PRODUCTION LINE CALIBRATION WITH DATA ANALYSIS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

By
İsmail Burak Taş
September 2022

PRODUCTION LINE CALIBRATION WITH DATA ANALYSIS
By İsmail Burak Taş
September 2022

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Savaş Dayanık(Advisor)

Arnab Basu

Kağan Gökbayrak

Approved for the Graduate School of Engineering and Science:

Orhan Arıkan
Director of the Graduate School

ABSTRACT

PRODUCTION LINE CALIBRATION WITH DATA ANALYSIS

İsmail Burak Taş
M.S. in Industrial Engineering
Advisor: Savaş Dayanık
September 2022

Product weights can be statistically related to controllable and uncontrollable factors of the production processes. Uncontrollable factors may be correlated with controllable factors. We fitted a response surface approximator of product weights and found sub-optimal controllable factors' values that minimize product weight. Furthermore, we found that the uncertainty of uncontrollable variables and the correlation among them may affect the result of product weight minimization. The company may implement these findings to reduce the cost of production. Also, we formulated a fully Bayesian experimental design problem to minimize product weight tolerance limits and built hierarchical models. Posterior distributions of the hierarchical models' parameters can be simulated by a Gibbs sampler. However, we conclude that the effectiveness and convergence of the Gibbs sampler may not be robust to candidate design settings while searching over the design space to solve the experimental design problem.

Keywords: Fully Bayesian experimental design, Bayesian hierarchical models, Markov chain Monte Carlo, Robust design.

ÖZET

VERİ ANALİZİ İLE ÜRETİM HATTI KALİBRASYONU

İsmail Burak Taş
Endüstri Mühendisliği, Yüksek Lisans
Tez Danışmanı: Savaş Dayanık
Eylül 2022

Ürün ağırlıkları üretim sürecinin hem kontrol edilebilir hem de kontrol edilemeyen faktörler ile istatistiksel olarak bağlantı olabilir. Kontrol edilemeyen faktörlerin kontrol edilebilenlerle korelasyona sahip olabilir. Ürün ağırlıklarını enküçükleyen altı kontrol edilebilir faktörlerin değerlerini bulduk. Buna ek olarak, kontrol edilemeyen faktörlerin belirsizliğini ve aralarındaki korelasyonun ürün ağırlığı enküçüklemesini etkileyebileceğini bulduk. Bulgular, şirketin üretim maaliyetlerini azatlmasını sağlayabilir. Ayrıca, doğrudan ürün ağırlığını enküçüklemesini geliştirmeyi hedefleyen tamamıyla Bayesian olan bir deneysel tasarım problemi formüle ettik. Bayesian hiyerarşik bir model kurup parametlerinin sonraki dağılımları Gibbs örnekleyicisi ile simüle ettik. Fakat bu örnekleyicinin dengeye ulaşma süresinin ve efektifliğinin, tasarım problemini çözmek için denenen aday çözüm noktalarından etkilenebileceği sonucuna ulaştık.

Anahtar sözcükler: Tamamıyla Bayesci deney tasarımı, Bayesci hiyerarşik modeller, Markov zinciri Monte Carlo, Dayanıklı tasarım.

Acknowledgement

First of all, I would like to express my sincerest gratitude to my advisor Prof. Savaş Dayanık for believing in me in the first place. He successfully taught me many things throughout the last three years with his never-ending support and patience. It has been a privilege to work under his guidance.

I would like to express my thanks to Prof. Kağan Gökbayrak and Assoc. Prof. Arnab Basu for allocating their valuable time to read and review this thesis.

I am so thankful to Şifa Çelik, Emre Düzoylum, and Efe Sertkaya for their valuable support. Without Emre Düzoylum and Efe Sertkaya, this journey might not have even started. Thanks for the life-changing friendship!

I would like to thank Seyit Ulutaş and Atahan Bayır for making 79/16 more fun! Also, I would like to thank all people who were members of EA327 and EA325 during 2020-2021.

I would like to express my special thanks to Selin Ordu for her respect and patience during my undergraduate and Masters.

Last but not least, I would like to thank my dear family, my sister Burcu, my brother-in-law İsmet, and my parents, Gülşen and Ender. Without their endless love, sacrifice, and pain, I definitely could not walk along this way.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

A manufacturer may want to improve or stabilize the measurable, specific characteristic of the end product. This characteristic, a function of the response variable of interest or itself, is expected to be affected by previous processes or product ingredients. In other words, generally, there is a set of explanatory variables $x_1, x_2, ..., x_p$ related to the response variable. Then we could write the response variable y in the form of $y = g(x_1, x_2, ..., x_p) + \varepsilon$, where ε denotes the error in the production system. In general, this function g is unknown and has to be approximated with an appropriate function f by employing an empirical model called the response surface model. If the data have been collected from the system already, it can be used to find function f, that is to say, to approximate the response surface. After that, process optimization can be conducted. That is, optimal values of the predictor set can be found that aim to stabilize or improve the function of the response variable or itself with respect to certain constraints. For example, variation of the end products' weights can be minimized while holding the mean of their weights at a specific target.

Although optimal values of the predictors are found, some of them can not be set directly in some cases, even if they have a statistical impact on the response variable. We call this subset of predictors uncontrollable factors and the remainder controllable factors. At this point, process optimization should be

conducted by taking into account the uncertainty that comes from uncontrollable factors. As a result, values of controllable factors that are optimal and robust to the uncertainty of uncontrollable factors are found and called robust design. Depending on the system, those uncontrollable factors could either be independent or dependent on the controllable factors. If they are dependent on the controllable factors, an additional probabilistic model could be built between controllable factors and uncontrollable factors in order to predict uncontrollable factors. Otherwise, the uncertainty of the uncontrollable factors could be added to the process optimization step directly.

A manufacturer may want to gather new information from the system to enhance his understanding of the system or to directly aims for better achievement of his goal. All new information gathered from the system may not effectively serve his goal. Also, observing the system can be expensive, and there could be some cost, time, and resource constraints regarding it. In those cases, the system has to be observed in a limited number of states. Statistical experiments that aim to maximize the amount of useful information gathered from the system considering the goal and such constraints can be designed, called optimal experimental design. These statistical experiments are commonly either sequential or static. If it is static, decisions are made before experimenting, regardless of the information that comes throughout the experiment. Otherwise, decisions are made consecutively courtesy of new information during the experiment.

Optimal experimental design has been widely studied in the classical framework under the name of alphabetical letters such as D-optimal, G-optimal, etc. Each one of them has different predetermined goals. For example, the A-optimal design aims to minimize the average variance of estimates of models, G-optimal instead aims to minimize the maximum variance of the predicted values. Those kinds of optimality criteria are generally derived based on the expected Fisher Information.

The decision-theoretical approach of the Bayesian optimal experimental design was first given by Lindley [1]. A design x^d selected from the set S will yield future data $y^d \in Y^d$ with respect to unknown model parameters $\beta \in \mathcal{B}$. Based

on y^d, a terminal decision x^t is selected from the set D to maximize $U(x^d)$ where utility function U represents the worth of the experiment. A probabilistic model $p(y, \beta|x^d)$ and a likelihood $p(y|x^d, \beta)$ are required as well. Utility function $U(x^d)$ for a fixed design x^d is defined as

$$U(x^d) = \int_{Y^d} \left\{ \max_{x^t \in D} \int_{\mathcal{B}} U(y^d, \beta, x^d, x^t) p(\beta|y^d, x^d) p(y^d|x^d) d\beta \right\} dy^d. \tag{1.1}$$

Optimal Bayesian experimental design x^{d*} which maximizes the equation above over the space S is given as

$$x^{d*} = \arg\max_{x^d \in S} \int_{Y^d} \left\{ \max_{x^t \in D} \int_{\mathcal{B}} U(y^d, \beta, x^d, x^t) p(\beta|y^d, x^d) p(y^d|x^d) d\beta \right\} dy^d. \tag{1.2}$$

In general, the problem above does not have a closed-form of solution and can be solved numerically.

Lindley's decision-theoretical approach can be modified depending on the experiment or goal by adhering to its concept. The terminal decision x^t can be ignored. Also, the input parameters of the utility function can be altered. If the utility function is not a functional of the posterior distribution of unknown parameters, the Bayesian experimental design is called as "pseudo-Bayesian". Otherwise, it is called a fully Bayesian experimental design [2]. Inherently, a fully Bayesian experimental design is more challenging to solve than the pseudo-Bayesian since it requires posterior calculations for every possible future realization of data. On the other hand, depending on the utility function and model specification, the results of the Bayesian experimental design can coincide with the classical framework [3].

In our study, our main interest is the calibration of the production line of a company. Although two kinds of products are produced in that production line, our project's scope covers only one of them. We are told a couple of operators operate the production line based on his/her experience. So, the optimal operation procedure was not explored. Thus, there exists variation in the products' weights which may be caused by the operators or current operating settings of

the production line. On the other hand, the company is not able to sell products whose weights are less than a predetermined threshold. This variation and threshold force the company to produce heavier products than its target weight.

The company has recorded measurements on some factors of the production process such as heat exposure time, temperature, speed of conveyor belt, etc., and has recorded measurements on products' physical features such as roundness, average diameter, etc. during the production. Finally, the end products' weights have been recorded at the end of the production. It is said that some of the measured factors, which are mostly related to the product's physical features are uncontrollable. That is to say, operators are not able to set values for uncontrollable factors and are able to set values for only controllable factors.

This thesis aims to minimize final products' weights as much as possible considering the scrap rate, in other words, without increasing the proportion of products whose weights are less than the threshold. It is aimed to achieve this objective, operating setting of controllable factors which are in operational space, optimal and robust to the uncertainty of the uncontrollable factors of the production line be found.

Preliminary data analysis is performed, and the problem definition is stated in Chapter 2. The literature review is given in Chapter 3. In Chapter 4, in the classical approach, sub-optimal values of uncontrollable factors that are not robust to the uncontrollable factors are found by solving a process optimization model. In Chapter 5, the process optimization model is converted into the Bayesian model, and sub-optimal values of uncontrollable factors that are robust to the uncontrollable factors are found. Bayesian hierarchical model and a Gibbs sampler were provided in Chapter 6. In Chapter 7, the hierarchical model is updated, and the fully Bayesian experimental problem is examined.

Chapter 2

Data and Problem Definition

2.1 Data

The original data provided by the company consists of 10,139,116 rows, and three columns in each measure type, value, time were recorded. Values of all measure types are continuous. It was observed that the value of a measure, which denotes the products' weights, sometimes abnormally stays constant for a long time. Later, it was found that the same product's weight could be recorded repetitively due to production stops. Therefore, more than two consecutive and constant values of the product's weights were filtered out. The remaining data were grouped based on the measure types and were separated into the 15 minutes buckets. Thus, each bucket contains 25 measure types, and their values were tried to fill with respect to their mean. There were some buckets that did not contain any value of some measure types. Those values were filled by the interpolation method as long as the bucket containing the product's weight value. Otherwise, the bucket is filtered out. At the end of this process, the final data set used in our study has 3694 observations of the product's weight and 24 predictors of it. Half of the 24 predictors are uncontrollable, and the remainder is controllable factors. In terms of timeline, it has intermittent records in 7 months or 76 days span.

Table 2.1: Summary statistics of the product weights

Mean	Std..Dev.	Min	Percentile.25	Percentile.50	Percentile.75	Max
107.2502	1.5953	103.3107	106.0971	107.2787	108.3093	111.8423

Histogram and summary of product weights in grams are given in Figure 2.1 and Table 2.1, respectively. The company determined the threshold for being counted as scrap as 98 grams for products and a scrap rate as less than 0.0001. The mean and median of the product weights are close, and the histogram shows that product weights can be assumed to have a normal distribution. Product weights' monthly means and standard deviations are shown in Figure 2.2. In the plot, error bars were created by using one standard deviation. Monthly standard deviations seem to be constant, but the means are changing. It might be caused by either operating settings or seasonal effects.

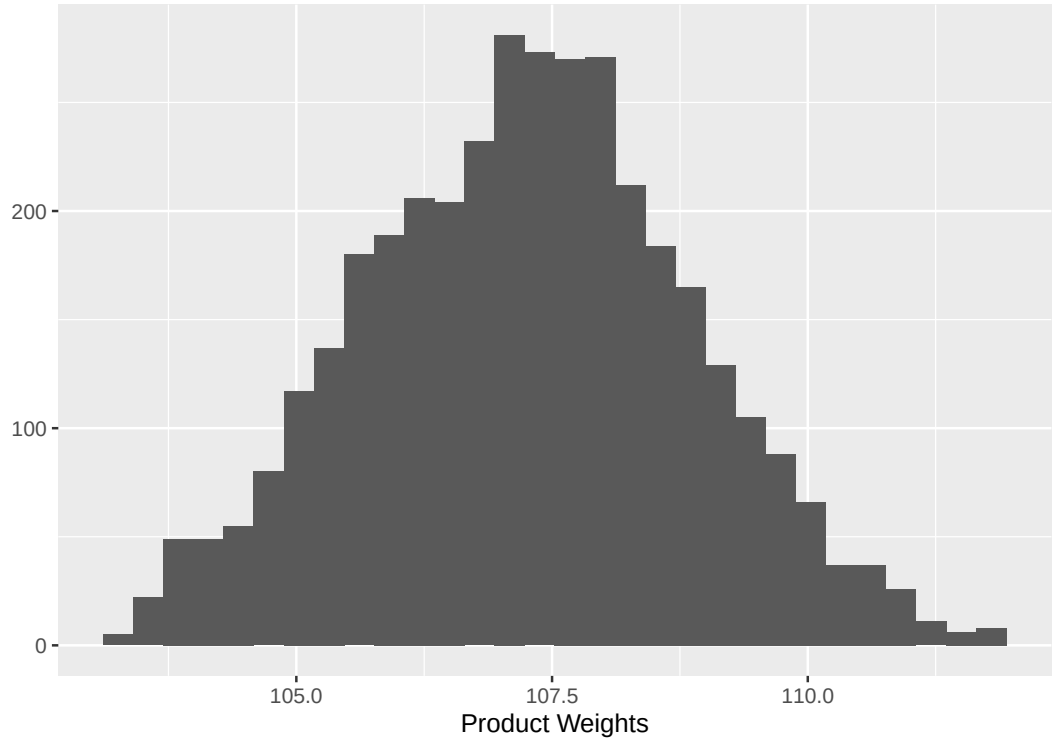

Figure 2.1: Histogram of product weights

Figure 2.3 shows the correlation matrix of controllable and uncontrollable factors. In the plot, variables beginning with "C" are controllable, and beginning

