot shows mean RMSE and 90% quantiles, while the right box-plot shows the variability in the results at the N_m when there is the largest range in the mean RMSE values. The performance difference for KOH-IMSPE in this higher dimensional example is more obvious than the sinusoid problem. Variance in RMSE over the MC reps is much less than the space filling alternatives.

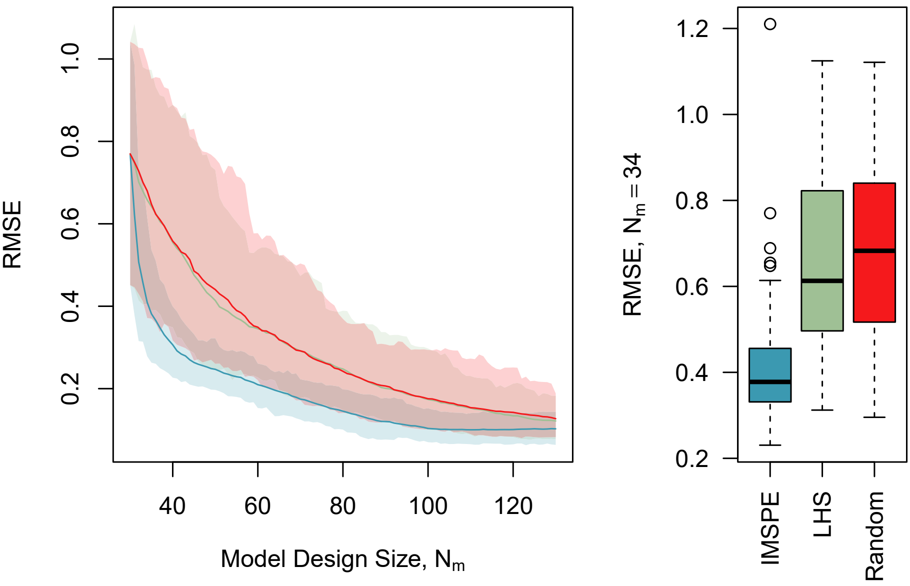

Figure 6.6: Mean RMSE and 90% quantiles for $X \in [0,1]^2$, $U \in [0,1]^2$ data generating mechanism for IMSPE design, LHS deisgn, and random design.

6.5 Solvent Extraction Kinetic Modeling

There are many ways to model the chemical reactions which take place as part of a solvent extraction process. The law of mass action is shown by Equation (6.30) and Equation (6.31), taken from L. Chen et al. (2010), and can be used to model chemical kinetics. Equation (6.30) specifies a single reaction with reactants which could be $R_1, R_2, \ldots, R_m$, which have stoichiometric coefficients $r_1, r_2, \ldots, r_m$, and products labeled similarly from a set of size n. k is the rate constant of the reaction. The reaction rate of Equation (6.30) is given in Equation (6.31) as a differential of a reactant or product concentration with respect to time. A system of differential equations for multiple reactions and all reactants can be derived by summing over all reaction rates found using Equation (6.31).

$$r_1R_1 + r_2R_2 + \cdots + r_mR_m \xrightarrow{k} p_1P_1 + p_2P_2 + \cdots + p_nP_n \tag{6.30}$$

$$r = -\frac{1}{r_i}\frac{d[R_i]}{dt} = \frac{1}{p_j}\frac{d[P_j]}{d_t} = k\prod_{l=1}^{m}[R_l]^{r_l} \tag{6.31}$$

For a rare earth element, or trivalent metal, in either the aqueous or organic phases, the reaction shown in Equation (6.32) (C. K. Gupta and Krishnamurthy 1992) can be used to specify a set of differential equations which model elemental concentrations in solution. $[\mathrm{RE}^{3+}]_{\mathrm{aq}}$ is a rare earth ion in the aqueous phase, $[(\mathrm{HA})_2]_{\mathrm{org}}$ is an organophosphorous acid, such as tri-butyl phosphate, in the organic phase, $[\mathrm{RE(HA_2)_3}]_{\mathrm{org}}$ is a rare earth element bound to the organophosphorous acid in the organic phase, and $[\mathrm{H^+}]_{\mathrm{aq}}$ is a hydrogen ion in the aqueous phase released as part of the forward reaction.

$$[\mathrm{RE^{3+}}]_{\mathrm{aq}} + 3[(\mathrm{HA})_2]_{\mathrm{org}} \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} [\mathrm{RE(HA_2)_3}]_{\mathrm{org}} + 3[\mathrm{H^+}]_{\mathrm{aq}} \tag{6.32}$$

For a set of trivalent metal ions $\mathrm{M}_1, \mathrm{M}_2, \ldots, \mathrm{M}_p$ ions in solution, the following constraints can be implemented to mass balance the set of differential equations.

$$[\mathrm{H^+}]_t = [\mathrm{H_2A_2}]_t - [\mathrm{H^+}]_0 + [\mathrm{H_2A_2}]_0 \tag{6.33}$$

$$[\mathrm{H_2A_2}]_t = [\mathrm{H_2A_2}]_0 + 3\sum_j^p [\mathrm{M}_j(\mathrm{HA_2})_3]_0 - 3\sum_j^p [\mathrm{M}_j(\mathrm{HA_2})_3]_t \tag{6.34}$$

$$[\mathrm{M}_j(\mathrm{HA_2})_3]_t = [\mathrm{M}_j(\mathrm{HA_2})_3]_0 + [\mathrm{M}_j^{3+}]_0 - [\mathrm{M}_j^{3+}]_0 \tag{6.35}$$

Where (6.33) is the balance around $\mathrm{H^+}$, (6.34) is the balance around $\mathrm{HA_2}$, and (6.35) is the balance around each of the j metals M_j. Additionally the subscripts of brackets $[\cdot]_t$ and $[\cdot]_0$ indicate concentrations at time t and initial conditions respectively. The constraints above can be used to produce a set of p differential equations, substituting the above constraints in for $[\mathrm{H^+}]$, $[\mathrm{H_2A_2}]$, and $[\mathrm{M}_j(\mathrm{HA_2})_3]$ when possible. The differential equation for the j^{th} differential equation is shown in Equation (6.36).

$$\frac{\partial[\mathrm{M}_j]}{\partial t} = k_{+j}[\mathrm{M}_j(\mathrm{HA_2})_3][\mathrm{H^+}]^3 - k_{-j}[\mathrm{M}_j][\mathrm{H_2A_2}]^3 \tag{6.36}$$

The set of j differential equations can then be solved numerically to estimate concentrations in both the aqueous phase and organic phase given a set of initial conditions. Using Runge-Kutta can be computationally expensive given a small step size, motivating the use of a GP to approximate the output of the simulator.

However, for many reasons, the model just outlined is likely wrong. The reaction order shown in (6.36), is actually unknown and is typically estimated through experimentation (Espenson 1995). The reaction order cannot be proven, only disproven (Espenson 1995). Additionally, the model above is a notable misspecification, as it assumes the simultaneous collision of three molecules, an event which has a probability approaching zero (Arnaut and Burrows 2006). To account for single molecule collisions, the reaction in (6.32) has to be broken down into a set of elementary chemical reactions. An example of a model utilizing only elementary reactions is included in Appendix C.2.1. The model using elementary reactions requires a larger number of kinetic constants, six per M_j. Additionally, a set of elementary reactions is not always equivalent to the overall reaction when modeling kinetics (Temkin, Zeigarnik, and Bonchev 1996). Further adding to complexity, the set of elementary reactions are unknown and necessary to infer (Arnaut and Burrows 2006; Temkin, Zeigarnik, and Bonchev 1996).

For the solvent extraction modeling problem, the set kinetic constants are the simulator tuning parameter u. Because the set of elementary reactions is unknown the dimension of $u^\star$ is unknown. Additionally, there may be elements unaccounted for in the model. Either laboratory analysis would not be available for these elements, or scientists might not think these elements are relevant. Therefore, interpreting u for this setting is not sensible, as u would simply be an estimate from the data for a wrong model. Additionally, modeling the bias between the computer model and real observations are useful to achieve accurate predictions from a computer simulator where there will be some discrepancy.

Utilization of this methodology would be as follows. Samples and pH are taken at each location of interest in a SX plant simultaneously. Time between each sample set is recorded. Besides the first data set, each data set would have the initial conditions of the previous data set and the elapsed time. Given this data set and an initial design for the computer simulator, the KOH-IMSPE criteria would be used to sequentially gather simulation data for improved prediction. All the data would then be assimilated and could be used to provide out of sample predictions, with the benefit of a reduced computation time relative to the simulator due to the predictions being obtained from GP surrogates.

6.6 Discussion and Conclusion

KOH-IMSPE was derived to aid in the prediction of a high dimensional problem where multiple elemental equilibria need to be predicted from a chemical process for which only observational data is available, and where a simulator is computationally expensive. Notably, with this application the goal is not accurate inference on u, but instead improved prediction. The KOH-IMSPE methodology has been shown to be able to provide improved predictions for field data with less computer simulator data compared to space filling alternatives. The methods also generalize to higher dimension. Interestingly, for both toy examples explored, not only was the mean RMSE better than the space filling alternatives, but the variance in the MC experiment was less.

For accurate inference on u another criteria, and possibly other calibration structure, would be appropriate. For example, one may undertake a simulation experiment toward other goals such as optimization, sensitivity analysis, or reliability and there are design/active learning criteria targeting those (Jones, Schonlau, and Welch 1998; Ranjan, Bingham, and Michailidis 2008; Cole et al. 2021).

Further work must be completed for adapting KOH-IMSPE for the motivating problem. First, all toy problems examined in this publication were functions with one output, while the set of differential equations used to model p elements would have greater than or equal to p outputs. Each element, as part of a simultaneous observation, could be modeled using the typical KOH framework using GPs which were independent between elements. KOH estimation of the tuning parameter u would then be possible using the joint likelihood of all elements, where each set of field data utilizes the same estimate of $\hat{u}$. Then, KOH-IMSPE would need to be adapted for vector valued functions. The original derivation for KOH-IMSPE was for vector valued functions. However, because that implementation is untested, details are not included. For a vector valued function, a scalar value of IMSPE can be produced by either using the trace or determinant of the IMSPE matrix. Implementation of this method and testing on a toy problem would be necessary before use on a real data set.

These methods still need to be tested with real data. Real data in part was not used because some of the statistical and computational details still need to be worked out before use. However, a full real data set was not yet available in part due to delays in testing a full scale processing plant. This derivation and testing of KOH-IMSPE on a data generating mechanism with a scalar output is a small step towards the efficient assimilation of real and simulated data, in order to reduce data requirements for modeling solvent extraction processes.

However, the contribution in this chapter is notable to the statistical literature. Calibration of simulators is useful to a wide range of scientific applications. The efficient collection of data near the estimate of the tuning parameter $\hat{u}$ has the potential to significantly reduce computational requirements for calibration problems.

Chapter 7

Conclusion

7.1 Summary

In this dissertation, Bayesian methods were presented which address specific process engineering problems, but can be applied outside of specific process cases. The Bayesian data reconciliation model in §3 provides more accurate results through the use of sensible prior distributions, full uncertainty quantification, and model selection. Particularly, the utilization of a normal prior distribution truncated at zero provides advancements in accuracy for the low and variable REE concentrations found in AMD.

§4 provides a quantitative standard for estimating if process streams are approaching steady state conditions, and provides estimates for steady state flows under the condition that the process is at steady state. Providing this standard allows for systematic comparisons between processes using all available data. Having a consistent method for determining steady state provides advantages over inconsistent opinion based methods by removing some of the human factors.

Bayesian models using a Gaussian process prior can be applied in conjunction with active learning to minimize data requirements for the prediction, optimization, and the calibration of simulators for highly complex systems. §5 illustrated the potential for optimization criteria to be used for finding and testing the optimum of an unknown process. Often less data was used in for the simulation experiment in §5 than in mineral processing related publications which use response surface methodology for prediction.

When real data is expensive or can only be collected observationally, but a process simulation is available, Bayesian Gaussian process models can be used to assimilate real and simulated data, model the bias between a simulation and real observations, and reduce the computation time required to evaluate a simulator. Utilization of simulator data is particularly useful for processes where direct control of experimental factors is difficult or impossible, but obtaining obseravtional data for a process is easy. For example in SX processes design of experiments which dictate initial organic and aqueous phase concentrations is often infeasable, but sampling the system after a new feedstock has been introduced is easy. §6 illustrates how Bayesian calibration methods can be useful for SX reactions, and introduces novel statistical methodology for reducing simulation computation time which is useful for the high dimensionality of a SX process.

7.2 Conclusions

Evidence has been provided which shows Bayesian methods can be used to provide further uncertainty quantification and prediction accuracy, while reducing data requirements, in mineral processing applications. The purpose of this document is to illustrate such utility in separation processes. The purpose is not to provide exact conclusions and recommendations about solvent extraction systems or rare earth element concentration process. Instead, the complexity of concentrating a set of multiple elements from a low concentration feedstock has provided the motivation for furthering Bayesian methods for separation processes, and the methods derived can be used for any separation process.