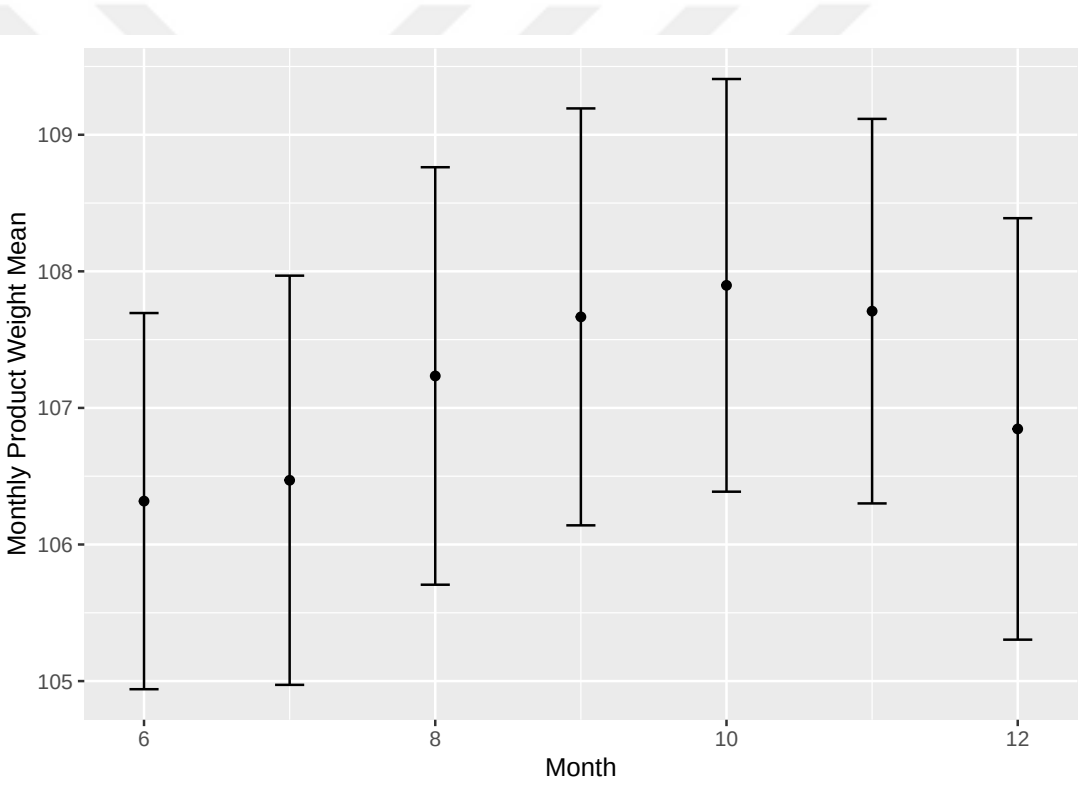

Figure 2.2: Monthly product weight means and standard deviations

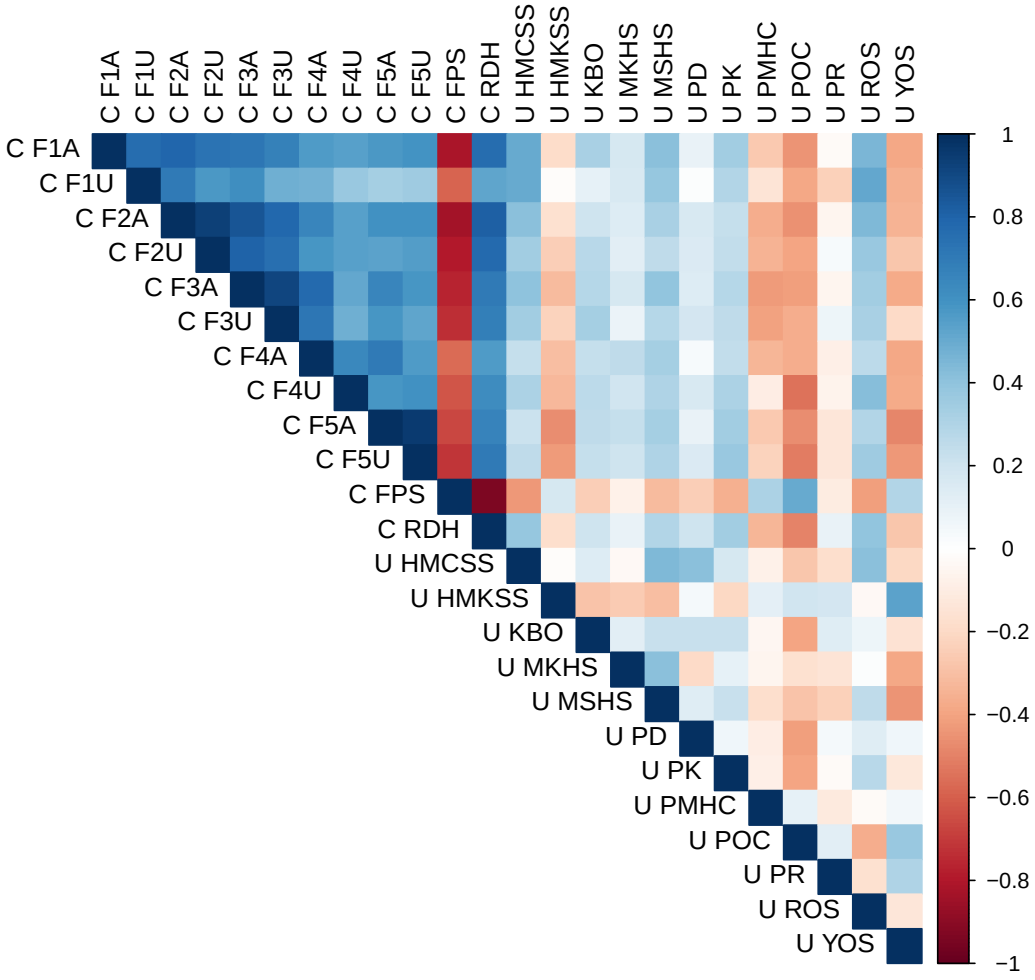

Figure 2.3: Correlation of predictors

with "U" are uncontrollable factors. The plot shows that correlations between controllable factors are strong, and it can be seen as a distinct dark small triangle on the top left. Thus, the company may not be able to set values of the controllable factors independently. However, in this study, we assumed that their values could be set independently. Uncontrollable factors tend to be correlated with controllable factors. Therefore, a probabilistic model can be considered between uncontrollable and controllable factors. Also, they are correlated with each other, as seen at the bottom of the plot.

2.2 Problem Definition

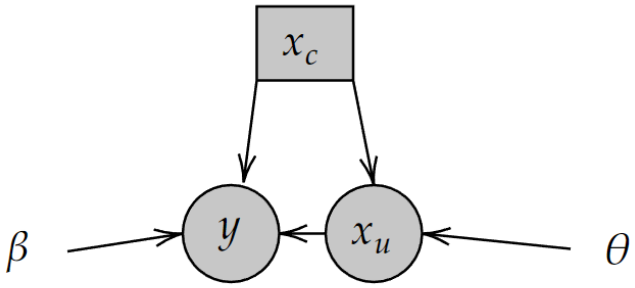

Figure 2.4: The relations between variables

Let y denote product weights. Let x_c, x_u denote controllable and uncontrollable factors, respectively. Note that we always set the first value of x_c as 1. Let y be the output of the function f of x_c, x_u and β such that $y = f(x_c, x_u, \beta)$. We assume $x_u = h(x_c, \theta)$ for some function h and parameter vector θ. Therefore, we have $y = f(x_c, h(x_c, \theta), \beta)$. Figure 2.4 summarizes the relations between

variables. Round nodes represent random variables. Square nodes represent deterministic variables. Filled nodes denote observable variables. Frequentist and Bayesian approaches treat unknown parameters β and θ as either deterministic parameters or random variables, respectively. Also, depending on the solution method, functions f, h and parameters β, θ be defined differently.

Let superscript "o" denote that we observed the variable. In the past, we have observed n pairs of y, x_u, and x_c. As a result, $y^o \in \mathcal{R}^n$, $X_u^o \in \mathcal{R}^{n \times p_u}$, $X_c^o \in \mathcal{R}^{n \times p_c}$ denote observed product weights, uncontrollable and controllable factors respectively. We called $D = \{y^o, X_u^o, X_c^o\}$ for the sake of notional simplicity.

Let superscript "F" denote the unobserved future value of the variable. The future may be during the process optimization or after the experiment is conducted. As a result, $y^F \in \mathcal{R}$, $x_u^F \in \mathcal{R}^{p_u}$ and $x_c^F \in \mathcal{R}^{p_c}$ denote product weights, uncontrollable and controllable factors respectively.

Let S denote the operational space of controllable factors. Let α, t be scrap rate and threshold, respectively. The framework of the process optimization problem that minimizes the expected value of future product weight $y^F \in \mathcal{R}$ considering the scrap rate when a controllable factor x_c^F is given can be defined as

$$\min_{x_c^F \in S} E[y^F | x_c^F, D] \tag{2.1}$$

$$s.t.$$

$$P(y^F \leq t | x_c^F, D) \leq \alpha.$$

In this study, we aimed to solve the problem (2.1) in both classical and Bayesian approaches under different assumptions.

The second objective of this study is to perform a static fully Bayesian optimal experimental design. We modified Lindley's approach since our case has two different observable variables. Let d_t denote the duration of the experiment. Let superscript "d" denote the variables observed in training. As a result, $y^d \in \mathcal{R}^{d_t}$ denotes the product weights, and $X_u^d \in \mathcal{R}^{d_t \times p_u}$ denotes the uncontrollable factors that will be observed throughout the experiment. Lastly, $X_c^d \in \mathcal{R}^{d_t \times p_c}$ denotes settled values of controllable factors during the experiment and each row of X_c^d

is identical. Design setting X_c^d will yield X_u^d with respect to model parameters $\theta \in \Theta$ and both will yield y^d with respect to model parameters $\beta \in \mathcal{B}$. Let loss function $L(X_c^d, D)$ represent loss of experiment for a design X_c^d. Then, optimal design X_c^{d*}, which minimizes the loss function, is defined as

$$X_c^{d*} = \arg\min_{X_c^d \in S} \quad L(X_c^d, D). \tag{2.2}$$

where

$$L(X_c^d, D) = E\left[L(y^d, X_u^d, X_c^d, D) \middle| D, X_c^d \right] \tag{2.3}$$

$$= \int_{\mathcal{X}_u^d} \int_{Y^d} L(y^d, X_u^d, X_c^d, D) p(y^d, X_u^d | X_c^d, D) dy^d dX_u^d.$$

Since the experiment is not conducted yet, the expectation of loss function with respect to y^d and X_u^d is taken. Lastly, $L(y^d, X_u^d, X_c^d, D)$ is defined as in (2.4), which directly aims to improve the result of process optimization.

$$L(y^d, X_u^d, X_c^d, D) = \min_{x_c^F \in S} \quad E[y^F | y^d, X_u^d, X_c^d, D, x_c^F] \tag{2.4}$$

$$s.t.$$

$$P\left(y^F \leq t | x_c^F, y^d, X_u^d, X_c^d, D \right) \leq \alpha.$$

Chapter 3

Literature Review

Many studies in the literature focus on the approximation of the response surface, process optimization, robust design, and experimental design.

In the classical approach, Myers et al. [4] studied Response Surface Methodology (RSM) to optimize the process using designed experiments. In general, RSM is applied sequentially. In the RSM approach, firstly, the response surface is approximated by a first-order model. Then, by using an optimization technique called the steepest ascent/descent, the moving direction in the predictor space can be detected to obtain a better target value. It hopefully leads the experimenter to the near-optimal region of predictor space. When no improvement or lack of fit in the first-order models is detected, a second-order model is fit to capture the curvature of the response surface at the near-optimal region. In the RSM approach, response surface approximation, process optimization, and design of experiments are carried out simultaneously. However, in our case, since we were not able to conduct sequential designs, we coped with all parts separately.

The idea of robustness has been deeply studied in terms of being insensitive to model, parameter, and sample variations. However, being robust to uncontrollable factors was firstly introduced by Taguchi. Although Taguchi's approach

was limited and not flexible, robust parameter design was extended in the classical approach in the context of RSM [4]. Also, the idea can be applied easily to the Bayesian context by integrating the predictive density of the response variable with respect to uncontrollable factors. Therefore, the idea can easily be broadened in the Bayesian approach. The common approach in the literature is defining uncontrollable factors as independent from each other and independent from controllable factors as well. However, we assumed that there is a relation between them. This assumption is not delusive since if there is no relation, the uncertainty of predictions of uncontrollable factors will be high. As a result, they will behave like noise.

E. Del Castillo [5] studied process optimization in classical and Bayesian approaches. He stated that in the existence of multiple responses, the accounting correlation between them is not clear in the classical approach. Even if correlations are accounted for in fitting the model, those will be neglected at the optimization step. In the Bayesian approach, correlations among multiple responses can be considered for both fitting the model and optimization. Also, the Bayesian approach provides various objective functions for the optimization step. For example, increasing the probability of staying in a certain space.

Expression in (1.2) generally does not have a closed form. Thus, it requires an approximation of the Utility function for a fixed design point. One of the simple methods is using Monte Carlo integration for its approximation. If the problem is fully Bayesian, in order to use Monte Carlo integration, posterior calculations are required for each possible future data. Depending on the specifications of a prior distribution and likelihood, the posterior could be intractable, and it must be approximated. Some techniques such as Importance Sampling, Markov Chain Monte Carlo (MCMC), Sequential Monte Carlo (SMC), and Laplace Approximation are developed to approximate target posterior distribution. Laplace Approximation is a deterministic approach that relies on the assumption that posterior distribution behaves like a multivariate normal distribution. When priors are informative or applications involve large amounts of data, this assumption could be reasonable [6]. At the same time, the dimension of the problem should be considered since it suffers from the curse of high dimensionality [2]. Importance

Sampling is a popular method when sampling from the target distribution is not possible. A common way is sampling from the prior distribution to approximate the posterior distribution. However, it could be inefficient in the case that the posterior distribution is not close to the prior distribution. [2]. Both Laplace approximation and importance Sampling can be combined in a way that importance distribution is defined as the result of Laplace approximation. Ryan [6] stated that this combined approach still may not be useful in case of high non-normality of posteriors. Markov Chain Monte Carlo (MCMC) methods offer an effective way to draw samples from the posterior distribution. Based on Markovian transition probabilities, the posterior distribution is simulated sequentially. For a large number of iterations, the chain converges to its equilibrium distribution [7]. On the other hand, Sequential Monte Carlo [8], aka particle filter methods, combines importance sampling with Monte Carlo schemes to approximate posterior distribution. It rejuvenates independent initial particles taken from some distribution (e.g., prior) throughout the importance sampling step, resampling step and a Markov kernel. Unlike the MCMC, posterior distributions can be updated iteratively in the presence of new observations.

Even if the experimental design problem is fully Bayesian, depending on the selection of a specific utility function, posterior calculations may not be required. For example, a commonly used utility function is the Kullback-Leiber divergence between prior and posterior distributions. It should be used if someone is interested in maximizing the expected information gained on model parameters. Although this utility function is a functional of the posterior distribution, with the help of Bayes theorem and the absence of terminal decision x_t, the problem can be solved with nested Monte Carlo approximations [9].

Even though the utility function is approximated for a fixed design, a search algorithm is required to travel over the design space to solve (1.2). Müller [10] fitted a smooth curve over the expected utility of simulated design points. After that, a deterministic solution can be exploited to find the optimal design. Huan [9] solved the optimization problem by using nested Monte Carlo approximations with the help of stochastic optimization algorithms, simultaneous perturbation stochastic approximation (SPSA), and Nelder-Mead nonlinear simplex (NMNS).

Bielza [11] proposed an augmented probability model to search over design space by using MCMC simulation. Augmented probability model (x^d, y^d, β) defined as proportional to $U(x^d, \beta, y^d)p(\beta, y^d|x^d)$. Under satisfaction of appropriate conditions by utility function and augmented probability model, a constructed MCMC on (x^d, y^d, β) is sampling designs in areas of high expected utility.

Chapter 4

Process Optimization in Classical Approach

In this chapter, we solved the process optimization model (2.1) in the classical approach by making some assumptions and modifications. First, to find Model (2.1)'s objective function, we first had to approximate the response surface. To approximate it, numerous methods exist such as linear regression models, artificial neural networks, decision tree models, etc. However, since our purpose is not only approximating the response surface but also optimizing the process subject to certain constraints, black-box methods might be cumbersome. Thus, rather than black-box methods, we used linear regression, which made derivative-based optimization techniques available for process optimization since the analytic expression of its prediction function exists, is continuously twice differentiable, and is easy to differentiate. We examined several linear regression models considering both uncontrollable and controllable factors as predictors.

Table 4.1: Summary statistics of the Model (4.1)

r.squared	adj.r.squared	sigma	F.statistic	p.value	df	df.residual	nobs
0.165	0.1595	1.4625	30.2081	0	24	3669	3694

4.1 Response Surface Approximation

4.1.1 Main Effects Model

Let $y_i^o \in \mathcal{R}$ be corresponding elements of y^o for $i = 1, ..., n$. Let $x_{ci}^o \in \mathcal{R}^{p_c}$ and $x_{ui}^o \in \mathcal{R}^{p_u}$ be transpose of corresponding rows of X_c^o and X_u^o, respectively. Lastly, let $x_i^o \in \mathcal{R}^p$ denote $x_i^o = (x_{ci}^{o\top}, x_{ui}^{o\top})^\top$ for ease of notion.

We first built the first-order model, which includes only the main effects of predictors. It is defined as

$$y_i^o = x_i^{o\top} a + \varepsilon_i. \tag{4.1}$$

where $\varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$ for $i = 1, ..., n$ and $a \in \mathcal{R}^p$ is a coefficient vector. Note that since the first elements of x_i^o's are always 1, the model above contains intercept.

Summary statistics related to the fitted Model (4.1) are given in Table 4.1. F-statistic and its p-value indicate a statistical relation between response variables y_i^o's and predictors x_i^o's. R^2 and adjusted R^2 statistics seem too low; they are approximately 0.15. Diagnostic plots are shown in Figure 4.1. The residual vs. fitted plot shows that there is no distinctive pattern between fitted values and residuals. It validates our linearity assumption. Q-Q plot shows that standardized residuals are normally distributed. Scale-location plot shows that their variance is constant throughout the fitted values, which validates our assumption of homoscedasticity. Lastly, the residual vs. leverage plot shows that there is no influential point. Although diagnostic plots are reasonably good, interaction effects are commonly observed in physical systems like ours. Therefore, we continued with a model that considers interaction effects as well.

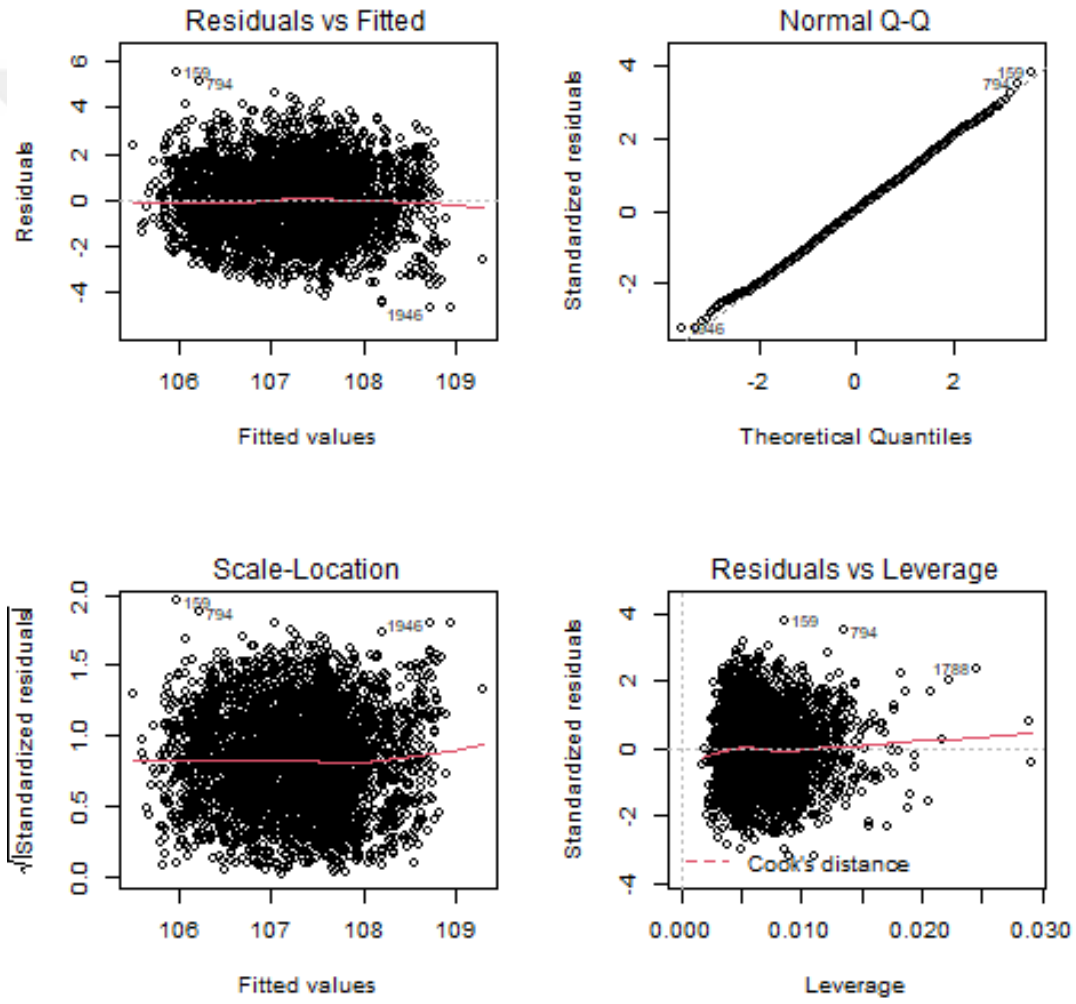

Figure 4.1: Diagnostic plots of the Model (4.1)

Table 4.2: Summary statistics of the Model (4.2)

r.squared	adj.r.squared	sigma	F.statistic	p.value	df	df.residual	nobs
0.4398	0.3903	1.2457	8.8799	0	300	3393	3694

4.1.2 Main and Interaction Effects Model

The second model, which also considers interaction effects, is defined as in Model (4.2). We did not provide the conventional representation of linear models since it might be hard to represent as models' order increases.

$$y_i^o = x_i^{o\top} a + x_i^{o\top} Q x_i^o + \varepsilon_i. \tag{4.2}$$

where $\varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$ for $i = 1, ..., n$ and $Q \in \mathcal{R}^{p\times p}$ is a coefficient matrix whose diagonal elements are zero.

Summary statistics and diagnostic plots are given in Table 4.2 and Figure 4.2, respectively. No red flags show up in diagnostic plots. R^2 and adjusted R^2 increased substantially compared to the main effects model. It concluded that second-order or high-order models might explain the data better.

4.1.3 High Order Model

Having seen the improvement after interaction terms were added to the model, we built a high-order model, which is defined as

$$y_i^o = x_i^{o\top} a + x_i^{o\top} Q x_i^o + x_i^{o\top} diag(x_i^o) T x_i^o + x_i^{o\top} diag(x_i^o) F diag(x_i^o) x_i^o + \varepsilon_i. \tag{4.3}$$

where $T, F \in \mathcal{R}^{p\times p}$ are coefficient matrices whose diagonal elements are zero and $\varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$, $diag(x_i^o) \in \mathcal{R}^{p\times p}$, which is a diagonal matrix of x_i^o, for $i = 1, ..., n$.

Summary statistics and diagnostic plots are shown in Table 4.3 and Figure 4.3 respectively. Adding high-order terms increased R^2 as expected. Also, adjusted R^2 increased, even if it penalizes adding unimportant terms. Diagnostic plots show that the model reasonably satisfies linear regression assumptions. However, there exists a too-slight upward trend in standardized residuals' variance.

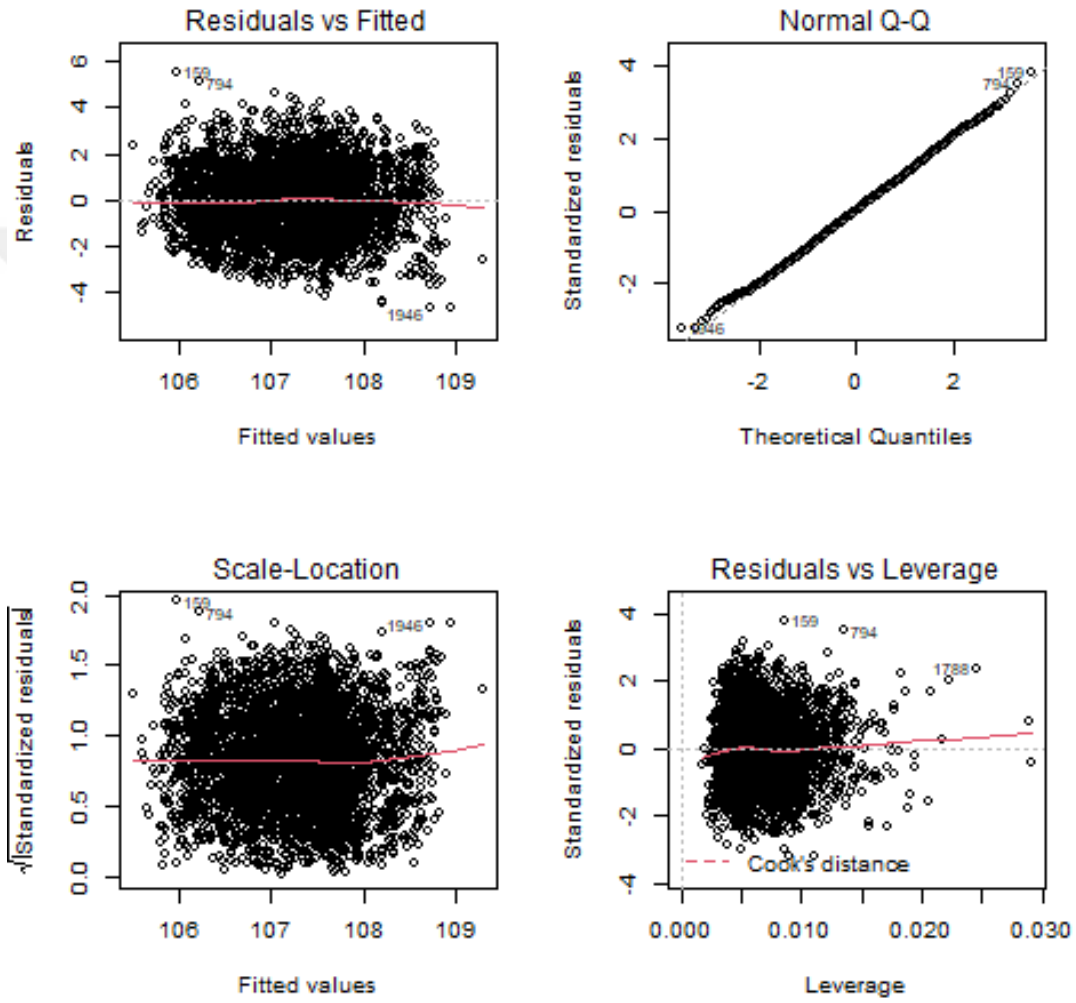

Figure 4.2: Diagnostic plots of the Model (4.2)