Some similarities are present between the methods presented, for example the use of a Gibbs sampler to obtain samples from marginal posterior distributions in §3 and §4. However, there is not clear coherence between all of the methods and some methods are not compatible for general cases. Using the steady state inference methods in §4 would be difficult if one wanted to model a SX system over time using the methods in §6

Instead, the methods are presented for use independently, which provide engineers the creativity to piece these general methods together as required for their problem. One would be able to use the steady state inference methods in §4 to consistently select when their process achieves steady state, and then use the relevant data for steady state Bayesian data reconciliation presented in §3. Similarly, one could use the methods in §4 and §5 for a process which needs to reach steady state by optimizing the inferred value of $[y]$ from a collected data set. The methods presented can be synthesized in an uncountable number of ways, and the independent nature of their presentation allows process engineers to implement Bayesian statistical methods as appropriate.

7.3 Recommendations for Future Work

Work on Bayesian methods for separations is in no way complete. An alternate title for this dissertation could be *A Handful of Bayesian Methods for Separations*, as many areas of present day Bayesian inference and statistical methods are completely absent. Instead this presentation illustrates how some methods can be useful, which may inspire further investigation into applied Bayesian methods for separations.

Some areas for future work in this area are clear. The methods in §5 and §6 are tested on simulation studies, and using these methods to optimize a process and calibrate a simulation for improved prediction with real data is an easy next step. The Monte-Carlo experiments in §5 and §6 would not be repeatable with real data, but the use of these methods on real data still needs to be demonstrated.

There is certainly room for the Bayesian methods presented to be further modified for improvement. However, more complicated models are not always better. More complex covariance structures for Bayesian data reconciliation could be inferred using Gaussian graphical models (Carvalho and Scott 2009). Such an implementation would limit the large number of comparisons required using Bayes factors in order to entertain covariance structures outside of what was introduced in §3.

Bayesian vector autoregression models (Litterman 1986) are vector valued versions of the scalar autoregressive functions overviewed in §4. An investigation into using vector autoregression models could potentially allow for similar utility of the methods in §4, but utilize the information from all sampling locations simultaneously.

Besides real world application, there are many more Bayesian methods which can contribute improvements in separations modeling. Active learning can be used to sequentially gather real data for predictive purposes, as part of a design of experiments. A significant effort was made to include this application in this dissertation using solvent extraction shake tests. Difficulty was found in stipulating the boundaries parameters used to run experiments which would produce usable results. Often after mixing an aqueous and organic phase a stable emulsion would form, sometimes settling into three phases over long periods of time. For these tests collecting an uncontimanated aqueous solution was difficult or impossible. Obtaining assay data on an experiment which produced three phases would be not be useful as there typically was no distinct boundary between aqueous and organic phases. Additionally equilibrium pH would often be far outside of the desired range, making the data not useful for process modeling.

The difficulties in collecting SX data for active learning lead to the more simple idea of obtaining observational data for calibrating a simulator in §6. Using active learning to gather real SX shake test data may require a more refined approach. Active learning could be used in an initial study to find the contour (Cole et al. 2021) separating tests which are unlikely to form a stable emulsion and have a pH in the correct range from those which do not satisfy those conditions in order to refine the input space. Additionally, a low dimensional example with fairly low concentrations would provide insight into using these methods for SX shake tests, and the use non-stationary GP models could better handle chemical reactions with abrupt changes.

Equilibrium modeling was explored in §6. Computational issues aside, derivations for the methods in §6 can be found in closed form for a vector valued co-kriging GP (Ver Hoef and Barry 1998). Strictly using the methods in §6 would require for multiple elements in equilibrium to be modeled independently, requiring a simulator to be recalibrated for each element in each phase. Utilizing a co-kriging GP to model the concentrations of all elements simultaneously would likely improve the calibration step and possibly reduce the number of simulator runs required.

Last, but not least, outside of a small part of §3 little consideration is given in this text to environmental or economic feasibility, with a primary focus on analysis of the technical capabilities of a process. Bayesian methods which provide samples from a posterior predictive distribution can be directly used with a Monte-Carlo financial analysis. Life cycle assessment (Klöpffer 1997; Finnveden et al. 2009) is a critical part of estimating environmental impacts and CO_2 emissions of separation processes. Research merging samples from a predictive distribution of process technical performance with the Monte-Carlo capabilities common with life cycle assessment software could inform a sensitivity analysis of environmental impact with technical performance results from real data.

Obviously work on adapting Bayesian methods to separation engineering is far from complete. Once again, the purpose of this dissertation is not to provide the details for every possible method. Instead the goal is to give others a starting point, via references and some inspiration, as to how they can better quantify uncertainty for decision making in their own process engineering problems.

Appendix A

Bayesian Data Reconsciliation

This section was accepted as part of a publication in the international peer reviewed journal Minerals Engineering. Citation details are provided below:

Koermer S, Noble A (2021). “The utility of Bayesian data reconciliationfor separations.” *Minerals Engineering*, *169*, 106837.

A.1 Conditional Posterior Derivations

A.1.1 Joint Likelihood

The likelihood, conditioned on the model parameters, of independent and identically distributed random observations is equal to a product of the probability of each observation.

$$\begin{aligned}\mathcal{L}(\boldsymbol{\Omega},\beta|y) &= \prod_{k=1}^{K}(2\pi)^{-\frac{N\times M}{2}}|\boldsymbol{\Omega}|^{-\frac{1}{2}}e^{-\frac{1}{2}(\boldsymbol{y}_k-\boldsymbol{X\beta})^T\boldsymbol{\Omega}^{-1}(\boldsymbol{y}_k-\boldsymbol{X\beta})} \\ &\propto \prod_{k=1}^{K}|\boldsymbol{\Omega}|^{-\frac{1}{2}}e^{-\frac{1}{2}(\boldsymbol{y}_k-\boldsymbol{X\beta})^T\boldsymbol{\Omega}^{-1}(\boldsymbol{y}_k-\boldsymbol{X\beta})} \\ &\propto |\boldsymbol{\Omega}|^{-\frac{K}{2}}e^{-\frac{1}{2}\sum_{k=1}^{K}(\boldsymbol{y}_k-\boldsymbol{X\beta})^T\boldsymbol{\Omega}^{-1}(\boldsymbol{y}_k-\boldsymbol{X\beta})}\end{aligned} \tag{A.1}$$

A.1.2 Inference on $\boldsymbol{\beta}$

The conditional posterior distribution $p(\boldsymbol{\beta}|\boldsymbol{X},\boldsymbol{y},\boldsymbol{\Omega},\boldsymbol{\mu}_0,\boldsymbol{V}_0)$, is derived by multiplying the truncated multi-variate normal conjugate prior distribution $p(\boldsymbol{\beta})$ with the conditional likelihood $\mathcal{L}(\boldsymbol{\beta}|\boldsymbol{\Omega},\boldsymbol{y})$, derived from (A.1) and dropping the conditioned terms when possible. Let $\boldsymbol{X}_K$ be the row-wise repetition of $\boldsymbol{X}$ K

times, and $\boldsymbol{\Omega}_K$ be the Kronecker product $\boldsymbol{I}_K \otimes \boldsymbol{\Omega}$. $\boldsymbol{\mu}_0$ and $\boldsymbol{V}_0$ are parameters of $p(\boldsymbol{\beta})$, specifying the prior mean and variance of $\boldsymbol{\beta}$. $I[\boldsymbol{\beta} > 0]$ is an indicator function equal to 1 when $\boldsymbol{\beta} > 0$ and 0 otherwise.

$$\begin{aligned}
&p(\beta|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\Omega}) \propto \mathcal{L}(\beta|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\Omega})p(\beta)\\
&\propto p(\boldsymbol{\beta})|\boldsymbol{\Omega}|^{-\frac{K}{2}} e^{-\frac{1}{2}\sum_{k=1}^{K}(\boldsymbol{y}_k - \boldsymbol{X}\boldsymbol{\beta})^T\boldsymbol{\Omega}^{-1}(\boldsymbol{y}_k - \boldsymbol{X}\boldsymbol{\beta})}\\
&\propto e^{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{X}_K\boldsymbol{\beta})^T\boldsymbol{\Omega}_K^{-1}(\boldsymbol{y}-\boldsymbol{X}_K\boldsymbol{\beta})} e^{-\frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\mu}_0)^T\boldsymbol{V}_0^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}_0)} I[\boldsymbol{\beta} > 0]\\
&\propto e^{-\frac{1}{2}\big((\boldsymbol{\beta}-(K\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}K\bar{\boldsymbol{y}}_{i,j})^T(K\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}\boldsymbol{X})(\boldsymbol{\beta}-(K\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}K\bar{\boldsymbol{y}}_{i,j})+}\\
&\qquad e^{\qquad (\boldsymbol{\beta}-\boldsymbol{\mu}_0)^T\boldsymbol{V}_0^{-1}(\boldsymbol{\beta}-\boldsymbol{\mu}_0)\big)} I[\boldsymbol{\beta} > 0]\\
&\propto e^{-\frac{1}{2}\big(\boldsymbol{\beta}^T(K\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}\boldsymbol{X}+\boldsymbol{V}_0^{-1})\boldsymbol{\beta}-2(\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}K\bar{\boldsymbol{y}}_{i,j}+\boldsymbol{V}_0^{-1}\boldsymbol{\mu}_0)\boldsymbol{\beta}+}\\
&\qquad e^{\qquad K\bar{\boldsymbol{y}}_i^T\boldsymbol{X}\boldsymbol{\Omega}^{-1}(K\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}K\bar{\boldsymbol{y}}_{i,j}\big)} I[\boldsymbol{\beta} > 0]\\
&\equiv \mathcal{N}_0^{\infty}\left((\boldsymbol{V}_0^{-1} + K\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}\boldsymbol{X})^{-1}(\boldsymbol{V}_0^{-1}\boldsymbol{\mu}_0 + K\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}\bar{\boldsymbol{y}}_i), (\boldsymbol{V}_0^{-1} + K\boldsymbol{X}^T\boldsymbol{\Omega}^{-1}\boldsymbol{X})^{-1}\right)
\end{aligned} \tag{A.2}$$

A.1.3 Inference on $\sigma_{i,j}^2$

The conditional posterior distribution $p(\sigma_{i,j}^2|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\beta})$, is derived using the inverse Gamma conjugate prior distribution $p(\sigma_{i,j}^2)$, with parameters α_0 and β_0. Note, elements of $\boldsymbol{y}$ and $\boldsymbol{X}$ are dropped when possible.

$$\begin{aligned}
&p(\sigma_{i,j}^2|\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{y}) \propto \mathcal{L}(\sigma_{i,j}^2|\boldsymbol{\beta}, \boldsymbol{X}, \boldsymbol{y})p(\sigma_{i,j}^2)\\
&\propto |\boldsymbol{\Omega}|^{-\frac{K}{2}} e^{-\frac{1}{2}\sum_{k=1}^{K}(\boldsymbol{y}_k - \boldsymbol{X}\boldsymbol{\beta})^T\boldsymbol{\Omega}^{-1}(\boldsymbol{y}_k - \boldsymbol{X}\boldsymbol{\beta})} (\sigma_{i,j}^2)^{-\alpha_0-1} e^{-\sigma_{i,j}^2\beta_0}\\
&\propto (\sigma_{i,j}^2)^{-\frac{K}{2}} e^{\frac{-\frac{1}{2}\sum_{k=1}^{K}(y_{i,j,k}-x_{i,j}^T\boldsymbol{\beta})^2}{\sigma_{i,j}^2}} (\sigma_{i,j}^2)^{-\alpha_0-1} e^{-\frac{\beta_0}{\sigma_{i,j}^2}}\\
&\propto (\sigma_{i,j}^2)^{-(\frac{K}{2}+\alpha_0)-1} e^{\frac{-\left(\frac{1}{2}\sum_{k=1}^{K}(y_{i,j,k}-x_{i,j}^T\boldsymbol{\beta})^2+\beta_0\right)}{\sigma_{i,j}^2}}\\
&\equiv \text{invGamma}\left(\frac{K}{2} + \alpha_0, \frac{1}{2}\sum_{k=1}^{K}(y_{i,j,k} - x_{i,j}^T\boldsymbol{\beta})^2 + \beta_0\right)
\end{aligned} \tag{A.3}$$

A.1.4 Inference on $\boldsymbol{\Sigma}_i$

The conditional posterior distribution $p(\boldsymbol{\Sigma}_i|\boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\beta})$, derived using the inverse Wishart conjugate prior distribution $p(\boldsymbol{\Sigma}_i)$, with parameters ν_0 and $\boldsymbol{S}_0$. The trace identities $tr(\boldsymbol{A} + \boldsymbol{B}) = tr(\boldsymbol{A}) + tr(\boldsymbol{B})$ and $tr(\boldsymbol{ABC}) = tr(\boldsymbol{CBA})$, as well as a determinant property of block diagonal matrices are required for this derivation.