Table 4.3: Summary statistics of the Model (4.3)

r.squared	adj.r.squared	sigma	F.statistic	p.value	df	df.residual	nobs
0.7203	0.592	1.019	5.6158	0	1161	2532	3694

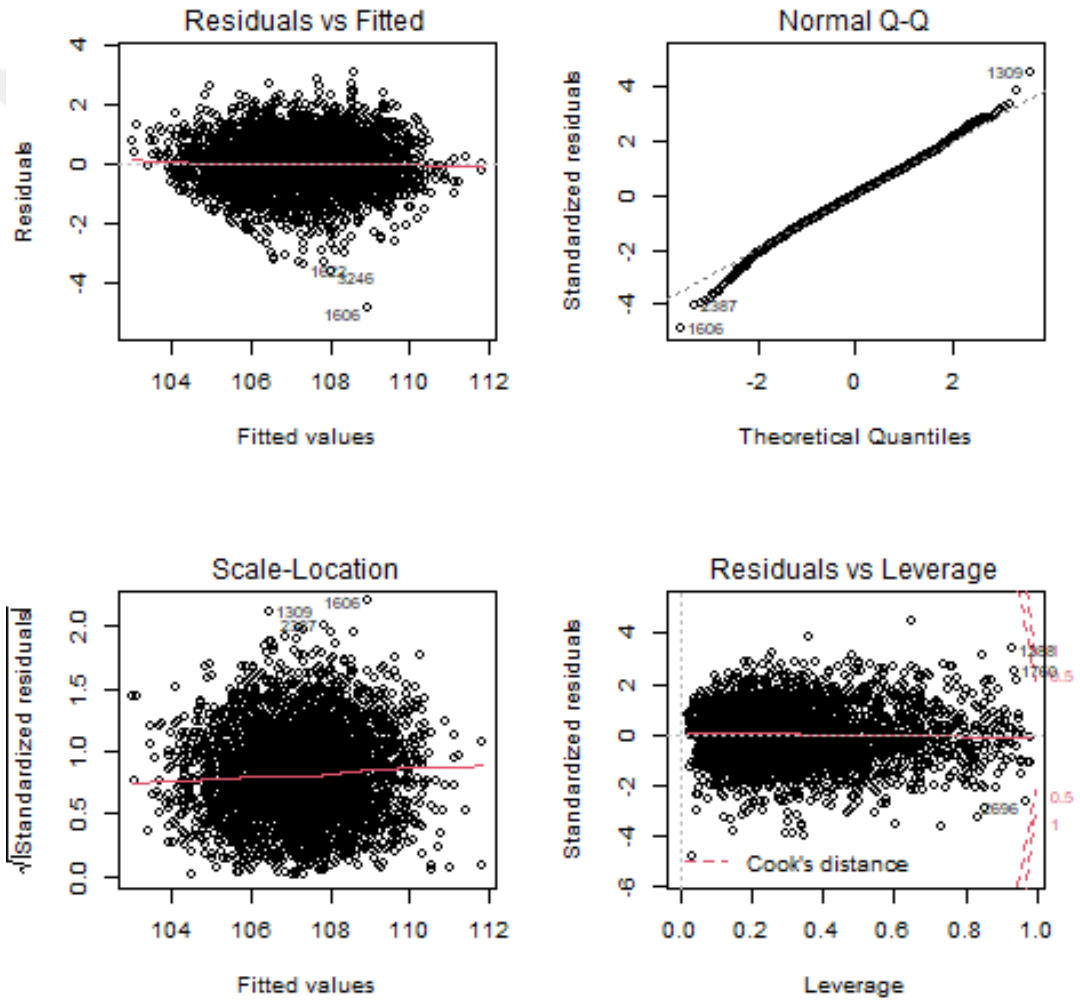

Figure 4.3: Diagnostic plots of the Model (4.3)

4.1.4 Stepwise Selected Model (SSM)

It should be considered that adding more terms to the model may lead it to be overfitted. Therefore, selecting a subset of terms should be considered. Unfortunately, since evaluating every possible subset would be computationally infeasible in our case, we performed the backward stepwise selection on the Model (4.3). It is performed based on the Akaike information criterion (AIC), which is equal to $2p - 2\log(L)$ where p is the number of estimated parameters and L is the maximum likelihood value of the likelihood function of the model. While performing backward stepwise selection, the terms' hierarchy should also be considered. The "step" function in R is suitable for this process. However, it does not take advantage of parallel computing, and it is slow for high-dimensional problems. Thus, we coded our own parallel version of the step function.

Throughout the selection procedure, 424 terms were eliminated. The remaining terms consist of both types of factors, controllable and uncontrollable, and their interactions and high orders. Interaction effect plots of some of the important interactions are given in Figure 4.4. Two plots at the top show interactions between two controllable variables while holding the rest of the variables at their means. The left top plot shows that increasing the value of variable C 5FU either increases or decreases the product weight with respect to the value of C F1A. If C F1A stays either its 50^{th} or 20^{th} percentiles, it leads to an increment of the product weight. However, for 80^{th} percentile, an increment of the value of two factors decreases the product weight. At the top right plot, a change in the value of C F2U has a similar effect on the product weight regardless of the value of C F4A. Left bottom shows the plot of interaction between a controllable factor C FPS and uncontrollable factor U YOS. Depending on the percentile of U YOS, C FPS has different impacts on the product weights. The right bottom plot displays the interaction effect of two uncontrollable factors. When U PD stays at its 80^{th} percentile, increasing U MSHS increases product weight. Otherwise, it may lead to a decrease.

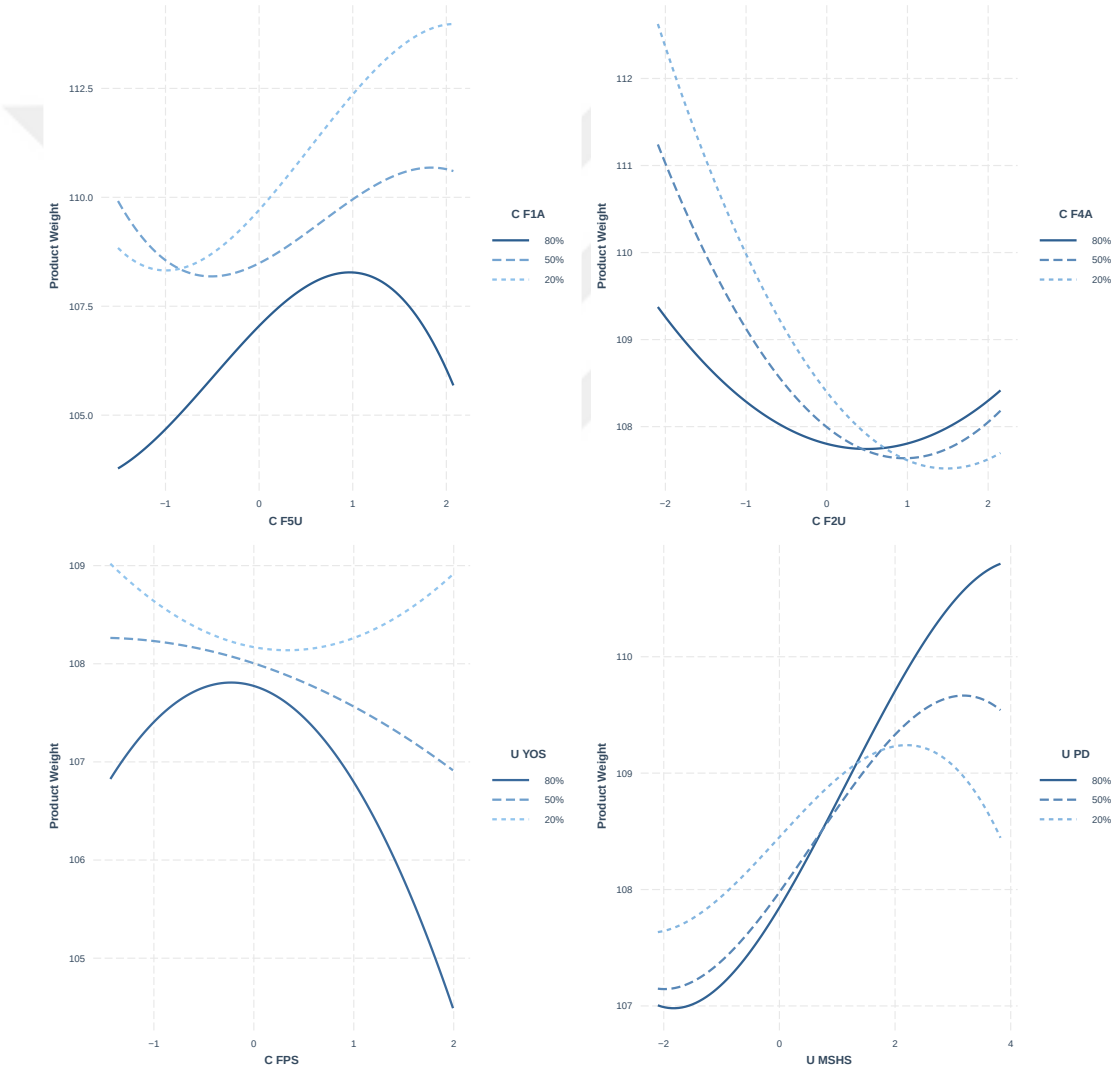

Figure 4.4: Some important interaction effects

Diagnostic plots given in Figure 4.5 are reasonably good except for a too-slight upward trend in standardized residuals' variance. Summary statistics is given in Table 4.4. Its adjusted R^2 statistic is the greatest among the fitted models. However, R^2, adjusted R^2 statistics, and our backward stepwise selection procedure, which considers AIC, do not directly assess the prediction power of our model for unseen data. Therefore, we performed a cross-validation study.

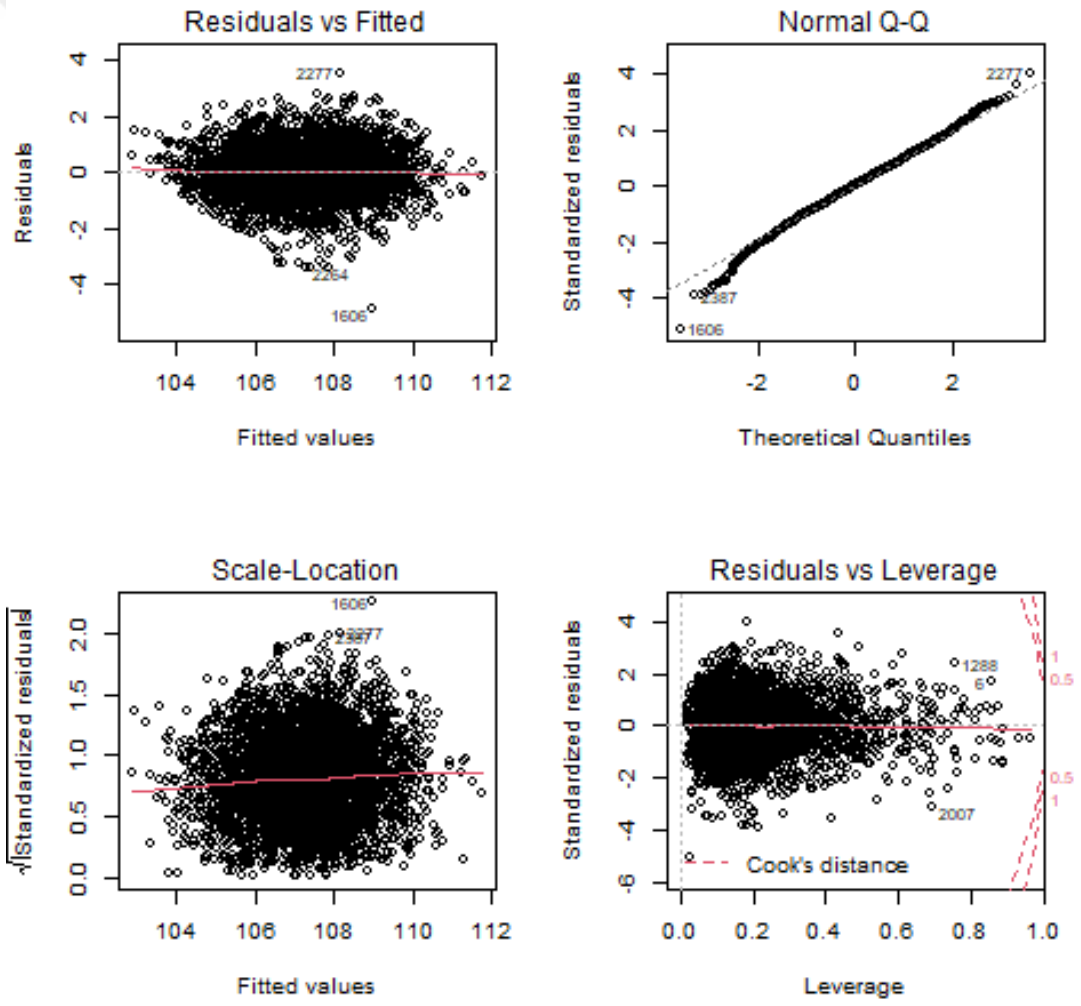

Figure 4.5: Diagnostic plots of the Stepwise Selected Model

Table 4.4: Summary statistics of SSM

r.squared	adj.r.squared	sigma	F.statistic	p.value	df	df.residual	nobs
0.6985	0.6233	0.9792	9.2903	0	737	2956	3694

Table 4.5: Top 10 H2O.ai model results

Model Type	RMSE	MSE	MAE
RF	1.0849	1.1769	0.8334
RF	1.0927	1.1940	0.8366
GBM	1.0955	1.2000	0.8435
GBM	1.1004	1.2110	0.8402
GBM	1.1110	1.2342	0.8559
GBM	1.1118	1.2360	0.8480
GBM	1.1130	1.2387	0.8569
GBM	1.1182	1.2503	0.8635
XGBoost	1.1238	1.2629	0.8672
XGBoost	1.1257	1.2671	0.8693

4.1.5 Benchmarking

It is unknown whether the real relationship between our response variable and predictors is linear, although our diagnostic plots indicate that the assumption is reasonable. Comparing the aforementioned models with the prediction models capable of non-linear modeling relations may also demonstrate how much is lost because of this assumption and how accurately the data can be fitted. To find the benchmarking model, we took advantage of an AI Cloud Platform called H2O.ai [12] that is capable of implementing prediction models such as Random Forest(RF), Gradient Boosting Machine(GBM), XGBoost, Neural Networks, and so on. We fitted 50 different models and evaluated them with 10-fold cross-validation using the "automl" function of H2O.ai. Top 10 models' Mean Squared Errors (MSE), Root Mean Squared Errors (RMSE) and Mean Absolute Errors (MAE) are given in Table 4.5. The model at the top of the Table 4.5 was chosen as benchmarking model since it overperforms based on all metrics.