$$
\begin{aligned}
&p(\boldsymbol{\Sigma}_i|\boldsymbol{\beta},\boldsymbol{X},\boldsymbol{y}) \propto \mathcal{L}(\boldsymbol{\Sigma}_i|\boldsymbol{\beta},\boldsymbol{X},\boldsymbol{y})p(\boldsymbol{\Sigma}_i)\\
&\propto |\boldsymbol{\Omega}|^{-\frac{K}{2}} e^{-\frac{1}{2}\sum_{k=1}^{K}(\boldsymbol{y}_k-\boldsymbol{X}\boldsymbol{\beta})^T\boldsymbol{\Omega}^{-1}(\boldsymbol{y}_k-\boldsymbol{X}\boldsymbol{\beta})}\sigma_{i,j}^{\alpha_0-1}e^{-\sigma_{i,j}\beta_0}|\boldsymbol{\Sigma}_i|^{-\frac{\nu_0+N+1}{2}}e^{-\frac{1}{2}tr(\boldsymbol{S_0}\boldsymbol{\Sigma}_i)}\\
&\propto |\boldsymbol{\Sigma}_i|^{-\frac{K}{2}} e^{-\frac{1}{2}\sum_{k=1}^{K}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})}|\boldsymbol{\Sigma}_i|^{-\frac{\nu_0+N+1}{2}}e^{-\frac{1}{2}tr(\boldsymbol{S_0}\boldsymbol{\Sigma}_i)}\\
&\propto |\boldsymbol{\Sigma}_i|^{-\frac{K+\nu_0+N+1}{2}} e^{-\frac{1}{2}\left(\sum_{k=1}^{K}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})tr(\boldsymbol{S_0}\boldsymbol{\Sigma}_i)\right)}\\
&\propto |\boldsymbol{\Sigma}_i|^{-\frac{K+\nu_0+N+1}{2}} e^{-\frac{1}{2}\left(tr\left(\sum_{k=1}^{K}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})\right)tr(\boldsymbol{S_0}\boldsymbol{\Sigma}_i)\right)}\\
&\propto |\boldsymbol{\Sigma}_i|^{-\frac{K+\nu_0+N+1}{2}} e^{-\frac{1}{2}\left(tr\left(\sum_{k=1}^{K}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})^T\boldsymbol{\Sigma}_i^{-1}\right)tr(\boldsymbol{S_0}\boldsymbol{\Sigma}_i)\right)}\\
&\propto |\boldsymbol{\Sigma}_i|^{-\frac{K+\nu_0+N+1}{2}} e^{-\frac{1}{2}\left(tr\left(\sum_{k=1}^{K}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})^T\boldsymbol{\Sigma}_i^{-1}+\boldsymbol{S_0}\boldsymbol{\Sigma}_i\right)\right)}\\
&\propto |\boldsymbol{\Sigma}_i|^{-\frac{K+\nu_0+N+1}{2}} e^{-\frac{1}{2}\left(tr\left(\left(\boldsymbol{S_0}+\sum_{k=1}^{K}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\boldsymbol{\beta})^T\right)\boldsymbol{\Sigma}_i^{-1}\right)\right)}\\
&\equiv \mathcal{W}^{-1}\left(K+\nu_0,\boldsymbol{S_0}+\sum_{k=1}^{K}(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\beta)(\boldsymbol{y}_{i,k}-\boldsymbol{X}_i\beta)^T\right)
\end{aligned}
\tag{A.4}
$$

A.2 Point Estimate Mass Balance

Derivation for point mass balance model adapted from Wills (2006) and used for the comparisons shown in §3.3.1.1, 3.3.1.2, Figure 3.4, Figure 3.6, and implemented in the `pointmassbal()` function.

Let $\boldsymbol{a}$ be a vector of assays, with elements $a_{i\ k}$ being the fractional assay for the k^{th} component of the mass sampled at location i, where $i \in \{f, c, t\}$ around a node. The goal is to find $\hat{\boldsymbol{a}}$, as illustrated in (A.5), where Ψ is a matrix with $\Psi_{i,i} = \ \mathrm{ar}(\boldsymbol{a}_{k,i})$, λ is a Lagrange multiplier, and C is a matrix of constraints specifying conservation of mass.

$$S = \sum_{k=1}^{K}(\boldsymbol{a}_k - \hat{\boldsymbol{a}})^T\Psi^{-1}(\boldsymbol{a}_k - \hat{\boldsymbol{a}}) + \lambda^T C\hat{\boldsymbol{a}} \tag{A.5}$$

$\hat{\boldsymbol{a}}$ must be chosen to minimize S. This is equivalent to minimizing the squared difference between the observed and estimated values of $\boldsymbol{a}$, weighted by the variance of each $\boldsymbol{a}$, with the constraint. The fractional assay values around a node are constrained by combining the two product formula and the conservation of mass equation (A.6).

$$\begin{aligned}
C &= \frac{(f-t)}{c-t} \qquad (A.6)\\
F &= C + T\\
&\Longrightarrow\\
1 &= C - (1-C)\\
f &= cC - t(1-C)\\
0 &= f - cC - t(1-C) \qquad (A.7)
\end{aligned}$$

Substitution gives the constraint around a node (A.7), which can be written in matrix form

$$\boldsymbol{C}\boldsymbol{a} = \begin{bmatrix} 1 & -C & -(1-C) \end{bmatrix} \begin{bmatrix} f \\ c \\ t \end{bmatrix} \qquad (A.8)$$

Solving for $\hat{\boldsymbol{a}}$ requires first differentiating with respect to $\hat{\boldsymbol{a}}$, setting this equal to 0, and solving and rearranging terms so that $\hat{\boldsymbol{a}}$ is on the left hand side.

$$\begin{aligned}
\frac{\partial S}{\partial \hat{\boldsymbol{a}}} &= \sum_{k=1}^{K} \left(-2\boldsymbol{a}_k^T \Psi^{-1} + 2\hat{\boldsymbol{a}}^T \Psi^{-1}\right) + \lambda^T \boldsymbol{C}\\
&= -2K\bar{\boldsymbol{a}}^T \Psi^{-1} + 2K\hat{\boldsymbol{a}}^T \Psi^{-1} + \lambda^T \boldsymbol{C}\\
\frac{\partial S}{\partial \hat{\boldsymbol{a}}} &\overset{\text{set}}{=} 0 \implies\\
\hat{\boldsymbol{a}} &= \bar{\boldsymbol{a}} - \frac{1}{2K}\Psi \boldsymbol{C}^T \lambda
\end{aligned} \qquad (A.9)$$

Then, finding the derivative with respect to lambda, and setting it equal to zero gives $0 = \boldsymbol{C}\hat{\boldsymbol{a}}$, which is our constraint. We can then multiply (A.9) by $\boldsymbol{C}$ on the left hand side, and set it equal to zero to solve for lambda.

$$\begin{aligned}
\boldsymbol{C}\hat{\boldsymbol{a}} &= \boldsymbol{C}\bar{\boldsymbol{a}} - \frac{1}{2K}\boldsymbol{C}\Psi\boldsymbol{C}^T\lambda \implies\\
0 &= \boldsymbol{C}\bar{\boldsymbol{a}} - \frac{1}{2K}\boldsymbol{C}\Psi\boldsymbol{C}^T\lambda\\
&\Longrightarrow\\
\lambda &= 2K(\boldsymbol{C}\Psi\boldsymbol{C}^T)^{-1}\boldsymbol{C}\bar{\boldsymbol{a}}
\end{aligned} \qquad (A.10)$$

The solution for λ in (A.10) can then be plugged into (A.9) to solve for $\hat{\boldsymbol{a}}$.

$$\hat{\boldsymbol{a}} = \bar{\boldsymbol{a}} - \Psi\boldsymbol{C}^T(\boldsymbol{C}\Psi\boldsymbol{C}^T)^{-1}\boldsymbol{C}\bar{\boldsymbol{a}} \qquad (A.11)$$

This closed form solution for $\hat{\boldsymbol{a}}$ is still dependent on the elements in Ψ and the estimate for C. Let $\hat{C}$ be the estimate for C, and an element of the matrix $\boldsymbol{C}$. Let Equation (A.8) be rewritten as Equation (A.12):

$$\boldsymbol{C}\boldsymbol{a} = \begin{bmatrix} 1 & -\hat{C} & -(1-\hat{C}) \end{bmatrix} \begin{bmatrix} f \\ c \\ t \end{bmatrix} \tag{A.12}$$

The diagonal of Ψ is simply calculated to be the sample variance of each element of $\boldsymbol{a}$. The solution $\hat{C}$ is given in Wills (2006) and found using two equations.

$$V_{rk} = V_{fk} + \hat{C}^2 V_{ck} + (1-\hat{C})^2 V_{tk} \tag{A.13}$$

$$\hat{C} = \frac{\sum_{k=1}^{n} \frac{(f_k - t_k)(c_k - t_k)}{V_{rk}}}{\sum_{k=1}^{n} \frac{(c_k - t_k)^2}{V_{rk}}} \tag{A.14}$$

Where k indexes a component in a sample, and V_{jk} is the variance in the assay of component k at sampling location j. To solve for $\hat{C}$ an initial value is chosen and then iteratively plugged into Equations (A.13) and (A.14) until $\hat{C}$ converges. Then, plugging this value into (A.11) completes the mass balance giving the best point estimate for the true assay.

For the two node application, it is important to note that $\hat{C}$ is a vector of length two. Equations (A.12), (A.13), and (A.14) are generalizable to a process with more than one node. Elements of $\hat{C}$ are calculated with Equation (A.14) using only the values directly related to the mass flows in and out of a the relevant node. For the two node process in this text, the matrix $\boldsymbol{C}$ can be written as is shown in Equation (A.15).

$$\boldsymbol{C} = \begin{bmatrix} 1 & -\hat{C}_1 & 0 & -(1-\hat{C}_1) & 0 \\ 0 & 1 & -\hat{C}_2 & 0 & -(1-\hat{C}_2) \end{bmatrix} \tag{A.15}$$

A.3 Process Simulation

Table A.1: Simulated Data

Sample Location	$CuFeS_2$ TPH					Gangue TPH				
	y_1	y_2	y_3	y_4	y_5	y_1	y_2	y_3	y_4	y_5
Test 1	1.353	1.417	1.200	0.020	0.047	95.753	6.299	0.356	91.466	7.790
Test 2	1.172	1.248	1.219	0.018	0.037	95.466	8.348	0.217	90.292	8.265
Test 3	1.364	1.111	1.103	0.038	0.075	100.457	6.846	0.273	87.855	6.877
Test 4	1.371	1.014	1.103	0.033	0.059	106.187	7.976	0.360	92.418	7.391
Test 5	1.252	1.331	1.160	0.017	0.047	100.545	8.683	0.364	98.761	6.872
Test 6	1.061	1.143	1.001	0.032	0.058	96.294	5.283	0.335	86.146	6.464
Test 7	1.180	1.211	1.181	0.024	0.042	109.416	7.904	0.241	93.446	7.472

A.4 MCMC Diagnostics

Table A.2: MCMC diagnostic summary for simulated data.

Model	Parameter	min(ESS)	max(ESS)	min(|CD|)	max(|CD|)
Full Cov.	$\boldsymbol{\beta}$	96934.15	100000.0	0.0459165	2.464641
	$\boldsymbol{\Sigma}_1$	97287.22	103756.5	0.0591586	2.216777
	$\boldsymbol{\Sigma}_2$	96620.76	100000.0	0.0336163	1.317138
Indep. Cov.	$\boldsymbol{\beta}$	96614.98	100000.0	0.4513866	1.659906
	$\sigma^2_{1,j}$	93805.70	100000.0	0.3469462	1.868041
	$\sigma^2_{2,j}$	94502.35	100516.1	0.3011865	1.933159

Table A.3: MCMC diagnostic summary for real data.