Table 4.6: 10-fold CV RMSE of models

Fold	Model (4.1)	Model (4.2)	Model (4.3)	SSM	Benchmark
Fold1	1.4519	1.2952	1.5449	1.1690	1.1272
Fold2	1.4648	1.3823	1.7580	1.1754	1.1505
Fold3	1.4261	1.3144	1.5435	1.1755	1.0404
Fold4	1.4798	1.2913	1.2876	1.0904	1.0655
Fold5	1.4031	1.3820	1.4860	1.0993	1.0457
Fold6	1.5398	1.3901	1.4826	1.1773	1.1694
Fold7	1.4947	1.3580	1.6713	1.3045	1.1370
Fold8	1.4797	1.3389	1.4153	1.1595	1.0968
Fold9	1.4543	1.3337	1.5508	1.1530	1.0799
Fold10	1.4980	1.4293	1.5918	1.1461	1.1103
Mean	1.4692	1.3515	1.5332	1.1650	1.1023

Table 4.7: 10-fold CV MAE of models

Fold	Model (4.1)	Model (4.2)	Model (4.3)	SSM	Benchmark
Fold1	1.1756	0.9943	1.1093	0.9023	0.8767
Fold2	1.1628	1.0657	1.1932	0.8803	0.8773
Fold3	1.1315	1.0315	1.1004	0.9020	0.7982
Fold4	1.1659	0.9742	0.9785	0.8536	0.8144
Fold5	1.1222	1.0546	1.1232	0.8704	0.8123
Fold6	1.2371	1.0890	1.1009	0.8941	0.8899
Fold7	1.1899	1.0412	1.1603	0.9856	0.8619
Fold8	1.1884	1.0433	1.0349	0.8882	0.8264
Fold9	1.1805	1.0505	1.1009	0.8923	0.8631
Fold10	1.1876	1.1132	1.0995	0.8826	0.8555
Mean	1.1742	1.0458	1.1001	0.8951	0.8476

Table 4.8: 10-fold CV MSE of models

Fold	Model (4.1)	Model (4.2)	Model (4.3)	SSM	Benchmark
Fold1	2.1079	1.6775	2.3867	1.3666	1.2706
Fold2	2.1458	1.9109	3.0906	1.3817	1.3235
Fold3	2.0339	1.7277	2.3823	1.3818	1.0825
Fold4	2.1897	1.6674	1.6579	1.1890	1.1354
Fold5	1.9687	1.9098	2.2081	1.2085	1.0935
Fold6	2.3710	1.9324	2.1982	1.3861	1.3675
Fold7	2.2340	1.8443	2.7932	1.7018	1.2927
Fold8	2.1895	1.7927	2.0030	1.3444	1.2029
Fold9	2.1150	1.7787	2.4051	1.3293	1.1663
Fold10	2.2440	2.0429	2.5339	1.3136	1.2327
Mean	2.1599	1.8284	2.3659	1.3603	1.2168

4.1.6 10-Fold Cross Validation

We randomly split data once into ten different folds. Models were trained using nine folds, and the remaining fold was predicted. RMSE, MAE and MSE of those predictions is given in Table 4.6, 4.7 and 4.8. Results conclude that adding interaction terms to Model (4.1) increased its predictive capability. However, adding more high-order terms (e.g., Model (4.3)) made the model overfitted. On the other hand, SSM overperforms compared to other fitted linear models. Also, although it is restricted to modeling linear relations, its performance is too close to Benchmarking Model. Therefore, SSM was chosen as a response surface approximator.

4.2 Process Optimization

SSM's terms consist of both controllable and uncontrollable factors. Therefore, in order to solve Model (2.1), we made further assumptions. We assumed that there is a linear relation between controllable factors x_{ci}^o's and uncontrollable factors x_{ui}^o's as

$$x_{ui}^o \sim N(A^\top x_{ci}^o, \Sigma_u). \tag{4.4}$$

where $A \in \mathcal{R}^{p_c \times p_u}$, and $\Sigma_u \in \mathcal{R}^{p_u \times p_u}$ is a positive semi-definite matrix. It can be rewritten as

$$X_u^o = X_c^o A + \mathcal{E}. \tag{4.5}$$

where $\mathcal{E} \in \mathcal{R}^{n \times p_u}$ whose each row is multivariate normal with zero mean and common covariance matrix Σ_u.

Let $\hat{A}$ be ordinary least squares (OLS) estimator of (4.4). Then, one can predict x_u^F by giving x_c^F as

$$E[x_u^F | x_c^F, X_u^o, X_c^o] = \hat{A}^\top x_c^F. \tag{4.6}$$

Let $\hat{a}$,$\hat{Q}$,$\hat{T}$,$\hat{F}$,$\hat{\sigma}^2$ be OLS estimators of SSM's parameters a,Q,T,F,σ^2 respectively. Recall that x^F which denotes all predictors was in the from of controllable and uncontrollable factors such that $x^F = ((x_c^F)^\top, (x_u^F)^\top)^\top$. The objective function of the (2.1) can be written as

$$E[y^F | x^F, D] = (x^F)^\top \hat{a} + (x^F)^\top \hat{Q} x^F + (x^F)^\top diag(x^F) \hat{T} x^F + (x^F)^\top diag(x^F) \hat{F} diag(x^F) x^F. \tag{4.7}$$

Let $g : \mathcal{R}^p \to \mathcal{R}^{p_m}$ be a non-linear mapping representing the SSM's arranged terms where p_m is the number of SSM's terms. Let $H \in \mathcal{R}^{n \times p_m}$ denote SSM's design matrix such that each row is filled with the transpose of corresponding $g(x_i^o)$'s. The constraint of the Model (2.1) can be written as

$$P(y^F \leq t | x^F, D) \leq \alpha$$

$$= P\Big(y^F - E[y^F | x^F, D] \leq t - E[y^F | x^F, D] \Big| x^F, D \Big) \leq \alpha$$

$$= P\left(\frac{y^F - E[y^F | x^F, D]}{\hat{\sigma}\sqrt{1 + g(x^F)^\top (H^\top H)^{-1} g(x^F)}} \leq \frac{t - E[y^F | x^F, D]}{\hat{\sigma}\sqrt{1 + g(x^F)^\top (H^\top H)^{-1} g(x^F)}} \middle| x^F, D \right) \leq \alpha.$$

The inequality above can be modified as in (4.8) due to the fact that the left expression in the probability is distributed t with the degree of freedom $n - p_m$.

$$E[y^F | x^F, D] - t_{1-\alpha, n-p_m} \hat{\sigma} \sqrt{1 + g(x^F)^\top (H^\top H)^{-1} g(x^F)} \geq t. \tag{4.8}$$

Then, final version of the optimization model is given as

$$\min_{x_c^F \in S} (x^F)^\top \hat{a} + (x^F)^\top \hat{Q} x^F + (x^F)^\top diag(x^F)\hat{T}x^F + (x^F)^\top diag(x^F)\hat{F}diag(x^F)x^F. \tag{4.9}$$

$$s.t$$

$$x^F = \begin{bmatrix} x_c^F \\ x_u^F \end{bmatrix}$$

$$E[y^F|x^F, D] - t_{1-\alpha,n-p_m}\hat{\sigma}\sqrt{1 + g(x^F)^\top (H^\top H)^{-1} g(x^F)} \geq t$$

$$x_u^F = \hat{A}^\top x_c^F.$$

4.2.1 Solution of the Optimization Model

Objective function of the Model (4.9) and its constraints are twice continuously differentiable. Therefore, we were able to exploit derivative-based solvers. We numerically calculated the Hessian of the objective function of the optimization model by randomly choosing values for decision variables from the domain of the problem. Hessian of the objective function was not positive semi-definite. It resulted in our problem being non-convex. Also, notice that the problem has non-linear constraints. As a result, a non-linear solution algorithm that can handle non-linear, equality, and inequality constraints was required. We used the Sequential (least-squares) Quadratic Programming (SQP) algorithm to solve the problem. To find a search direction, SQP solves iterative quadratic problems where the original problem's objective function is approximated in quadratic form and constraints are approximated in the first order. The algorithm is already embedded in the R package "nloptr"[13]. We should note that different solvers or approaches in the literature might be more appropriate for our optimization problem. However, we only selected a solver considering the problem type since comparing solvers is not within the scope of this study.

We set α and k as 0.0001 and 98 respectively. We defined operational space S as real numbers between 20^{th} and 80^{th} percentiles of controllable factors. Since the

problem is not convex, it is not guaranteed to find a global minimum. Therefore, we started the algorithm from 100 initial points randomly chosen from convex combinations of the 20^{th} and 80^{th} percentiles. The best solution, values of factors, and their percentiles are given in Table 4.9. The objective value of the solution shows that we can minimize the expected value of the future product's weight to 102.66, which is less than all observed product weights. It can be concluded that the result of the Model (4.9) is satisfying. By implementing these findings, the company may decrease the mean of the product weights by 4.27 percent, reducing the production cost.

Also, we must point out the deficiencies of the Model (4.9). We have modeled the relation between x_{ui}^o's and x_{ci}^o' by building a multivariate linear regression as given in (4.5). Although parameters A and Σ are estimated jointly, their OLS estimators are the same as corresponding OLS estimators of fitted multiple linear regression models for each response variable using all predictors. Therefore, even if elements of x_u^o might be correlated, we could not find those correlations. Secondly, instead of taking into account the prediction uncertainty of the uncontrollable factors, we used their pointwise predictions, which might result in the relaxation of the constraint. Hence, Model (4.9) is not a robust design.

Table 4.9: Process Optimization Results

Variable	Type	Value	Percentile of Value
Product Weight	Objective Value	102.66	0.00
Controllable F1A	optimization	0.64	0.66
Controllable F1U	optimization	-0.86	0.20
Controllable F2A	optimization	-0.24	0.46
Controllable F2U	optimization	0.29	0.60
Controllable F3A	optimization	-0.17	0.53
Controllable F3U	optimization	-0.08	0.51
Controllable F4A	optimization	-0.70	0.20
Controllable F4U	optimization	0.24	0.54
Controllable F5A	optimization	0.58	0.68
Controllable F5U	optimization	1.10	0.80
Controllable FPS	optimization	-0.42	0.39
Controllable RDH	optimization	1.14	0.64
Uncontrollable HMCSS	prediction	-0.01	0.80
Uncontrollable KBO	prediction	0.53	0.66
Uncontrollable MKHS	prediction	0.11	0.53
Uncontrollable MSHS	prediction	0.00	0.52
Uncontrollable PMHC	prediction	-0.28	0.25
Uncontrollable PD	prediction	0.17	0.56
Uncontrollable PK	prediction	0.59	0.58
Uncontrollable POC	prediction	-0.17	0.50
Uncontrollable PR	prediction	0.42	0.64
Uncontrollable HHKSS	prediction	-0.83	0.24
Uncontrollable ROS	prediction	-0.15	0.48
Uncontrollable YOS	prediction	0.20	0.68

Chapter 5

Process Optimization in Bayesian Approach

In this Chapter, we transferred the previous study, which was performed in the classical approach, to the Bayesian context. We directly used terms of SSM. For ease of notation, let $\xi \in \mathcal{R}^{p_m}$ denote latent variables correspond to SSM's design matrix H. Then, the model can be built as

$$y^o|\xi, \sigma^2, H \sim N(H\xi, \sigma^2 I_n). \tag{5.1}$$

Jeffrey defined non-informative prior as a prior which is invariant to parameter transformations. For example, a prior $p(\xi)$ is invariant if the below holds.

$$p(\xi) = p(\phi)\left|\frac{d\phi}{d\xi}\right|.$$

where $\xi = h(\phi)$ is one to one transformation.

Then, Jeffrey showed that a prior that satisfies the equation above is as

$$p(\xi) \propto I(\xi)^{1/2}.$$

where $I(\xi)$ is Fisher information of the parameter ξ.