Model	Parameter	min(ESS)	max(ESS)	min(|CD|)	max(|CD|)
Full Cov.	$\boldsymbol{\beta}$	92188.61	94256.58	0.0533900	2.327887
	$\boldsymbol{\Sigma}_1$	93421.65	100622.99	0.0033434	2.473025
Indep. Cov.	$\boldsymbol{\beta}$	98459.75	99000.00	0.2361097	1.938821
	$\sigma^2_{1,j}$	97660.06	102566.16	0.1769926	1.804199

A.5 `BayesMassBal` Example

After loading the package:

```
library(BayesMassBal)
```

First data is generated:

```
y <- twonodeSim()$simulation
```

Then a matrix of linear constraints for the two node process is specified and the `constrainProcess` function is used to find the `X` required for the `BMB` function.

```
C <- matrix(c(1,-1,0,-1,0,0,1,-1,0,-1), byrow = TRUE, ncol = 5,
            nrow = 2)
X <- constrainProcess(C = C)
```

Next, the `BMB` function is run twice, once for each model structure. The `cov.structure` argument specifies the error structure. Setting `lml = TRUE` also calculates $\log(p(y|M_l))$ to aid in model selection later. The vector passed to argument `BTE` specifies the number of samples that will be collected.

```
BTE <- c(10000,100000,1)
indep.model <- BMB(X = X, y = y, cov.structure = "indep",
                   BTE = BTE, lml = TRUE)
```

```
component.model <- BMB(X = X, y = y, cov.structure = "component",
                       BTE = BTE, lml = TRUE)
```

The resulting variables `indep.model` and `component.model` are objects of class `"BayesMassBal"` which included samples from the marginal posterior distributions, information on the priors used, the log marginal likelihood. After running the `BMB` function using both model structures, the model that best fits the data can be chosen using a Bayes Factor.

```
component.model$lml - indep.model$lml
```

```
## [1] 127.67
```

There is much stronger support for the model that includes error correlation between sampling locations for a particular component. It is possible to print a summary of the model where `cov.structure = "component"` model has been selected to the `R` console, and save a `*.csv` file of this summary to the working directory for use in other programs, using the `summary()` function.

```
summary(component.model, export = "mass_balance_summary.csv")
```

A "BayesMassBal" object can be supplied to the `mainEff` function to produce an output that can be used to help make a plot like Figure 3.5. `mainEff` also requires a function and the range of independent uniformly distributed random variables. The range of each x is taken from Table 3.1, and `fn` supplied to `mainEff` is to calculate NSR.

```
netRev <- function(X,ybal){
      cu.frac <- 63.546/183.5
      feed.mass <- ybal$CuFeS2[1] + ybal$gangue[1]
      # Concentrate mass per ton feed
      con.mass <- (ybal$CuFeS2[3] + ybal$gangue[3])/feed.mass
      # Copper mass per ton feed
      cu.mass <- (ybal$CuFeS2[3]*cu.frac)/feed.mass
      gam <- c(-1,-1/feed.mass,cu.mass,-con.mass,-cu.mass,
               -con.mass)
      f <- X %*% gam
      return(f)
}

rangex <- matrix(c(4.00,6.25,1125,1875,3880,9080,20,60,96,208,
                   20.0,62.5), ncol = 6, nrow = 2)
location.model[["MainEffects"]] <- mainEff(location.model,
                                           fn = "netRev",
                                           rangex =  rangex,
                                           xj = 3,
                                           N = 100, res = 100)
```

See `vignette("Two_Node_Process", package = "BayesMassBal")` for code taking the output of `mainEff()` and generating Figure 3.5

Appendix B

Steady State Estimation

B.1 R Code

```
library(BayesMassBal)

ssest <- function (y, BTE = c(500, 20000, 1), stationary = FALSE)
  {
  require(LaplacesDemon)
  require(tmvtnorm)
    burn <- BTE[1]
    total <- BTE[2]
    every <- BTE[3]
    collected <- ceiling((total - burn)/every)
    y <- drop(y)
    Y <- y[-1]
    X <- matrix(1, nrow = length(y) - 1, ncol = 2)
    X[, 2] <- y[-length(y)]
    sig <- rep(NA, times = collected)
    beta <- matrix(NA, nrow = 2, ncol = collected)
    sigsamp <- var(y)
    if (stationary == TRUE) {
        B0 <- c(mean(y), 0)
        V0i <- diag(c(1/(sigsamp * 100), 1/(1000)))
        V0iB0 <- V0i %*% B0
        XTX <- t(X) %*% X
        V <- solve((1/sigsamp)*XTX  + V0i)
        bhat <- as.vector(V %*% (V0iB0 + (1/sigsamp) * t(X) %*%
            Y))
        lb <- c(-Inf, -1)
        ub <- c(Inf, 1)
    }
    else if (stationary == FALSE) {
        bhat <- as.vector(solve(t(X) %*% X) %*% t(X) %*% Y)
        XTXi <- solve(t(X) %*% X)
        V <- XTXi * sigsamp
        lb <- c(-Inf, -Inf)
        ub <- c(Inf, Inf)
    }
    bsamp <- bhat
    a <- length(Y)/2
    for (i in 1:total) {
        bsamp <- as.vector(rtmvnorm(1, mean = bhat, sigma = V,
            lower = lb, upper = ub))
        ymXB <- Y - X %*% bsamp
        b <- 0.5 * t(ymXB) %*% ymXB
        sigsamp <- rinvgamma(1, shape = a, scale = b)
        if (i > burn & ((i - burn)/every)%%1 == 0) {
            save.sel <- (i - burn)/every
            beta[, save.sel] <- bsamp
            sig[save.sel] <- sigsamp
        }
        if (stationary == TRUE) {
            V <- solve(XTX * (1/sigsamp) + V0i)
            bhat <- as.vector(V %*% (V0iB0 + (1/sigsamp) *
                                       t(X) %*% Y))
        }
        else {
            V <- XTXi * sigsamp
        }
    }
    if (stationary == TRUE) {
        expectation <- beta[1, ]/(1 - beta[2, ])
        samples <- list(mu = beta[1, ], alpha = beta[2, ],
                        expectation = expectation, s2 = sig)
    }
    else if (sum(beta[2, ] <= -1) == 0 & sum(beta[2, ] >= 1) ==
        0) {
        expectation <- beta[1, ]/(1 - beta[2, ])
        samples <- list(mu = beta[1, ], alpha = beta[2, ],
                        expectation = expectation, s2 = sig)
    }
    else {
        samples <- list(mu = beta[1, ], alpha = beta[2, ],
                        s2 = sig)
    }
    out <- list(samples = samples, stationary = stationary, y = y,
        type = "time-series")
    class(out) <- "BayesMassBal"
    return(out)
}
```

Appendix C

Active Learning for Bayesian Calibration of Solvent Extraction Simulators

C.1 Kennedy and O'Hagan IMSPE Derivations

C.1.1 Ingegrals

Details are provided for evaluating the integrals required to calculate KOH-IMSPE in closed form when the GPs used in modeling the computer simulation and bias function both utilize a Gaussian coraviance kernel.

For matricies $W_{M,M} = \int_0^1 k_M k_M^T dx$, $W_{M,B} = \int_0^1 k_M k_B^T dx$, and $W_{B,B} = \int_0^1 k_B k_B^T dx$, where k_M and k_B utilize the Gaussian covariance function and data locations occupy the rectangular space $[0,1]^s$, the i,j element of $W_{\alpha,\beta}$ can generally be found as:

$$w_{\alpha,\beta}(x_i, x_j) = \prod_{l=1}^{s} \int_0^1 k_\alpha(x_{i,l}, x_l) k_\beta(x_l x_{j,l}) dx_l \tag{C.1}$$

C.1.1.1 $W_{M,M}$

$$W_{M,M} = \begin{bmatrix} \int k^M_{(f,x_f)} k^M_{(x_f,f)} dx_f & \int k^M_{(f,x_f)} k^M_{(x_f,m,)} dx_f \\ \int k^M_{(m,x_f)} k^M_{(x_f,f)} dx_f & \int k^M_{(m.x_f)} k^M_{(x_f,m)} dx_f \end{bmatrix} \tag{C.2}$$

When integrating the matricies in (C.2) over the dimensions of Z, integrals used in evaluating the product (C.1) are equivalent to the derivations provided in Binois et al. (2019), shown below.

$$w_{M,M}(z_{i,l}, z_{j,l}) = \frac{\sqrt{2\pi\theta_M}}{4} \exp\left(-\frac{(z_{i,l} - z_{j,l})^2}{2\theta_{M,l}}\right) \left(\operatorname{erf}\left(\frac{2 - (z_{i,l} + z_{j,l})}{\sqrt{2\theta_{M,l}}}\right) + \operatorname{erf}\left(\frac{z_{i,l} + z_{j,l}}{\sqrt{2\theta_{M,l}}}\right)\right) \tag{C.3}$$

To use gradients for the search of the $\tilde{z} \in \tilde{x}$ which minimizes KOH-IMSPE, the derivative of Equation (C.3) is:

$$\frac{\partial w_{M,M}(z_{i,l}, \tilde{z}_l)}{\partial \tilde{z}_l} = \sqrt{\frac{\pi}{2}} \exp\left(-\frac{(z_{i,l} - \tilde{z}_l)^2}{2\theta_M}\right) \left((z_{i,l} - \tilde{z}_l) \frac{\operatorname{erf}\left(\frac{2-(z_{i,l}-\tilde{z}_l)}{\sqrt{2\theta_M}}\right) + \operatorname{erf}\left(\frac{z_{i,l}+\tilde{z}_l}{\sqrt{2\theta_M}}\right)}{2\sqrt{\theta_M}} + \frac{1}{2}\sqrt{\frac{2}{\pi}} \left(\exp\left(-\frac{(z_{i,l} + \tilde{z}_l)^2}{2\theta_M}\right) - \exp\left(-\frac{(2 - z_{i,l} - \tilde{z}_l)^2}{2\theta_M}\right)\right)\right)$$

Where erf() is the Gauss error function. However, for the case where $z_{i,l} = \tilde{z}_l$, instead the derivative in (C.4) should be used.

$$\frac{\partial w_{M,M}(\tilde{z}_l, \tilde{z}_l)}{\partial \tilde{z}_l} = \exp\left(-\frac{2\tilde{z}_l^2}{\theta_M}\right) - \exp\left(-\frac{2(\tilde{z}_l - 1)^2}{\theta_M}\right) \tag{C.4}$$

Because point estimates of $\hat{u}$ are taken, integrating over U space results in the following terms for evaluating (C.1):

$$\begin{aligned} \int k^M_{(f,u_f)} k^M_{(u_f,f)} du_f &= 1 \\ \int k^M_{(f,u_f)} k^M_{(u_f,m)} du_f &= 1_{N_f} k^M_{(\hat{u},m)} \\ \int k^M_{(f,u_f)} k^M_{(u_f,m)} du_f &= k^M_{(\hat{u},m)} k^M_{(\hat{u},m)} \end{aligned} \tag{C.5}$$

Where, when in U space, $f = \hat{u}$, predictive location $u_f = \hat{u}$. m is at the model data location in U space.

For gradient based minimization of KOH-IMSPE, differentiation of (C.5) with respect to $\tilde{u}$ is as shown in (C.6).

$$
\begin{aligned}
\frac{\partial}{\partial \tilde{u}_l} \int k^M_{(f,u_f)} k^M_{(u_f,f)} du_f &= 0 \\
\frac{\partial}{\partial \tilde{u}_l} \int k^M_{(f,u_f)} k^M_{(u_f,m)} du_f &= 1_{N_f} \frac{\partial}{\partial \tilde{u}_l} k^M_{(\hat{u},m)} \\
\frac{\partial}{\partial \tilde{u}_l} \int k^M_{(m,u_f)} k^M_{(u_f,m)} du_f &= \frac{\partial k^M_{(m,\hat{u})}}{\partial \tilde{u}_l} k^M_{(\hat{u},m)} + k^M_{(m,\hat{u})} \frac{\partial k^M_{(\hat{u},m)}}{\partial \tilde{u}_l}
\end{aligned}
\tag{C.6}
$$

C.1.1.2 $W_{M,B}$

$$
W_{M,B} = \begin{bmatrix} \int k^M_{(f,x_f)} k^B_{(x_f,f)} dx_f & 0_{N_f \times N_m} \\ \int k^M_{(m,x_f)} k^B_{(x_f,f)} dx_f & 0_{N_m \times N_f} \end{bmatrix}
\tag{C.7}
$$

When integrating the matricies in (C.7) over the dimensions of Z, integrals used in evaluating the product (C.1) require care due to the fact that k^M and k^B have different lengthscale values. The result is shown below, where z_j originates from k^B, and therefore only contains field data coordinates in Z space.

$$
\begin{aligned}
w_{M,B}(z_{i,l}, z_{j,l}) = \exp\left(-\frac{(z_{j,l} - z_{i,l})^2}{\theta_{B,l} + \theta_{M,l}}\right) \left(\frac{1}{2}\sqrt{\pi\left(\frac{1}{\theta_{M,l}} + \frac{1}{\theta_{B,l}}\right)^{-1}}\right) \times \\
\operatorname{erf}\left(\frac{\left(\frac{\theta_{B,l} z_{i,l} + \theta_{M,l} z_{j,l}}{\theta_{B,l} + \theta_{M,l}}\right)}{\sqrt{\left(\frac{1}{\theta_{M,l}} + \frac{1}{\theta_{B,l}}\right)^{-1}}}\right) - \operatorname{erf}\left(\frac{\left(\frac{\theta_{B,l} z_{i,l} + \theta_{M,l} z_{j,l}}{\theta_{B,l} + \theta_{M,l}}\right) - 1}{\sqrt{\left(\frac{1}{\theta_{M,l}} + \frac{1}{\theta_{B,l}}\right)^{-1}}}\right)
\end{aligned}
\tag{C.8}
$$

Differentiation of (C.8) with respect to $\tilde{z}$ for gradient based minimization of KOH-IMSPE produces:

$$
\begin{aligned}
&\frac{\partial w_{M,B}(\tilde{z}_l, x_{j,l})}{\partial \tilde{z}_l} = \\
&\frac{e^{-\frac{(\tilde{z}_l - z_{j,l})^2}{\theta_{B,l}+\theta_{M,l}}}}{\theta_{M,l}+\theta_{B,l}} \left(\sqrt{\pi \left(\frac{1}{\theta_{B,l}} + \frac{1}{\theta_{M,l}} \right)^{-1}} (z_{j,l} - \tilde{z}_l) \left(\operatorname{erf} \left(\frac{\left(\frac{\theta_{M,l} z_{j,l} + \theta_{B,l} \tilde{z}_l}{\theta_{M,l}+\theta_{B,l}} \right)}{\sqrt{\left(\frac{1}{\theta_{M,l}} + \frac{1}{\theta_{B,l}} \right)^{-1}}} \right) - \right. \right. \\
&\left. \operatorname{erf} \left(\frac{\left(\frac{\theta_{M,l} z_{j,l} + \theta_{B,l} \tilde{z}_l}{\theta_{M,l}+\theta_{B,l}} \right) - 1}{\sqrt{\left(\frac{1}{\theta_{M,l}} + \frac{1}{\theta_{B,l}} \right)^{-1}}} \right) \right) + \\
&\left. \theta_{B,l} \left(e^{-\frac{\left(\frac{1}{\theta_{M,l}} + \frac{1}{\theta_{B,l}} \right) (\theta_{M,l} z_{j,l} + \theta_{B,l} \tilde{z}_l)^2}{(\theta_{M,l}+\theta_{B,l})^2}} - e^{-\left(\frac{1}{\theta_{M,l}} + \frac{1}{\theta_{B,l}} \right) \left(\frac{\theta_{M,l} z_{j,l} + \theta_{B,l} \tilde{z}_l}{\theta_{M,l}+\theta_{B,l}} - 1 \right)^2} \right) \right)
\end{aligned}
$$

Note that because of the form of (C.7), when differentiating z_j only corresponds to field data.

Because point estimates of $\hat{u}$ are taken, integrating over U space results in the following terms for evaluating (C.1):

$$
\begin{aligned}
\int k^M_{(f,u_f)} k^B_{(u_f,f)} du_f &= 1 \\
\int k^M_{(m,u_f)} k^B_{(u_f,f)} du_f &= k^M_{(m,\hat{u})} 1^T_{N_f}
\end{aligned}
\tag{C.9}
$$

Differentiating equations (C.9) with respect to $\tilde{u}_l$ for gradient based minimization of KOH-IMSPE produces:

$$
\begin{aligned}
\frac{\partial}{\partial \tilde{u}_l} \int k^M_{(f,u_f)} k^B_{(u_f,f)} du_f &= 0 \\
\frac{\partial}{\partial \tilde{u}_l} \int k^M_{(m,u_f)} k^B_{(u_f,f)} du_f &= \frac{\partial k^M_{(m,\hat{u})}}{\partial \tilde{u}_l} 1^T_{N_f}
\end{aligned}
\tag{C.10}
$$

C.1.1.3 $W_{B,B}$

$$
W_{B,B} = \begin{bmatrix} \int k^B_{(f,x_f)} k^B_{(x_f,f)} dx_f & 0_{N_f \times N_m} \\ 0_{N_m \times N_f} & 0_{N_m \times N_m} \end{bmatrix}
\tag{C.11}
$$

Evaluation of all entries in $W_{B,B}$ is only related to X space. For the Gaussian kernel, the integral required for evaluation is equivalent to (C.3). Because only the dimensions in X are included in $k_{\cdot}^{B}$, the product (C.1) is over a lower dimensional space than $[Z, U]$.

Additionally, in the search to find the additional data point $\tilde{x}$ which minimizes KOH-IMSPE, $\frac{\partial W_{B,B}}{\partial \tilde{x}} = 0$.

C.1.2 Block Matrix Inversion

Let:

$$K_{M+1,B} = \begin{bmatrix} K_{M,B} & \nu_M k_{\tilde{x}} \\ \nu_M k_{\tilde{x}}^T & \nu_M k_{(\tilde{x},\tilde{x})} \end{bmatrix}$$

Where $K_{M,B}$ is of the form specified in Equation (6.15) and:

$$\nu_M k_{\tilde{x}} = \nu_M \begin{bmatrix} k^M_{(f,\tilde{x})} \\ k^M_{(m,\tilde{x})} \end{bmatrix}$$

$$\nu_M k_{(\tilde{x},\tilde{x})} = \nu_M k_{(\tilde{x},\tilde{x})}$$

$K^{-1}_{M+1,B}$ can be found as:

$$K^{-1}_{M+1,B} = \begin{bmatrix} K^{-1}_{M,B} + \frac{1}{b}\nu_M^2 K^{-1}_{M,B} k_{\tilde{x}} k_{\tilde{x}}^T K^{-1}_{M,B} & -\frac{1}{b}\nu_M K^{-1}_{M,B} k_{\tilde{x}} \\ -\frac{1}{b}\nu_M k_{\tilde{x}}^T K^{-1}_{M,B} & \frac{1}{b} \end{bmatrix} \tag{C.12}$$

Where $b = \nu_M k_{(\tilde{x},\tilde{x})} - \nu_M^2 k_{\tilde{x}}^T K^{-1}_{M,B} k_{\tilde{x}}$.

To differentiate block matrix (C.12) with respect to $\tilde{x}$ we need to differentiate each block individually. Using the chain rule to differentiate, the resulting derivatives are:

$$\frac{\partial b^{-1}}{\partial \tilde{x}_l} = b^{-2}\nu_M^2 \left(k_{\tilde{x}}^T K^{-1}_{M,B} \frac{\partial k_{\tilde{x}}}{\partial \tilde{x}_l} + \frac{\partial k_{\tilde{x}}^T}{\partial \tilde{x}_l} K^{-1}_{M,B} k_{\tilde{x}} \right) \tag{C.13}$$

$$\begin{aligned} \frac{\partial \left(\frac{1}{b}\nu_M^2 K^{-1}_{M,B} k_{\tilde{x}} k_{\tilde{x}}^T K^{-1}_{M,B} \right)}{\partial \tilde{x}_l} = & \frac{1}{b}\nu_M^2 K^{-1}_{M,B} k_{\tilde{x}} \frac{\partial k_{\tilde{x}}^T}{\partial \tilde{x}_l} K^{-1}_{M,B} + \frac{\partial b^{-1}}{\partial \tilde{x}_l}\nu_M^2 K^{-1}_{M,B} k_{\tilde{x}} k_{\tilde{x}}^T K^{-1}_{M,B} + \\ & \frac{1}{b}\nu_M^2 K^{-1}_{M,B} \frac{\partial k_{\tilde{x}}}{\partial \tilde{x}_l} k_{\tilde{x}}^T K^{-1}_{M,B} \end{aligned} \tag{C.14}$$

$$\frac{\partial \frac{1}{b}\nu_M K^{-1}_{M,B} k_{\tilde{x}}}{\partial \tilde{x}_l} = \frac{\partial b^{-1}}{\partial x_l}\nu_M K^{-1}_{M,B} k_{\tilde{x}} + \frac{1}{b}\nu_M K^{-1}_{M,B} \frac{\partial k_{\tilde{x}}}{\partial \tilde{x}_l} \tag{C.15}$$

$$\frac{\partial \frac{1}{b}\nu_M k_{\tilde{x}}^T K^{-1}_{M,B}}{\partial \tilde{x}_l} = \left(\frac{\partial \frac{1}{b}\nu_M K^{-1}_{M,B} k_{\tilde{x}}}{\partial \tilde{x}_l} \right)^T \tag{C.16}$$

C.2 Solvent Extraction Modeling

C.2.1 Field Data Generation

A REE or metal ion with a charge of 3+ reacts with D2EHPA and TBP as shown in Equation (C.17). Since it is difficult to measure $[\mathrm{RE(TBP)_3}]_{\mathrm{org}}$ and $[\mathrm{RE(D2EHPA)_3}]_{\mathrm{org}}$, $\mathrm{H_2A_2}$ is taken to be the sum of both in practice. Additionally, this derivation is for mixing of an aqueous phase with a fresh organic phase containing no concentrations of rare earths or gangue metals.

$$[\mathrm{RE}^{3+}]_{\mathrm{aq}} + [3(\mathrm{HA})_2]_{\mathrm{org}} \rightleftharpoons [\mathrm{RE(HA_2)_3}]_{\mathrm{org}} + 3[\mathrm{H}^+]_{\mathrm{aq}} \tag{C.17}$$

For a single element (RE), the set of elementary reactions in equation set (C.18) can be assumed for modeling the equilibrium between the two phases. We cannot know the true reaction mechanism. k_i^{RE} and k_{-i}^{RE} are the kinetic constants for the forward and backward reactions respectively of the i^{th} elementary reaction.

$$\begin{aligned}
[\mathrm{RE}]^{3+}_{\mathrm{aq}} + [\mathrm{H_2A_2}]_{\mathrm{org}} &\underset{k_{-1}^{\mathrm{RE}}}{\overset{k_{+1}^{\mathrm{RE}}}{\rightleftharpoons}} [\mathrm{RE(HA_2)}]^{2+}_{\mathrm{org}} + [\mathrm{H}]^{+}_{\mathrm{aq}} \\
[\mathrm{RE(HA_2)}]^{2+}_{\mathrm{org}} + [\mathrm{H_2A_2}]_{\mathrm{org}} &\underset{k_{-2}^{\mathrm{RE}}}{\overset{k_{+2}^{\mathrm{RE}}}{\rightleftharpoons}} [\mathrm{RE(HA_2)_2}]^{1+}_{\mathrm{org}} + [\mathrm{H}]^{+}_{\mathrm{aq}} \\
[\mathrm{RE(HA_2)_2}]^{1+}_{\mathrm{org}} + [\mathrm{H_2A_2}]_{\mathrm{org}} &\underset{k_{-3}^{\mathrm{RE}}}{\overset{k_{+3}^{\mathrm{RE}}}{\rightleftharpoons}} [\mathrm{RE(HA_2)_3}]_{\mathrm{org}} + [\mathrm{H}]^{+}_{\mathrm{aq}}
\end{aligned} \tag{C.18}$$

While the phase location and ion charge are noted in Equation Set (C.18), these properties are not explicitly stated afterwards solely to simplify notation. The law of mass action is then used to derive a system of differential equations for all reactants and products. However, constraints of the closed system can be used to reduce the number of differential equations required. Equations in (C.20) shows how the mass balance around all the RE complexes can be used to find an equation for the RE concentration at time t. Note, for compound M, $[M]_t$ is the concentration at time t and $[M]_0$ is the initial concentration of compound M at time $t = 0$.

$$[\mathrm{RE}]_t + \sum_{i=1}^{3}[\mathrm{RE(HA_2)}_i]_t = [\mathrm{RE}]_0 + \sum_{i=1}^{3}[\mathrm{RE(HA_2)}_i]_0 \tag{C.19}$$

$$\Longrightarrow$$

$$[\mathrm{RE}]_t = [\mathrm{RE}]_0 + \cancelto{0}{\sum_{i=1}^{3}[\mathrm{RE(HA_2)}_i]_0} - \sum_{i=1}^{3}[\mathrm{RE(HA_2)}_i]_t$$

$$[\mathrm{RE}]_t = [\mathrm{RE}]_0 - \sum_{i=1}^{3}[\mathrm{RE(HA_2)}_i]_t \tag{C.20}$$

Similarly, the mass balance around A in Equation (C.21) leads to the constraint in (C.22).

$$[\mathrm{H_2A_2}]_t + \sum_{i=1}^{3} i \times [\mathrm{RE(HA_2)}_i]_t = [\mathrm{H_2A_2}]_0 + \sum_{i=1}^{3} i \times [\mathrm{RE(HA_2)}_i]_0 \tag{C.21}$$

$$\Longrightarrow$$

$$[\mathrm{H_2A_2}]_t = [\mathrm{H_2A_2}]_0 + \cancelto{0}{\sum_{i=1}^{3} i \times [\mathrm{RE(HA_2)}_i]_0} - \sum_{i=1}^{3} i \times [\mathrm{RE(HA_2)}_i]_t$$

$$[\mathrm{H_2A_2}]_t = [\mathrm{H_2A_2}]_0 - \sum_{i=1}^{3} i \times [\mathrm{RE(HA_2)}_i]_t \tag{C.22}$$

Lastly, the balance around [H] in (C.23) is used in conjunction with the constraint (C.22) to find a constraining equation for $[\mathrm{H}]_t$, as shown in (C.24).

$$[\mathrm{H}]_t + [\mathrm{H_2A_2}]_t = [\mathrm{H}]_0 + [\mathrm{H_2A_2}]_0 \tag{C.23}$$

$$[\mathrm{H}]_t = [\mathrm{H}]_0 + [\mathrm{H_2A_2}]_0 - \left([\mathrm{H_2A_2}]_0 - \sum_{i=1}^{3} i \times [\mathrm{RE(HA_2)}_i]_t\right)$$

$$[\mathrm{H}]_t = [\mathrm{H}]_0 + \sum_{i=1}^{3} i \times [\mathrm{RE(HA_2)}_i]_t \tag{C.24}$$

The equations in (C.20), (C.22), and (C.24) provide concentrations for all elements besides those which are RE bonded with $\mathrm{HA_2}$. Differential equations based on the law of mass action are required for these complexes, and are shown as Equations (C.25), (C.26), and (C.27).

$$\begin{aligned}\frac{d[\mathrm{RE(HA_2)}]}{dt} &= k_{+1}^{\mathrm{RE}}[\mathrm{RE}][\mathrm{H_2A_2}] + k_{-2}^{\mathrm{RE}}[\mathrm{RE(HA_2)_2}][\mathrm{H}] \\ &\quad - k_{-1}^{\mathrm{RE}}[\mathrm{RE(HA_2)}][\mathrm{H}] - k_{+2}^{\mathrm{RE}}[\mathrm{RE(HA_2)}][\mathrm{H_2A_2}] \end{aligned} \tag{C.25}$$

$$\begin{aligned}\frac{d[\mathrm{RE(HA_2)_2}]}{dt} &= k_{+2}^{\mathrm{RE}}[\mathrm{RE(HA_2)}][\mathrm{H_2A_2}] + k_{-3}^{\mathrm{RE}}[\mathrm{RE(HA_2)_3}][\mathrm{H}] \\ &\quad - k_{-2}^{\mathrm{RE}}[\mathrm{RE(HA_2)_2}][\mathrm{H}] - k_{+3}^{\mathrm{RE}}[\mathrm{RE(HA_2)_2}][\mathrm{H_2A_2}] \end{aligned} \tag{C.26}$$

$$\frac{d[\mathrm{RE(HA_2)_3}]}{dt} = k_{+3}^{\mathrm{RE}}[\mathrm{RE(HA_2)_2}][\mathrm{H_2A_2}] - k_{-3}^{\mathrm{RE}}[\mathrm{RE(HA_2)_3}][\mathrm{H}] \tag{C.27}$$

In practice, Equations (C.20), (C.22), and (C.24) are substituted in for variables [RE], [$\mathrm{H_2A_2}$], and [H] which allows for the differential equations to only be dependent on the initial conditions as well as [$\mathrm{RE(HA_2)}$], [$\mathrm{RE(HA_2)_2}$], [$\mathrm{RE(HA_2)_3}$]. The system of differential equations can then be solved with the Runge-Kutta algorithm.