We used Jeffrey's non-informative priors not only as they are defined as non-informative but also to coincide Bayesian estimates and standard errors with the classical results in the context of multiple linear regression [5]. Jeffrey's non-informative priors for the Model (5.1) are defined as

$$p(\xi, \sigma^2) = p(\xi)p(\sigma^2),$$

$$p(\xi) \propto 1,$$
$$p(\sigma^2) \propto \frac{1}{\sigma^2}.$$

Then, posterior distributions of ξ's and σ^2's are given as

$$p(\xi, \sigma^2|y^o, H) \propto p(y^o|\xi, \sigma^2, H)p(\xi, \sigma^2) \implies \tag{5.2}$$
$$\xi|y^o, H \sim t_{n-p_m}\left(\hat{\xi}, s^2(H^\top H)^{-1}\right),$$
$$\sigma^2|y^o, H \sim Inv\ \chi^2(n - p_m, s^2).$$

where

$$\hat{\xi} = (H^\top H)^{-1}Hy^o,$$
$$s^2 = \frac{(y^o - H\hat{\xi})^\top(y^o - H\hat{\xi})}{n - p_m}.$$

Distribution of future product weight $y^F|h^F, y^o, H$ for a fixed predictor vector $h^F \in \mathcal{R}^{p_m}$ is given as

$$y^F|h^F, y^o, H = \int\int p(y^F|h^F, y^o, H, \xi, \sigma^2)p(\xi, \sigma^2|y^o, H)d\sigma^2 d\xi.$$

It implies

$$y^F|h^F, y^o, H \sim t_{n-p_m}\left((h^F)^\top\hat{\xi}, s^2(1 + (h^F)^\top(H^\top H)^{-1}h^F)\right).$$

Recall that matrix H was mapped by g with x^o_{ci}'s and x^o_{ui}'s. Therefore it can be written for x^F_u and x^F_c as

$$y^F|x^F_c, x^F_u, y^o, H \sim t_{n-p_m}\left(g(x^F_c, x^F_u)^\top\hat{\xi}, s^2(1 + g(x^F_c, x^F_u)^\top(H^\top H)^{-1}g(x^F_c, x^F_u))\right). \tag{5.3}$$

We assumed the relation between uncontrollable and controllable factors is linear, as previously. It was defined as

$$x_{ui}^{o} \sim N(A^{\top} x_{ci}^{o}, \Sigma_u). \tag{5.4}$$

where $A \in \mathcal{R}^{p_c \times p_u}$, and $\Sigma_u \in \mathcal{R}^{p_u \times p_u}$ is a positive semi definite matrix.

We used Jeffrey's non-informative priors for A and Σ_u, as shown below.

$$p(A, \Sigma_u) \propto p(A)p(\Sigma_u) \quad \text{where } p(A) \propto 1 \text{ and } p(\Sigma_u) \propto \frac{1}{|\Sigma_u|^{\frac{p_u+1}{2}}}. \tag{5.5}$$

Then, the distribution of the future value of $x_u^F | x_c^F, X_u^o, X_c^o$ given a predictor vector x_c^F is given as

$$x_u^F | x_c^F, X_u^o, X_c^o \sim t_{n-p_c-p_u-1}\left(\hat{A}^{\top} x_c^F, M^{-1}\right). \tag{5.6}$$

where

$$\hat{A} = (X_c^{o\top} X_c^o)^{-1} X_c^{o\top} X_u^o,$$

$$M = \frac{(n - p_c - p_u - 1)[(X_u^o - X_c^o \hat{A})^{\top})(X_u^o - X_c^o \hat{A})]^{-1}}{1 + (x_c^F)^{\top}(X_c^{o\top} X_c^o)^{-1} x_c^F}.$$

A multivariate t distribution $t_v(\mu, \Sigma)$ converges to multivariate normal distribution $N(\mu, \Sigma)$ as $v \to \infty$. Similarly, a t distribution $t_v(b, \sigma^2)$ converges to normal distribution $N(b, \sigma^2)$. Therefore, expression in (5.3) and (5.6) can be approximated as shown in (5.7) and (5.8), respectively, due to high degree of freedoms.

$$y^F | x_c^F, x_u^F, y^o, H \sim N\left(g(x_c^F, x_u^F)^{\top}\hat{\xi}, s^2(1 + g(x_c^F, x_u^F)^{\top}(H^{\top}H)^{-1} g(x_c^F, x_u^F))\right). \tag{5.7}$$

$$x_u^F | x_c^F, X_u^o, X_c^o \sim N\left(\hat{A}^{\top} x_c^F, M^{-1}\right). \tag{5.8}$$

Derivations of the expressions regarding posterior predictive distributions and posterior distributions of parameters are found in [5].

5.1 Optimization Model

The Model (2.1) can be written as

$$\min_{x_c^F \in S} E_{x_u^F}\left[g(x_c^F, x_u^F)^\top \hat{\xi} | D\right] \tag{5.9}$$

$$E_{x_u^F}\left[P(y^F \leq t | x_c^F, D, x_u^F) | x_c^F, D\right] \leq \alpha.$$

where its objective function immediately comes from the power property as shown in (5.10) and constraint comes from as shown in (5.11).

$$E_{y^F}\left[y^F | x_c, D\right] = E_{x_u^F}\left[E[y^F | x_c^F, D, x_u^F]\right] = E_{x_u^F}\left[g(x_c^F, x_u^F)^\top \hat{\xi} | D\right]. \tag{5.10}$$

$$P(y^F \leq t | x_c^F, D) = \int_\infty^t \int_{x_u^F} p(y^F | x_c, x_u, D) p(x_u | x_c, D) dx_u^F dy \tag{5.11}$$

$$= \int_{x_u^F} P(y^F \leq t | x_c^F, x_u^F, D) p(x_u^F | x_c^F, D) dx_u^F = E_{x_u^F}[P(y^F \leq t | x_c^F, D, x_u^F) | x_c^F, D].$$

5.2 Solution of the Optimization Model

The objective function of the Model (5.9) has a nice form since the expectation is linearly separable. Also, mapping g maps elements of the x_u^F to its moments. Since the moments of multivariate distribution exist, the objective function has a closed form.

Let x_{uj}^F and μ_j denote j^{th} element of x_u^F and $\hat{A}^\top x_c^F$ for $j = 1, ..., p_u$, respectively. Let m_{jk} denote corresponding element of M^{-1} for $j, k = 1, ..., p_u$. Required moments of x_u^F are given as

$$E\left[x_{uj}^F x_{uk}^F\right] = \mu_j \mu_k + m_{jk},$$

$$E\left[(x_{uj}^F)^2 x_{uk}^F\right] = \mu_k m_{jj} + 2\mu_j m_{jk} + \mu_j^2 \mu_k,$$

$$E\left[(x_{uj}^F)^2 (x_{uk}^F)^2\right] = \mu_k^2 m_{jj} + \mu_j^2 m_{kk} + 4\mu_k \mu_j m_{jk} + \mu_j^2 \mu_k^2 + m_{jj} m_{kk} + 2(m_{jk})^2.$$

Derivation algorithm of the expressions above is given in [14].

The constraint of the model does not have a closed form since it contains an error function that must be computed using numerical methods. To solve the problem, we followed the naive approach which is approximating the second constraint by Monte Carlo integration, then solving the problem deterministically. If one can simulate the $Z \in \mathcal{R}^{N_b \times p_u}$ matrix one time, each element is an independent standard normal variable, and can also simulate x_u^F for a fixed x_c^F as

$$^{(i)}x_u^F = \hat{A}^\top x_c^F + L^{(i)} z \quad \text{for} \quad i = 1..., N_b.$$

where $L^\top L$ is Cholesky decomposition of M^{-1} and $^{(i)}z$ is the transpose of Z's i^{th} row. After that, the constraint can be approximated as

$$E_{x_u^F}\left[P(y^F \leq t | x_c^F, D, x_u^F) | x_c^F, D\right] \approx \frac{1}{N_b} \sum_{i=1}^{N_b} P(y^F \leq t | x_c^F, D, {}^{(i)}x_u^F).$$

We set t,α,S as usual and N_b as 10^4. We used the SQP algorithm as we did previously. We started the algorithm from 100 random initial points. Results are given in Table 5.1. The minimum expected value of the future product weight is found as 106.73. Notice that in this model, we took into account correlations and uncertainty of the uncontrollable variables. Therefore, it is a robust design. The objective value of this model is high compared to the previous models' objective value, which was 102.66. It concludes that the uncertainties and correlations of the uncontrollable factors should not be neglected. Furthermore, we must also note that independent probabilistic models for y^o and X_u^o are built and independently combined. As a result, parameters A, Σ_u were taken as independent from y^o and parameters ξ, σ^2 were taken independent from X_u^o. Also, it resulted in independent posterior distributions for both models. However, posterior distributions of both models might be dependent, which may result in our approach being faulty. A decent approach would be defining a Bayesian hierarchical model, which is explained in the next chapter.

Table 5.1: Process optimization results

Variable	Value	Percentile of Value
Product Weight	106.73	0.37
Controllable F1A	-0.22	0.57
Controllable F1U	-0.06	0.49
Controllable F2A	-0.76	0.29
Controllable F2U	-0.51	0.36
Controllable F3A	-0.26	0.51
Controllable F3U	-0.11	0.50
Controllable F4A	-0.38	0.35
Controllable F4U	-0.31	0.32
Controllable F5A	-0.69	0.31
Controllable F5U	-0.76	0.33
Controllable FPS	0.51	0.46
Controllable RDH	0.40	0.57

Chapter 6

Bayesian Hierarchical Models

The built probabilistic models so far do not include timeline information of the data, even if the production process is observed with respect to time. Therefore, adding time information to the probabilistic model may fit better. No pooling phenomenon, which defines independent parameters for each group, may consider group information (e.g., time). However, since our observed data spans only seven months or 76 days, groups may not overlap at the time of experimenting. Therefore, we benefited from partial pooling while building the probabilistic model. In partial pooling, each grouped parameter is fitted by corresponding response variables while they are pooling from a common parameter. Partial pooling can be easily implemented by employing Bayesian hierarchical models. However, the posterior distribution of Bayesian hierarchical models is generally not recognizable and tractable due to their complexity. In those cases, posterior distributions must be approximated or simulated.

6.1 Markov Chain Monte Carlo Algorithms

The general idea behind the Markov Chain Monte Carlo (MCMC) is drawing values from parameter β to approximate distributions and then correcting those

values to approximate posterior distribution $p(\beta|y)$ by formulating a Markov chain on the parameter space. One of the well-known MCMC algorithms is the Metropolis-Hasting algorithm. The algorithm starts with an initial value of β^0 and updates it iteratively in a way that the last drawn value might remain or be updated by a new draw. For example, with having last value β^{t-1}, a candidate value β^* is sampled from a jumping distribution $J_t(\beta^*|\beta^{t-1})$ at t^{th} iteration. This jumping distribution, called proposal distribution, can be selected as any distribution. After that, an acceptance ratio r is calculated as

$$r = \frac{p(\beta^*|y)/J_t(\beta^*|\beta^{t-1})}{p(\beta^{t-1}|y)/J_t(\beta^{t-1}|\beta^*)}.$$

The new value for β^t is set as

$$\beta^t = \begin{cases} \beta^* & \text{with probability } \min(r, 1), \\ \beta^{t-1} & \text{otherwise.} \end{cases} \tag{6.1}$$

Depending on the value of r, jumping may be realized or not. Realization can be guaranteed by defining a special jumping distribution. Let β_j denote a sub-vector of parameter β and β_{-j} denote rest of the parameter. At each iteration t, every sub-vector of β will be updated consecutively. The special jumping distribution $J_{t,j}(\beta^*|\beta^{t-1})$ for j^{th} sub-vector at iteration t is defined as

$$J_{t,j}(\beta^*|\beta^{t-1}) = \begin{cases} p(\beta_j^*|\beta_{-j}^{t-1}, y) & \text{if } \beta_{-j}^* = \beta_{-j}^{t-1} \\ 0 & \text{otherwise} \end{cases} \tag{6.2}$$

This special jumping distribution ensures that jumping is realized when $\beta_{-j}^* = \beta_{-j}^{t-1}$ and it implies a guaranteed jump due to

$$\begin{aligned} r &= \frac{p(\beta^*|y)/J_{t,j}(\beta^*|\beta^{t-1})}{p(\beta^{t-1}|y)/J_{t,j}(\beta^{t-1}|\beta^*)} \\ &= \frac{p(\beta^*|y)/p(\beta_j^*|\beta_{-j}^{t-1}, y)}{p(\beta^{t-1}|y)/p(\beta_j^{t-1}|\beta_{-j}^{t-1}, y)} \\ &= \frac{p(\beta_j^{t-1}|\beta_{-j}^{t-1}, y)}{p(\beta_j^{t-1}|\beta_{-j}^{t-1}, y)} \\ &= 1. \end{aligned} \tag{6.3}$$

This special case is called Gibbs sampling. It is easy to implement when full conditional distribution $p(\beta_j^*|\beta_{-j}^{t-1}, y)$ is available for each j^{th} sub-vector.

MCMC algorithms require some warm-up iterations to start sampling from the posterior distribution. Commonly, half of the simulations are burned. Also, m number of parallel chains, each having a different initial start, are run to benefit from parallel computation and check whether each chain is sampling from the same target distribution. Convergence of chains can be checked by calculation $\hat{R}$ statistic, which considers estimations of between chain and within chain variances [7]. If chains are not mixed well, $\hat{R}$ is greater than 1. Otherwise, it is close to 1. Furthermore, it must be pointed out that, inherently, MCMC draws can be correlated. Therefore, before making an inference about unknown parameter β, the effective sample size (ESS) of simulation draws must be estimated. It briefly represents the estimated number of independent draws and affects the accuracy of the inference. Higher ESS values might be needed depending on the desired accuracy of inference.