Sodium concentrations in the organic phase must be taken into account. Since the sodium ion has a charge of just +1, only one elementary reaction is used for equilibrium modeling. However, because the fresh organic is saponified with NaOH, there will be sodium complexed with the extractant before the shake test takes place. The constraint for modeling sodium in the aqueous phase is in (C.28).

$$[\mathrm{Na}]_t = [\mathrm{NaHA_2}]_0 + [\mathrm{NA}]_0 - [\mathrm{NaHA_2}]_t \tag{C.28}$$

The differential equation for modeling sodium concentration in the organic phase is shown in Equation (C.29).

$$\frac{d[\mathrm{NaHA_2}]}{dt} = k_{+1}^{\mathrm{Na}}[\mathrm{Na}][\mathrm{H_2A_2}] - k_{-1}^{\mathrm{Na}}[\mathrm{H}][\mathrm{NaHA_2}] \tag{C.29}$$

For modeling multiple elements, the constraints, notation, and set of differential equations need only to be adjusted slightly. A metal $\mathrm{M}_j^{\alpha_j}$, may require α_j mols of $\mathrm{H_2A_2}$ to form the complex $\mathrm{M(HA_2)}_{\alpha_j}$. To model all M_j's in the set of metals Ω, the simplifications in (C.30), (C.31), and (C.32) can be made. To model the metals complexed with organophosphorous acid, a_j differential equations are required for each M_j. $\sum_{j=1}^{p} a_j$ differential equations with $2 \times \sum_{j=1}^{p} a_j$ kinetic constants are required, where $p =$ the length of Ω.

$$[\mathrm{M}_j]_t = [\mathrm{M}_j]_0 - \sum_{i=1}^{a_j} [\mathrm{M}_j(\mathrm{HA}_2)_i]_t \tag{C.30}$$

$$[\mathrm{H}_2\mathrm{A}_2]_t = [\mathrm{H}_2\mathrm{A}_2]_0 - \sum_{j=1}^{p} \sum_{i=1}^{a_j} i \times [\mathrm{M}_j(\mathrm{HA}_2)_i]_t - [\mathrm{NaHA}_2]_t + [\mathrm{NaHA}_2]_0 \tag{C.31}$$

$$[\mathrm{H}]_t = [\mathrm{H}]_0 + \sum_{j=1}^{p} \sum_{i=1}^{a_j} i \times [\mathrm{M}_j(\mathrm{HA}_2)_i]_t + [\mathrm{NaHA}_2]_t - [\mathrm{NaHA}_2]_0 \tag{C.32}$$

References

Al-Thyabat, S. 2008. “On the Optimization of Froth Flotation by the Use of an Artificial Neural Network.” *Journal of China University of Mining and Technology* 18 (3): 418–26.

Anderson, William J. 2012. *Continuous-Time Markov Chains: An Applications-Oriented Approach.* Springer Science & Business Media.

Ankenman, Bruce, Barry L Nelson, and Jeremy Staum. 2008. “Stochastic Kriging for Simulation Metamodeling.” In *2008 Winter Simulation Conference*, 362–70. IEEE.

Arendt, Paul D, Daniel W Apley, and Wei Chen. 2016. “A Preposterior Analysis to Predict Identifiability in the Experimental Calibration of Computer Models.” *IIE Transactions* 48 (1): 75–88.

Arnaut, Luis, and Hugh Burrows. 2006. *Chemical Kinetics: From Molecular Structure to Chemical Reactivity.* Elsevier.

Ashton, Gregory, Moritz Hübner, Paul D Lasky, Colm Talbot, Kendall Ackley, Sylvia Biscoveanu, Qi Chu, et al. 2019. “BILBY: A User-Friendly Bayesian Inference Library for Gravitational-Wave Astronomy.” *The Astrophysical Journal Supplement Series* 241 (2): 27.

Aslan, NEVZAT. 2008. “Application of Response Surface Methodology and Central Composite Rotatable Design for Modeling and Optimization of a Multi-Gravity Separator for Chromite Concentration.” *Powder Technology* 185 (1): 80–86.

Bai, Shuanghua, Jules Thibault, and David D McLean. 2006. “Dynamic Data Reconciliation: Alternative to Kalman Filter.” *Journal of Process Control* 16 (5): 485–98.

Balaram, V. 2019. “Rare Earth Elements: A Review of Applications, Occurrence, Exploration, Analysis, Recycling, and Environmental Impact.” *Geoscience Frontiers* 10 (4): 1285–1303.

Bartos, Paul J. 2007. “Is Mining a High-Tech Industry?: Investigations into Innovation and Productivity Advance.” *Resources Policy* 32 (4): 149–58.

Bastos, Leonardo S, and Anthony O’Hagan. 2009. “Diagnostics for Gaussian Process Emulators.” *Technometrics* 51 (4): 425–38.

Bayarri, MJ, JO Berger, and F Liu. 2009. “Modularization in Bayesian Analysis, with Emphasis on Analysis of Computer Models.” *Bayesian Analysis* 4 (1): 119–50.

Berger, James O. 2013. *Statistical Decision Theory and Bayesian Analysis.* Springer Science & Business Media.

Bernstein, Dennis S. 2009. *Matrix Mathematics.* Princeton university press.

Bethea, Robert M, and R Russell Rhinehart. 1991. *Applied Engineering Statistics.* Vol. 121. CRC Press.

Binois, Mickaël, and Robert B Gramacy. 2021a. *hetGP: Heteroskedastic Gaussian Process Modeling and Design Under Replication.*

———. 2021b. "hetGP: Heteroskedastic Gaussian Process Modeling and Sequential Design in r." *Journal of Statistical Software* 98 (1): 1–44.

Binois, Mickaël, Robert B Gramacy, and Mike Ludkovski. 2018. "Practical Heteroscedastic Gaussian Process Modeling for Large Simulation Experiments." *Journal of Computational and Graphical Statistics* 27 (4): 808–21.

Binois, Mickaël, Jiangeng Huang, Robert B Gramacy, and Mike Ludkovski. 2019. "Replication or Exploration? Sequential Design for Stochastic Simulation Experiments." *Technometrics* 61 (1): 7–23.

Bourgault, Gilles, and Denis Marcotte. 1991. "Multivariable Variogram and Its Application to the Linear Model of Coregionalization." *Mathematical Geology* 23 (7): 899–928.

Box, George EP. 1976. "Science and Statistics." *Journal of the American Statistical Association* 71 (356): 791–99.

Box, George EP, and Norman R Draper. 2007. *Response Surfaces, Mixtures, and Ridge Analyses.* John Wiley & Sons.

Brynjarsdottir, J, and A O'Hagan. 2014. "Learning about Physical Parameters: The Importance of Model Discrepancy." *Inverse Problems* 30 (11): 114007.

Bu, Xiangning, Guangyuan Xie, Yaoli Peng, and Yuran Chen. 2016. "Kinetic Modeling and Optimization of Flotation Process in a Cyclonic Microbubble Flotation Column Using Composite Central Design Methodology." *International Journal of Mineral Processing* 157: 175–83.

Canada, Natural Resources. 2021. "Government of Canada." *Natural Resources Canada.* / Gouvernement du Canada. https://www.nrcan.gc.ca/our-natural-resources/minerals-mining/critical-minerals/23414.

Cao, Songling, and R Russell Rhinehart. 1995. "An Efficient Method for on-Line Identification of Steady State." *Journal of Process Control* 5 (6): 363–74.

Carvalho, Carlos M, and James G Scott. 2009. "Objective Bayesian Model Selection in Gaussian Graphical Models." *Biometrika* 96 (3): 497–512.

Casella, George, and Edward I George. 1992. "Explaining the Gibbs Sampler." *The American Statistician* 46 (3): 167–74.

Cencic, Oliver, and Rudolf Frühwirth. 2015. "A General Framework for Data Reconciliation-Part i: Linear Constraints." *Computers & Chemical Engineering* 75: 196–208.

Chen, Anran, Shixing Wang, Libo Zhang, and Jinhui Peng. 2015. "Optimization of the Microwave Roasting Extraction of Palladium and Rhodium from Spent Automobile Catalysts Using Response Surface Analysis." *International Journal of Mineral Processing* 143: 18–24.

Chen, J, A Bandoni, and JA Romagnoli. 1997. "Robust Estimation of Measurement Error Variance/Covariance from Process Sampling Data." *Computers & Chemical Engineering* 21 (6): 593–600.

Chen, Luonan, Ruiqi Wang, Chunguang Li, and Kazuyuki Aihara. 2010. *Modeling Biomolecular Networks in Cells: Structures and Dynamics.* Springer

Science & Business Media.

Chen, Ming-Hui, and Qi-Man Shao. 1999. “Monte Carlo Estimation of Bayesian Credible and HPD Intervals.” *Journal of Computational and Graphical Statistics* 8 (1): 69–92.

Chen, Tao, Julian Morris, and Elaine Martin. 2007. “Gaussian Process Regression for Multivariate Spectroscopic Calibration.” *Chemometrics and Intelligent Laboratory Systems* 87 (1): 59–71.

Chib, Siddhartha. 1995. “Marginal Likelihood from the Gibbs Output.” *Journal of the American Statistical Association* 90 (432): 1313–21.

Cohn, DA. 1994. “Neural Network Exploration Using Optimal Experiment Design.” In *Advances in Neural Information Processing Systems*, 679–86.

Cole, D Austin, Ryan B Christianson, and Robert B Gramacy. 2021. “Locally Induced Gaussian Processes for Large-Scale Simulation Experiments.” *Statistics and Computing* 31 (3): 1–21.

Cole, D Austin, Robert B Gramacy, James E Warner, Geoffrey F Bomarito, Patrick E Leser, and William P Leser. 2021. “Entropy-Based Adaptive Design for Contour Finding and Estimating Reliability.” *arXiv Preprint arXiv:2105.11357*.

Darouach, Mohamed, M Zasadzinski, G Krzakala, and J Ragot. 1990. “Maximum Likelihood Estimator of Measurement Error Variances in Data Reconciliation.” In *Advanced Information Processing in Automatic Control (AIPAC’89)*, 109–12. Elsevier.

Darvey, Ivan G, BW Ninham, and PJ Staff. 1966. “Stochastic Models for Second-Order Chemical Reaction Kinetics. The Equilibrium State.” *The Journal of Chemical Physics* 45 (6): 2145–55.

DasGupta, Anirban. 2008. *Asymptotic Theory of Statistics and Probability*. Springer Science & Business Media.

Deisenroth, Marc Peter, Dieter Fox, and Carl Edward Rasmussen. 2013. “Gaussian Processes for Data-Efficient Learning in Robotics and Control.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37 (2): 408–23.

Demir, Funda, and Emek Moroydor Derun. 2019. “Modelling and Optimization of Gold Mine Tailings Based Geopolymer by Using Response Surface Method and Its Application in Pb2+ Removal.” *Journal of Cleaner Production* 237: 117766.

Efron, Bradley. 1979. “Computers and the Theory of Statistics: Thinking the Unthinkable.” *SIAM Review* 21 (4): 460–80.

Emery, Xavier. 2009. “The Kriging Update Equations and Their Application to the Selection of Neighboring Data.” *Computational Geosciences* 13 (3): 269–80.

Espenson, James H. 1995. *Chemical Kinetics and Reaction Mechanisms*. Vol. 102. Citeseer.

Fernandez, Carmen, Eduardo Ley, and Mark FJ Steel. 2001. “Benchmark Priors for Bayesian Model Averaging.” *Journal of Econometrics* 100 (2): 381–427.

Feroz, F, MP Hobson, and M Bridges. 2009. “MultiNest: An Efficient and Robust Bayesian Inference Tool for Cosmology and Particle Physics.” *Monthly Notices of the Royal Astronomical Society* 398 (4): 1601–14.

Finnveden, Göran, Michael Z Hauschild, Tomas Ekvall, Jeroen Guinée, Reinout Heijungs, Stefanie Hellweg, Annette Koehler, David Pennington, and Sangwon Suh. 2009. "Recent Developments in Life Cycle Assessment." *Journal of Environmental Management* 91 (1): 1–21.

Gagniuc, Paul A. 2017. *Markov Chains: From Theory to Implementation and Experimentation*. John Wiley & Sons.

Gardner, Jacob, Geoff Pleiss, Kilian Q Weinberger, David Bindel, and Andrew G Wilson. 2018. "Gpytorch: Blackbox Matrix-Matrix Gaussian Process Inference with Gpu Acceleration." *Advances in Neural Information Processing Systems* 31.

Gbor, Philip K, and Charles Q Jia. 2004. "Critical Evaluation of Coupling Particle Size Distribution with the Shrinking Core Model." *Chemical Engineering Science* 59 (10): 1979–87.

Gelfand, Alan E, Susan E Hills, Amy Racine-Poon, and Adrian FM Smith. 1990. "Illustration of Bayesian Inference in Normal Data Models Using Gibbs Sampling." *Journal of the American Statistical Association* 85 (412): 972–85.

Gelfand, Alan E, and Erin M Schliep. 2016. "Spatial Statistics and Gaussian Processes: A Beautiful Marriage." *Spatial Statistics* 18: 86–104.

Gelman, Andrew et al. 2006. "Prior Distributions for Variance Parameters in Hierarchical Models (Comment on Article by Browne and Draper)." *Bayesian Analysis* 1 (3): 515–34.

Gelman, Andrew. 2008. "Objections to Bayesian Statistics." *Bayesian Analysis* 3 (3): 445–49.

George, Edward I, and Robert E McCulloch. 1993. "Variable Selection via Gibbs Sampling." *Journal of the American Statistical Association* 88 (423): 881–89.

Geweke, John et al. 1991. *Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments*. Vol. 196. Federal Reserve Bank of Minneapolis, Research Department Minneapolis, MN.

Goh, Joslin, Derek Bingham, James Paul Holloway, Michael J Grosskopf, Carolyn C Kuranz, and Erica Rutter. 2013. "Prediction and Computer Model Calibration Using Outputs from Multifidelity Simulators." *Technometrics* 55 (4): 501–12.

Gonzalez, Javier, Joseph Longworth, David C James, and Neil D Lawrence. 2015. "Bayesian Optimization for Synthetic Gene Design." *arXiv Preprint arXiv:1505.01627*.

Goonan, Thomas G. 2011. "Rare Earth Elements: End Use and Recyclability." US Geological Survey. https://doi.org/10.3133/sir20115094.

Gosen, Bradley S. Van, Philip L. Verplanck, Keith R. Long, Joseph Gambogi, and Robert R. Seal. 2014. "The Rare-Earth Elements: Vital to Modern Technologies and Lifestyles." US Geological Survey. https://doi.org/10.3133/fs20143078.

Gramacy, Robert B. 2005. *Bayesian Treed Gaussian Process Models*. University of California, Santa Cruz.

———. 2016. "laGP: Large-Scale Spatial Modeling via Local Approximate Gaussian Processes in r." *Journal of Statistical Software* 72: 1–46.

———. 2020. *Surrogates: Gaussian Process Modeling, Design and Optimization for the Applied Sciences.* Boca Raton, Florida: Chapman Hall/CRC.

Gramacy, Robert B, and Matthew Taddy. 2009. "Categorical Inputs, Sensitivity Analysis, Optimization and Importance Tempering with Tgp Version 2, an r Package for Treed Gaussian Process Models." *University of Cambridge Statistical Laboratory Tech. Rep.*

Gu, M. 2019. "Jointly Robust Prior for Gaussian Stochastic Process in Emulation, Calibration and Variable Selection." *Bayesian Analysis* 14 (3): 857–85.

Gupta, Ankur, and James B Rawlings. 2014. "Comparison of Parameter Estimation Methods in Stochastic Chemical Kinetic Models: Examples in Systems Biology." *AIChE Journal* 60 (4): 1253–68.

Gupta, Chiranjib Kumar, and Nagaiyar Krishnamurthy. 1992. "Extractive Metallurgy of Rare Earths." *International Materials Reviews* 37 (1): 197–248.

Hartman, Howard L, and Jan M Mutmansky. 2002. *Introductory Mining Engineering.* John Wiley & Sons.

Hastings, W Keith. 1970. "Monte Carlo Sampling Methods Using Markov Chains and Their Applications."

Hellman, Phillip L, and Robert K Duncan. 2014. "Evaluation of Rare Earth Element Deposits." *Applied Earth Science* 123 (2): 107–17.

Henckens, MLCM, EC Van Ierland, PPJ Driessen, and E Worrell. 2016. "Mineral Resources: Geological Scarcity, Market Price Trends, and Future Generations." *Resources Policy* 49: 102–11.

Higdon, Dave, Marc Kennedy, James C. Cavendish, John A. Cafeo, and Robert D. Ryne. 2004. "Combining Field Data and Computer Simulations for Calibration and Prediction." *SIAM Journal on Scientific Computing* 26 (2): 448–66.

Hoff, Peter. 2009. *A First Course in Bayesian Statistical Methods.* New York London: Springer.

Honaker, RQ, J Groppo, R-H Yoon, GH Luttrell, A Noble, and J Herbst. 2017. "Process Evaluation and Flowsheet Development for the Recovery of Rare Earth Elements from Coal and Associated Byproducts." *Minerals & Metallurgical Processing* 34 (3): 107–15.

Horn, Fritz, and Roy Jackson. 1972. "General Mass Action Kinetics." *Archive for Rational Mechanics and Analysis* 47 (2): 81–116.

Huang, Jiangeng, Robert B. Gramacy, Mickaël Binois, and Mirko Libraschi. 2020. "On-Site Surrogates for Large-Scale Calibration." *Applied Stochastic Models in Business and Industry* 36 (2): 283–304.

Huang, Qingqing, Aaron Noble, John Herbst, and Rick Honaker. 2018. "Liberation and Release of Rare Earth Minerals from Middle Kittanning, Fire Clay, and West Kentucky No. 13 Coal Sources." *Powder Technology* 332: 242–52.

Huelsenbeck, John P, Fredrik Ronquist, Rasmus Nielsen, and Jonathan P Bollback. 2001. "Bayesian Inference of Phylogeny and Its Impact on Evolutionary Biology." *Science* 294 (5550): 2310–14.

Hvala, Nadja, and Juš Kocijan. 2020. "Design of a Hybrid Mechanistic/Gaussian Process Model to Predict Full-Scale Wastewater Treatment Plant Effluent."

Computers & Chemical Engineering 140: 106934.

IEA, Paris. 2021. "The Role of Critical Minerals in Clean Energy Transitions." /url%7Bhttps://pubs.usgs.gov/periodicals/mcs2021/mcs2021-rare-earths.pdf%7D.

Jeffreys, Harold. 1935. "Some Tests of Significance, Treated by the Theory of Probability." In *Mathematical Proceedings of the Cambridge Philosophical Society*, 31:203–22. 2. Cambridge University Press.

———. 1946. "An Invariant Form for the Prior Probability in Estimation Problems." *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 186 (1007): 453–61.

Johnson, L. R. 2008. "Microcolony and Biofilm Formation as a Survival Strategy for Bacteria." *Journal of Theoretical Biology* 251: 24–34.

Johnson, Mark E, Leslie M Moore, and Donald Ylvisaker. 1990. "Minimax and Maximin Distance Designs." *Journal of Statistical Planning and Inference* 26 (2): 131–48.

Jones, DR, M Schonlau, and WJ Welch. 1998. "Efficient Global Optimization of Expensive Black-Box Functions." *Journal of Global Optimization* 13 (4): 455–92.

Kass, Robert E, and Adrian E Raftery. 1995. "Bayes Factors." *Journal of the American Statistical Association* 90 (430): 773–95.

Kass, Robert E, and Duane Steffey. 1989. "Approximate Bayesian Inference in Conditionally Independent Hierarchical Models (Parametric Empirical Bayes Models)." *Journal of the American Statistical Association* 84 (407): 717–26.

Kass, Robert E, and Larry Wasserman. 1996. "The Selection of Prior Distributions by Formal Rules." *Journal of the American Statistical Association* 91 (435): 1343–70.

Keller, JY, M Zasadzinski, and M Darouach. 1992. "Analytical Estimator of Measurement Error Variances in Data Reconciliation." *Computers & Chemical Engineering* 16 (3): 185–88.

Kennedy, Marc C, and Anthony O'Hagan. 2001. "Bayesian Calibration of Computer Models." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63 (3): 425–64.

Klöpffer, Walter. 1997. "Life Cycle Assessment." *Environmental Science and Pollution Research* 4 (4): 223–28.

Koermer, Scott. 2020a. *BayesMassBal: Bayesian Data Reconciliation of Separation Processes.* https://github.com/skoermer/BayesMassBal.

———. 2020b. "The Utility of Bayesian Data Reconciliation for Separations: Validation Data and Script."

Koermer, Scott, and Aaron Noble. 2021. "The Utility of Bayesian Data Reconciliation for Separations." *Minerals Engineering* 169: 106837.

Krishna, Arvind, V Roshan Joseph, Shan Ba, William A Brenneman, and William R Myers. 2021. "Robust Experimental Designs for Model Calibration." *Journal of Quality Technology*, 1–12.

Larochelle, Tommee, and Henry Kasaini. 2016. "Predictive Modeling of Rare Earth Element Separation by Solvent Extraction Using Metsim." *Proceedings of the IMPC.*

Leatherman, Erin R, Angela M Dean, and Thomas J Santner. 2017. "Designing Combined Physical and Computer Experiments to Maximize Prediction Accuracy." *Computational Statistics & Data Analysis* 113: 346–62.

Li, Cheng, David Rubin de Celis Leal, Santu Rana, Sunil Gupta, Alessandra Sutti, Stewart Greenhill, Teo Slezak, Murray Height, and Svetha Venkatesh. 2017. "Rapid Bayesian Optimisation for Synthesis of Short Polymer Fiber Materials." *Scientific Reports* 7 (1): 1–10.

Lide, David R. 2004. *CRC Handbook of Chemistry and Physics.* Vol. 85. CRC press.

Litterman, Robert B. 1986. "Forecasting with Bayesian Vector Autoregressions—Five Years of Experience." *Journal of Business & Economic Statistics* 4 (1): 25–38.

Liu, Jun S, and Rong Chen. 1995. "Blind Deconvolution via Sequential Imputations." *Journal of the American Statistical Association* 90 (430): 567–76.

Lo, Teh C, Malcolm HI Baird, and Carl Hanson. 1983. *Handbook of Solvent Extraction.* Wiley New York.

Lyman, Geoffrey J. 2020. "Theory and Practice of Particulate Sampling: An Engineering Approach." In *TOS Forum.* 1. IM Publications Open.

Lyon, Kevin L, Vivek P Utgikar, and Mitchell R Greenhalgh. 2017. "Dynamic Modeling for the Separation of Rare Earth Elements Using Solvent Extraction: Predicting Separation Performance Using Laboratory Equilibrium Data." *Industrial & Engineering Chemistry Research* 56 (4): 1048–56.

MacKay, David JC. 1992. "Information-Based Objective Functions for Active Data Selection." *Neural Computation* 4 (4): 590–604.

Madron, Frantisek. 1992. *Process Plant Performance : Measurement and Data Processing for Optimization and Retrofits.* New York: Ellis Horwood.

Makni, S, and D Hodouin. 1994. "Recursive BILMAT Algorithm: An on-Line Extension of Data Reconciliation Techniques for Steady-State Bilinear Material Balance." *Minerals Engineering* 7 (9): 1179–91.

Marrel, A, B Iooss, B Laurent, and O Roustant. 2009. "Calculations of Sobol Indices for the Gaussian Process Metamodel." *Reliability Engineering & System Safety* 94 (3): 742–51.

Matheron, Georges. 1963. "Principles of Geostatistics." *Economic Geology* 58 (8): 1246–66.

McCabe, Warren L, and EW Thiele. 1925. "Graphical Design of Fractionating Columns." *Industrial & Engineering Chemistry* 17 (6): 605–11.

McKay, Michael D, Richard J Beckman, and William J Conover. 1979. "Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code." *Technometrics* 21 (2): 239–45.

———. 2000. "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code." *Technometrics* 42 (1): 55–61.

McQuarrie, Donald A. 1967. "Stochastic Approach to Chemical Kinetics." *Journal of Applied Probability* 4 (3): 413–78.

Metropolis, Nicholas, and Stanislaw Ulam. 1949. "The Monte Carlo Method." *Journal of the American Statistical Association* 44 (247): 335–41.