6.2 Bayesian Hierarchical Model I

Let subscript "$_g$" represent the group of the variable. We defined variables of the Bayesian Hierarchical Model I as

$$
\begin{gathered}
y_g^o | X_g^o, \sigma^2 \sim N(X_g^o \beta_g, \sigma^2 I_{n_g}) \quad \text{for} \quad g = 1, .., G \\
\beta_g | \bar{\beta}, \Sigma \sim N(\bar{\beta}, \Sigma) \quad \text{for} \quad g = 1, .., G \\
p(\bar{\beta}) \propto 1 \\
\sigma^2 | \frac{\tau}{2}, \frac{\lambda}{2} \sim IG(\frac{\tau}{2}, \frac{\lambda}{2}) \\
\Sigma | \nu, V \sim IW(\nu + p - 1, V) \\
\Sigma_u | \nu_u, V_u \sim IW(\nu_u + p_u - 1, V_u) \\
vec(A) | \Sigma_u, C, \bar{A} \sim N(vec(\bar{A}), \Sigma_u \otimes C^{-1}) \\
vec(X_{ug}^o) | X_{cg}^o, A, \Sigma_u \sim N(vec(X_{cg}^o A), \Sigma_u \otimes I_{n_g}) \quad \text{for} \quad g = 1, .., G \\
p(\bar{A}) \propto 1
\end{gathered}
\tag{6.4}
$$

where

- $y_g^o \in \mathcal{R}^{n_g}$ denotes product weights of group g,
- $\beta_g \in \mathcal{R}^p$ denotes group level effect of group g,
- $\bar{\beta} \in \mathcal{R}^p$ pooling for β_g's,
- $\sigma^2 \in \mathcal{R}$ denotes population dispersion,
- $\Sigma \in \mathcal{R}^{p \times p}$ is covariance matrix of group level effects,
- $X_{ug}^o \in \mathcal{R}^{n \times p_u}$ denotes uncontrollable variables of group g,
- $X_{cg}^o \in \mathcal{R}^{n \times p_c}$ denotes controllable variables of group g,
- $X_g^o \in \mathcal{R}^{n_g \times p}$ represents $[X_{cg}^o X_{ug}^o]$ for $g = 1, ..., G$,
- A denotes coefficients of uncontrollable variables,
- $\bar{A}$ denotes mean of A,
- Σ_u denotes dispersion of uncontrollable variables,
- $V \in \mathcal{R}^{p \times p}, V_u \in \mathcal{R}^{p_u \times p_u}, C \in \mathcal{R}^{p_c \times p_c}, \nu \in \mathcal{R}, \nu_u \in \mathcal{R}, \tau \in \mathcal{R}, \lambda \in \mathcal{R}$ denote hyperparameters.

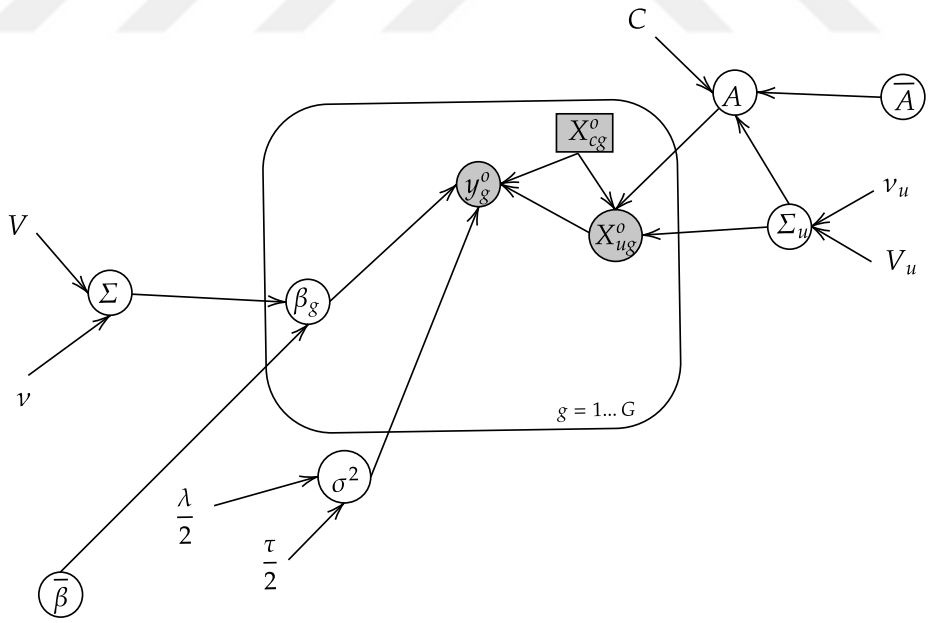

Figure 6.1: DAG of Bayesian Hierarchical Model I

Directed Acyclic Graph (DAG), which shows the relations between variables, is given in Figure 6.1. Round nodes represent random variables. Square nodes

represent deterministic variables. Filled nodes denote observable variables. Notice that β which we defined before as set of latent parameters of y includes parameters $\beta_g, \sigma^2, \bar{\beta}$ and Σ. On the other hand, previously defined θ contains $A, \bar{A}, \Sigma_u$. Distributions of the variables were selected to be proper for Gibbs Sampling. Full conditional distributions of variables can be found in Appendix A. Notice that, in this approach, we could not employ SSM's variables since we could only consider the main effects because of the exponential increment of the number of parameters.

We set hyper-parameters V, V_u and C as identity matrices. Also, ν and ν_u were taken as 2, which leads $p+1, p_u+1$ degrees of freedom for Σ and Σ_u respectively. Thus, each correlation element of covariance matrices Σ and Σ_u has a uniform marginal distribution [7]. We set τ as 2.02 and λ as 4.02 to ensure that mean of the prior distribution of σ^2 is one and its variance is 100. τ and λ can be set to low values (e.g., 0.001) to ensure informativeness. However, it must be considered that as $\tau, \lambda \to 0$, it leads to an improper posterior distribution. Also, depending on the data, it could be informative too [15]. Hence, we set informative prior according to our belief about σ^2.

6.2.1 Monthly Synthetic Data Simulation

Synthetic data with known true values of parameters can be simulated either for debugging the code of the Gibbs sampler or verifying full conditional distributions. Also, it enables to check of statistical procedures and model fits as advertised [16]. We simulated monthly grouped synthetic data, which is desired to be close to real data. The generation process can be found in Appendix B. Figure 6.2 shows histogram plots of real and synthetic data.

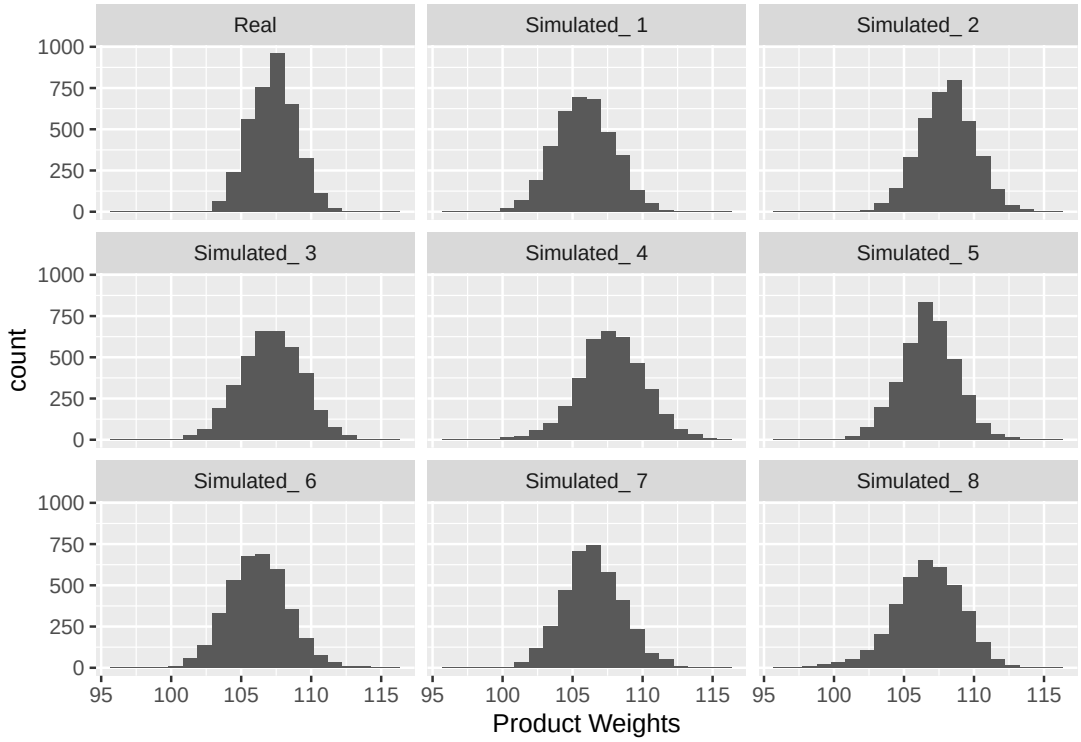

Figure 6.2: Simulated data and real data

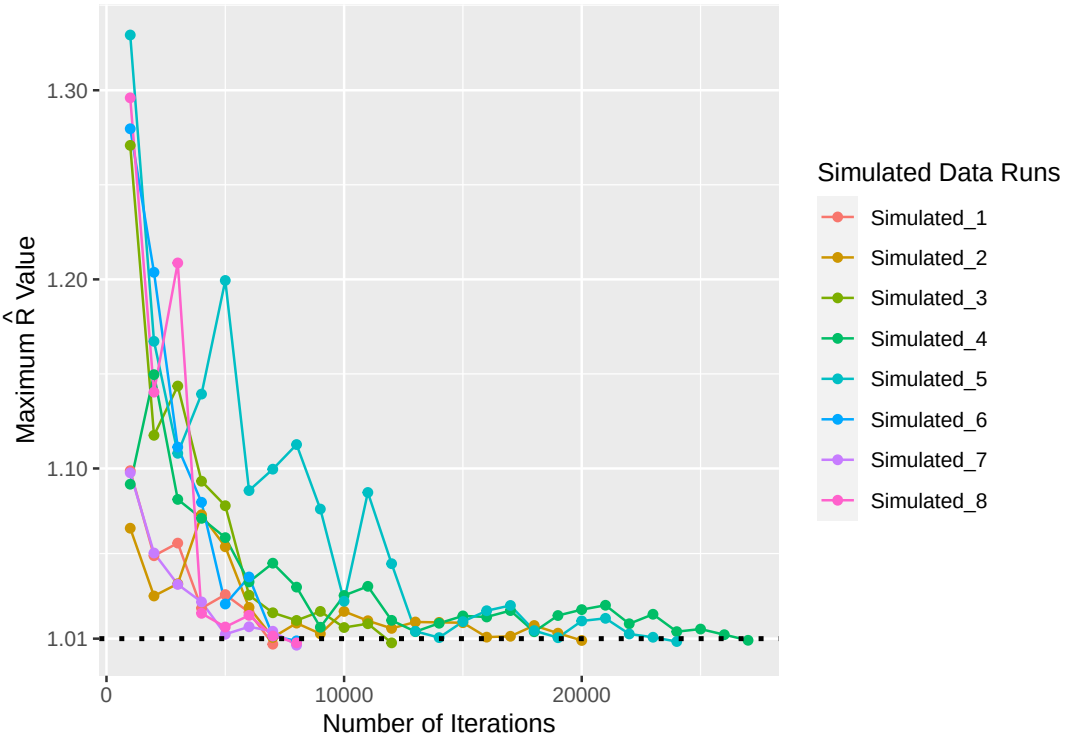

Figure 6.3: Maximum $\hat{R}$ values of runs

6.2.2 Gibbs Sampling on Synthetic Data

We ran four parallel chains, each starting from a random initial point. We burned half of the iterations and checked the convergence status of the Gibbs sampler via $\hat{R}$ by calculating it at each 1000 iteration. It was calculated after each chain was split into two parts to be sure chains mixed well, as Gelman [7] suggested. Although Gelman [7] recommended 1.1 $\hat{R}$ value to ensure convergence, we continued each run till the maximum value of $\hat{R}$ of its parameters was less than 1.01. Maximum $\hat{R}$ values of each run is given in Figure 6.3. Since true values of parameters are known, we calculated p-values as

$$\text{p-value} = \frac{2}{N_{iter}} \min\left(\sum_{i=1}^{N_{iter}} I_{(\beta^i \leq \beta^{true})}, \sum_{i=1}^{N_{iter}} I_{(\beta^i \geq \beta^{true})} \right).$$

where N_{iter} is total draws and β^i denotes i^{th} draw of a parameter β whose true value is β^{true}.

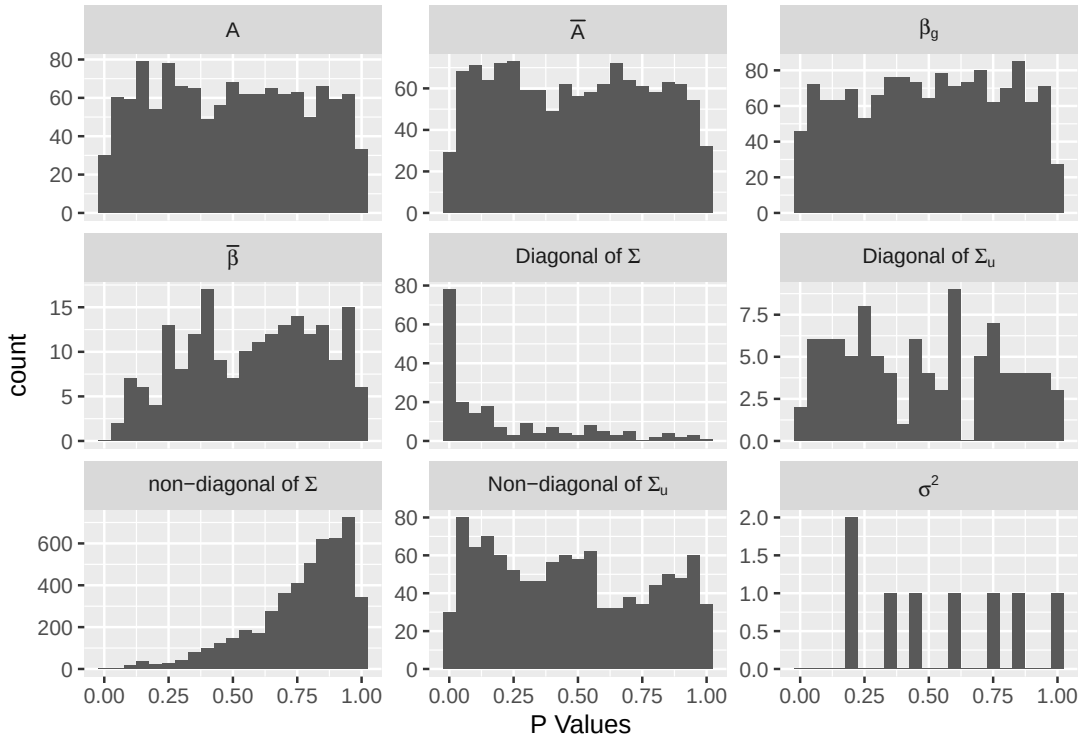

Figure 6.4: Histogram of parameters' p-values

P-values are expected to be distributed uniformly. We merged all runs and exchangeable parameters to see their distribution. Histograms of p-values are given in Figure 6.4. P values of A,$\bar{A}$,β_g and σ^2 seem uniformly distributed. P-values of parameters $\bar{\beta}$, diagonal and non-diagonal of Σ_u are reasonably good. However, p-values of Σ's elements seem far from a uniform distribution, particularly its diagonal elements. Posterior samples regarding Σ do not cover their true values.