Mitchell, Toby J, and John J Beauchamp. 1988. "Bayesian Variable Selection in Linear Regression." *Journal of the American Statistical Association* 83 (404): 1023–32.

Morris, Max D. 2015. "Physical Experimental Design in Support of Computer Model Development." *Technometrics* 57 (1): 45–53.

Nariyan, Elham, Mika Sillanpää, and Christian Wolkersdorfer. 2018. "Uranium Removal from Pyhäsalmi/Finland Mine Water by Batch Electrocoagulation and Optimization with the Response Surface Methodology." *Separation and Purification Technology* 193: 386–97.

Nassar, Nedal T., and Steven M. Fortier. 2021. "Methodology and Technical Input for the 2021 Review and Revision of the u.s. Critical Minerals List." US Geological Survey. https://doi.org/10.3133/ofr20211045.

Neal, Radford M. 2003. "Slice Sampling." *The Annals of Statistics* 31 (3): 705–67.

Nel, Eugene, Chris Martin, and Hans Rabbe-Sgs. 2004. "PGM ORE PROCESSING AT IMPALALS UG-2 CONCENTRATOR IN RUSTENBURG, SOUTH AFRICA." *Technical Bulletin* 2.

Núñez-Gómez, Dámaris, Flávio Rubens Lapolli, Maria Elisa Nagel-Hassemer, and María Ángeles Lobo-Recio. 2020. "Optimization of Fe and Mn Removal from Coal Acid Mine Drainage (AMD) with Waste Biomaterials: Statistical Modeling and Kinetic Study." *Waste and Biomass Valorization* 11 (3): 1143–57.

Nuzzo, Regina. 2014. "Scientific Method: Statistical Errors." *Nature News* 506 (7487): 150.

Paula, Julio de. 2006. *ATKINS'PHYSICAL CHEMISTRY*. WH Freeman; Company New York.

Pavón, Sandra, Agustí Fortuny, M Teresa Coll, and Ana M Sastre. 2019. "Solvent Extraction Modeling of Ce/Eu/y from Chloride Media Using D2ehpa." *AIChE Journal* 65 (8): e16627.

Pitard, Francis F. 1993. *Pierre Gy's Sampling Theory and Sampling Practice: Heterogeneity, Sampling Correctness, and Statistical Process Control*. CRC press.

Plumlee, Matthew. 2017. "Bayesian Calibration of Inexact Computer Models." *Journal of the American Statistical Association* 112 (519): 1274–85.

———. 2019. "Computer Model Calibration with Confidence and Consistency." *Journal of the Royal Statistical Society: Series B* 81 (3): 519–45.

Plummer, Martyn, Nicky Best, Kate Cowles, and Karen Vines. 2006. "CODA: Convergence Diagnosis and Output Analysis for MCMC." *R News* 6 (1): 7–11. https://journal.r-project.org/archive/.

Qian, Peter Z G, Huaiqing Wu, and CF Jeff Wu. 2008. "Gaussian Process Models for Computer Experiments with Qualitative and Quantitative Factors." *Technometrics* 50 (3): 383–96.

Qian, Peter ZG. 2012. "Sliced Latin Hypercube Designs." *Journal of the American Statistical Association* 107 (497): 393–99.

Ranjan, Pritam, Derek Bingham, and George Michailidis. 2008. "Sequential Experiment Design for Contour Estimation from Complex Computer Codes." *Technometrics* 50 (4): 527–41.

Ranjan, Pritam, Wilson Lu, Derek Bingham, Shane Reese, Brian J Williams, Chuan-Chih Chou, Forrest Doss, Michael Grosskopf, and James Paul Holloway. 2011. "Follow-up Experimental Designs for Computer Models and Physical Processes." *Journal of Statistical Theory and Practice* 5 (1): 119–36.

Rasmussen, Carl Edward. 2003. "Gaussian Processes in Machine Learning." In *Summer School on Machine Learning*, 63–71. Springer.

Reimers, Claus, Joachim Werther, and Guenter Gruhn. 2008. "Flowsheet Simulation of Solids Processes: Data Reconciliation and Adjustment of Model Parameters." *Chemical Engineering and Processing: Process Intensification* 47 (1): 138–58.

Ritcey, GM, and AW Ashbrook. 1979. "Solvent Extraction: Principles and Applications to Process." *Metallurgy* 1: 106–7.

Robbins, Herbert. 1952. "Some Aspects of the Sequential Design of Experiments." *Bulletin of the American Mathematical Society* 58 (5): 527–35.

Robert, Christian, and George Casella. 2013. *Monte Carlo Statistical Methods.* Springer Science & Business Media.

Roberto Danesi, Pier, Renato Chiarizia, and Charles F Coleman. 1980. "The Kinetics of Metal Solvent Extraction."

Romagnoli, Jose A, and Mabel Cristina Sanchez. 1999. *Data Processing and Reconciliation for Chemical Process Operations.* Elsevier.

Rötzer, Nadine, and Mario Schmidt. 2018. "Decreasing Metal Ore Grades—Is the Fear of Resource Depletion Justified?" *Resources* 7 (4): 88.

Saber, Scott William. 2018. "Characteristic Analysis of Acid Mine Drainage Precipitates for the Optimization of Rare Earth Extraction Processes." PhD thesis, Virginia Tech.

Sacks, Jerome, William J Welch, Toby J Mitchell, and Henry P Wynn. 1989. "Design and Analysis of Computer Experiments." *Statistical Science* 4 (4): 409–23.

Saltelli, Andrea. 2002. "Making Best Use of Model Evaluations to Compute Sensitivity Indices." *Computer Physics Communications* 145 (2): 280–97.

Santner, Thomas J, Brian J Williams, William I Notz, and Brain J Williams. 2003. *The Design and Analysis of Computer Experiments.* Vol. 1. Springer.

Santner, TJ, BJ Williams, and W Notz. 2018. *The Design and Analysis of Computer Experiments, Second Edition.* New York, NY: Springer–Verlag.

Sauer, Annie, Robert B Gramacy, and David Higdon. 2020. "Active Learning for Deep Gaussian Process Surrogates." *arXiv Preprint arXiv:2012.08015.*

———. 2021. "Active Learning for Deep Gaussian Process Surrogates." *Technometrics*, no. just-accepted: 1–39.

Schonlau, Matthias. 1997. "Computer Experiments and Global Optimization."

Seo, Sambu, Marko Wallat, Thore Graepel, and Klaus Obermayer. 2000. "Gaussian Process Regression: Active Data Selection and Test Point Rejection." In *Mustererkennung 2000*, 27–34. Springer.

Shewry, Michael C, and Henry P Wynn. 1987. "Maximum Entropy Sampling." *Journal of Applied Statistics* 14 (2): 165–70.

Shumway, Robert H, and David S Stoffer. 2000. *Time Series Analysis and Its Applications.* Vol. 3. Springer.

Skinner, Brian J. 1976. "Second Iron Age Ahead." *Am. Sci.;(United States)* 64

(3).

Skirrow, Roger G, David Lowell Huston, Terrence P Mernagh, Jane P Thorne, Helen Duffer, and A Senior. 2013. *Critical Commodities for a High-Tech World: Australia's Potential to Supply Global Demand.* Geoscience Australia Canberra.

Slade, Margaret E. 1982. "Trends in Natural-Resource Commodity Prices: An Analysis of the Time Domain." *Journal of Environmental Economics and Management* 9 (2): 122–37.

Tabak, John. 2014. *Geometry: The Language of Space and Form.* Infobase Publishing.

Tamhane, Ajit C, Corneliu Iordache, and Richard SH Mah. 1988. "A Bayesian Approach to Gross Error Detection in Chemical Process Data: Part i: Model Development." *Chemometrics and Intelligent Laboratory Systems* 4 (1): 33–45.

Temkin, Oleg N, Andrew V Zeigarnik, and DG Bonchev. 1996. *Chemical Reaction Networks: A Graph-Theoretical Approach.* CRC Press.

Tesch, Matthew, Jeff Schneider, and Howie Choset. 2011. "Using Response Surfaces and Expected Improvement to Optimize Snake Robot Gait Parameters." In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1069–74. IEEE.

The European Commission. 2020. *Critical Raw Materials Resilience: Charting a Path Towards Greater Security and Sustainability.* The European Commission. https://ec.europa.eu/transparency/documents-register/detail?ref=COM(2020)474&lang=en.

Tonner, Peter D, Cynthia L Darnell, Barbara E Engelhardt, and Amy K Schmid. 2017. "Detecting Differential Growth of Microbial Populations with Gaussian Process Regression." *Genome Research* 27 (2): 320–33.

Trafimow, David, and Michael Marks. 2015. "Editorial in Basic and Applied Social Pschology." *Basic and Applied Social Pschology* 37: 1–2.

Trotta, Roberto. 2008. "Bayes in the Sky: Bayesian Inference and Model Selection in Cosmology." *Contemporary Physics* 49 (2): 71–104.

Tuo, Rui, and C. F. Jeff Wu. 2015. "Efficient Calibration for Imperfect Computer Models." *Annals of Statistics* 43 (6): 2331–52.

———. 2016. "A Theoretical Framework for Calibration in Computer Models: Parameterization, Estimation and Convergence Properties." *Journal of Uncertainty Quantification* 4: 767–95.

USGS. 2021. "Mineral Commodity Summaries 2021." US Geological Survey. https://doi.org/10.3133/mcs2021.

Vasebi, Amir, Éric Poulin, and Daniel Hodouin. 2014. "Selecting Proper Uncertainty Model for Steady-State Data Reconciliation–Application to Mineral and Metal Processing Industries." *Minerals Engineering* 65: 130–44.

Vass, Christopher R, Aaron Noble, and Paul F Ziemkiewicz. 2019. "The Occurrence and Concentration of Rare Earth Elements in Acid Mine Drainage and Treatment by-Products: Part 1—Initial Survey of the Northern Appalachian Coal Basin." *Mining, Metallurgy & Exploration* 36 (5): 903–16.

Veglio, F, and S Ubaldini. 2001. "Optimisation of Pure Stibnite Leaching Conditions by Response Surface Methodology." *European Journal of Mineral*

Processing and Environmental Protection 1 (2): 103–12.

Ver Hoef, Jay M, and Ronald Paul Barry. 1998. "Constructing and Fitting Models for Cokriging and Multivariable Spatial Prediction." *Journal of Statistical Planning and Inference* 69 (2): 275–94.

Von Neumann, John. 1941. "Distribution of the Ratio of the Mean Square Successive Difference to the Variance." *The Annals of Mathematical Statistics* 12 (4): 367–95.

Wei, Kai, Rishabh Iyer, and Jeff Bilmes. 2015. "Submodularity in Data Subset Selection and Active Learning." In *International Conference on Machine Learning*, 1954–63. PMLR.

Williams, Brian J, Jason L Loeppky, Leslie M Moore, and Mason S Macklem. 2011. "Batch Sequential Design to Achieve Predictive Maturity with Calibrated Computer Models." *Reliability Engineering & System Safety* 96 (9): 1208–19.

Williams, Christopher K, and Carl Edward Rasmussen. 2006. *Gaussian Processes for Machine Learning.* Vol. 2. 3. MIT press Cambridge, MA.

Wills, B. A. 2006. *Mineral Processing Technology : An Introduction to the Practical Aspects of Ore Treatment and Mineral Recovery.* Oxford Boston: Butterworth-Heinemann.

Wong, Raymond K. W., Curtis B. Storlie, and Thomas C. M. Lee. 2017. "A Frequentist Approach to Computer Model Calibration." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 79 (2): 635–48.

Wycoff, Nathan, Mickaël Binois, and Stefan M. Wild. 2021. "Sequential Learning of Active Subspaces." *Journal of Computational and Graphical Statistics* 30 (4): 1224–37. https://doi.org/10.1080/10618600.2021.1874962.

Yagi, Sakae, and Daizo Kunii. 1955. "Studies on Combustion of Carbon Particles in Flames and Fluidized Beds." In *Symposium (International) on Combustion*, 5:231–44. 1. Elsevier.

Yang, Ruoyong, and James O Berger. 1996. *A Catalog of Noninformative Priors.* Institute of Statistics; Decision Sciences, Duke University.

Zhang, C, X Zhang, and Bryan Schreiner. 1995. "Rare Earth Extractive Metallurgy: Principle and Process." Beijing: Metallurgy Industrial Press.

Zhang, Jack, Baodong Zhao, and Bryan Schreiner. 2016. "Separation Hydrometallurgy of Rare Earth Elements."

Zhang, Yichi, Siyu Tao, Wei Chen, and Daniel W Apley. 2020. "A Latent Variable Approach to Gaussian Process Modeling with Qualitative and Quantitative Factors." *Technometrics* 62 (3): 291–302.

Ziemkiewicz, Paul, Tom He, Aaron Noble, and Xingbo Liu. 2016. "Recovery of Rare Earth Elements (REEs) from Coal Mine Drainage." In *West Virginia Mine Drainage Task Force Symposium: Morgantown, WV, USA*.

Ziemkiewicz, Paul, Aaron Noble, and Chris Vass. U.S. Patent 10 954 582, Feb. 19, 2020. "Systems and Processes for Recovery of High-Grade Rare Earth Concentrate from Acid Mine Drainage."