There exist several downsides to modeling covariance matrix by using Inverse-Wishart distribution. Firstly, in our hyper-parameters setting, the marginal distributions of variances are distributed as $IG(1, 1/2)$, which has extremely low density near zero. Secondly, if true variances are low compared to the prior mean, posteriors of variances are biased toward larger values, and correlations are biased to zero [17]. Thirdly, each variation parameter is controlled by a single degree of freedom parameter. The information that comes from β_g might not allow Σ to dominate prior distribution. Interestingly, we did not observe the same behavior for parameters of Σ_u. It can be rooted in the fact that the number of β_g's are few, and Σ stays at a deeper level compared to Σ_u.

6.3 Bayesian Hierarchical Model II

Gelman [7] recommended the Scaled-inverse Wishart distribution instead of the Inverse-Wishart distribution. In that case, it breaks our full conditional state, and we had to mix Gibbs sampling with the Metropolis Hasting algorithm. Therefore, we used Hierarchical Half-t prior, which was stated as unproblematic by Alvarez [17] and proposed by Huang [18]. We only updated Σ parameter of Bayesian Hierarchical Model I. Updated parameters are shown below, and the DAG of Bayesian Hierarchical Model II is given in Figure 6.5. Full conditional distributions regarding this update are given in Appendix C.

$$\Sigma|\nu, V_a \sim IW(\nu + p - 1, 2\nu V_a).$$

where $V_a = diag(1/a_1, ...1/a_p)$ and $a_k \sim IG(1/2, 1/A_k^2) \quad \text{for} \quad k = 1, ..., p.$

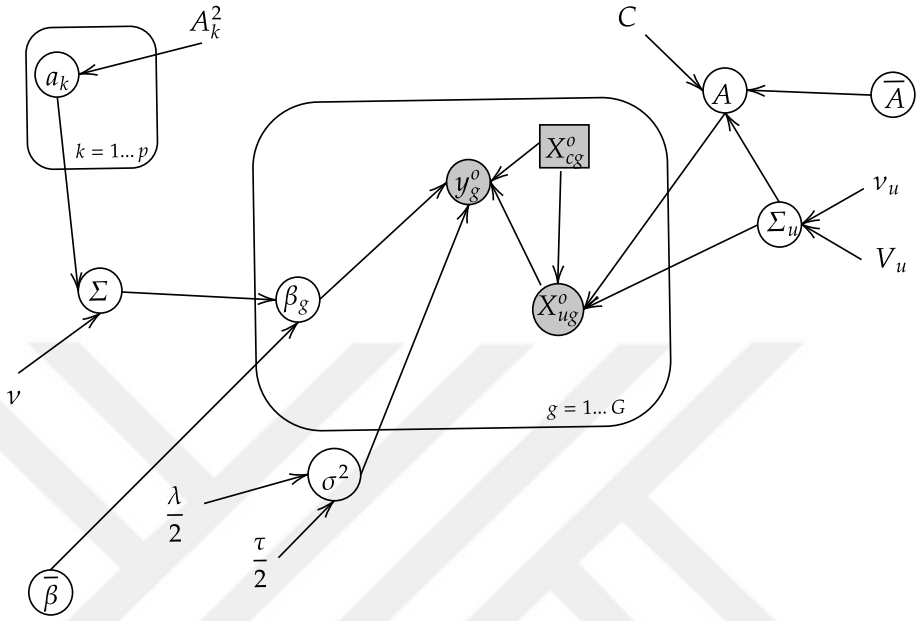

Figure 6.5: DAG of Bayesian Hierarchical Model II

We set ν as 2 to imply marginal uniform distribution on correlations. Also, each A_k is set to 1.04 to coincide the median of the marginal distribution of variances with the median of $IG(1, 1/2)$. Other hyperparameters settings remained the same.

6.3.1 Gibbs Sampling on Synthetic Data

Notice that the update we did on the Bayesian Hierarchical Model I does not affect the part of it related to X_u^o. Therefore, we ran four chains in parallel considering only the y^o part of the model, burned the first half of it, and checked $\hat{R}$ at each 1000 iteration. Maximum $\hat{R}$ values of runs are shown in Figure 6.6. Chains converged too slowly compared to the previous model. This might indicate that posterior distributions became more correlated. Figure 6.7 displays p-values of the updated model. P-values of Σ's diagonal elements are distributed more uniformly than in the previous model. Regarding correlations, there still exists a

concentration toward the right. However, it is less compared to the previous one.

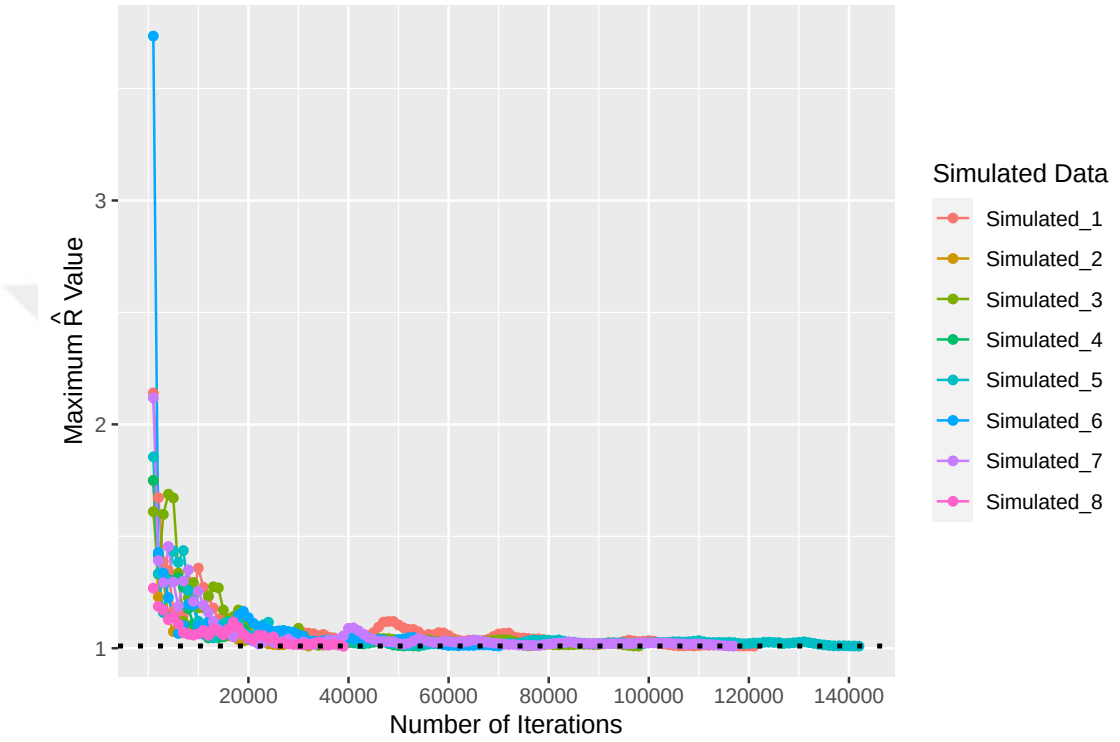

Figure 6.6: Maximum $\hat{R}$ values of runs

6.4 Gibbs Sampling on Monthly Grouped Real Data

Four parallel chains were run for both models on real data till 20000^{th} iteration was reached. The first half of the chains were burned as usual. Maximum $\hat{R}$ values of both models are given in Figure 6.8. Bayesian Hierarchical Model I converged faster than Model II, as we observed on synthetic data runs. Also, runs on real data converged faster than synthetic data runs. This might be caused due to our synthetic data generation process. We assigned uniform distributions on correlations of covariance matrices by using LKJ prior. As a result, it might lead to synthetic data being more correlated than real data. Figure 6.9 shows the

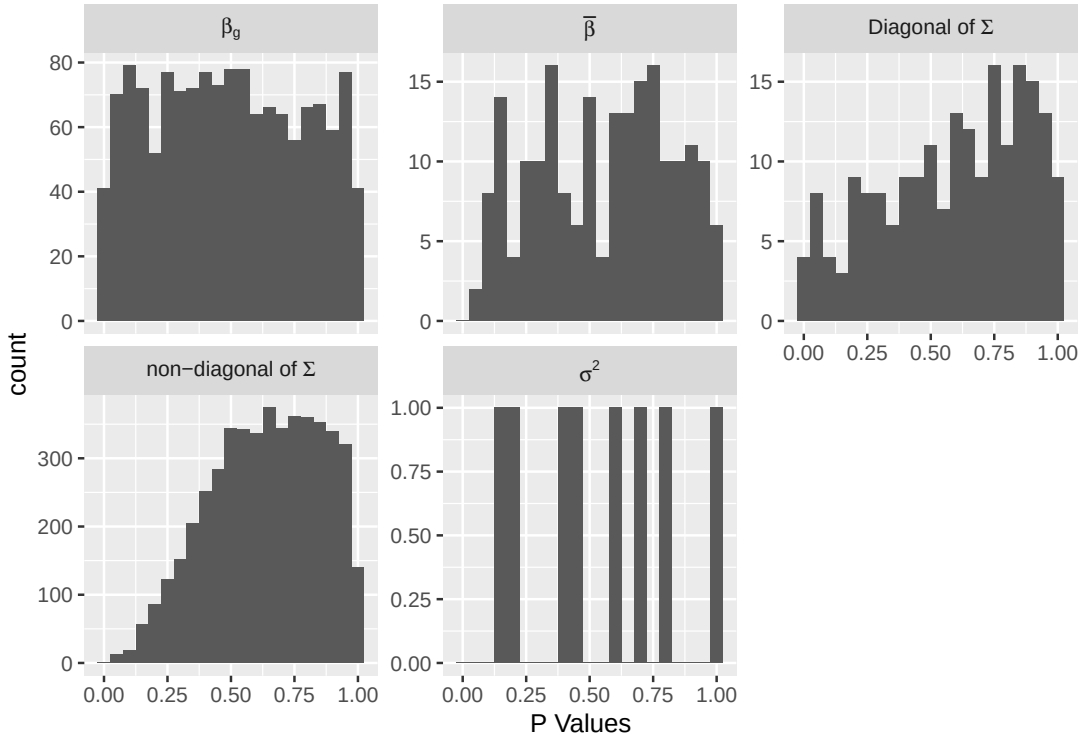

Figure 6.7: Histogram of parameters' p-values

minimum ESS of both models. Bayesian Hierarchical Model I can more effectively sample compared to Model II. However, the ESS ratios of both models are too low. Since we ran four parallel chains, we sampled 40000 values for each parameter. Having roughly 500 or 600 independent samples based on 40000 samples might not be efficient.

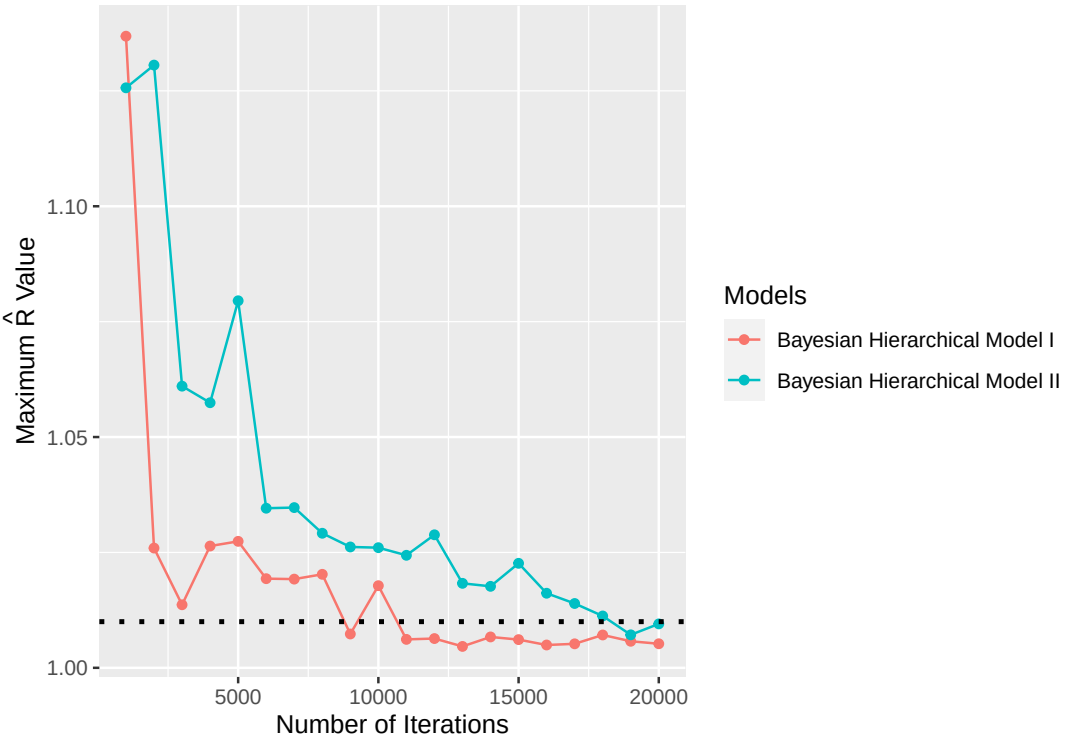

Figure 6.8: Maximum $\hat{R}$ values of real data runs

6.4.1 Posterior Predictive Checking

If the models fit well with the data, replicated data using the same predictors from posterior predictive distributions are expected to look similar to real data. Distribution of replicated data y_{rep} is given as

$$p(y_{rep}|y^o, X^o) = \int p(y_{rep}|\beta, X^o)p(\beta|y^o, X^o)d\beta.$$

where X^o is the same predictor set.

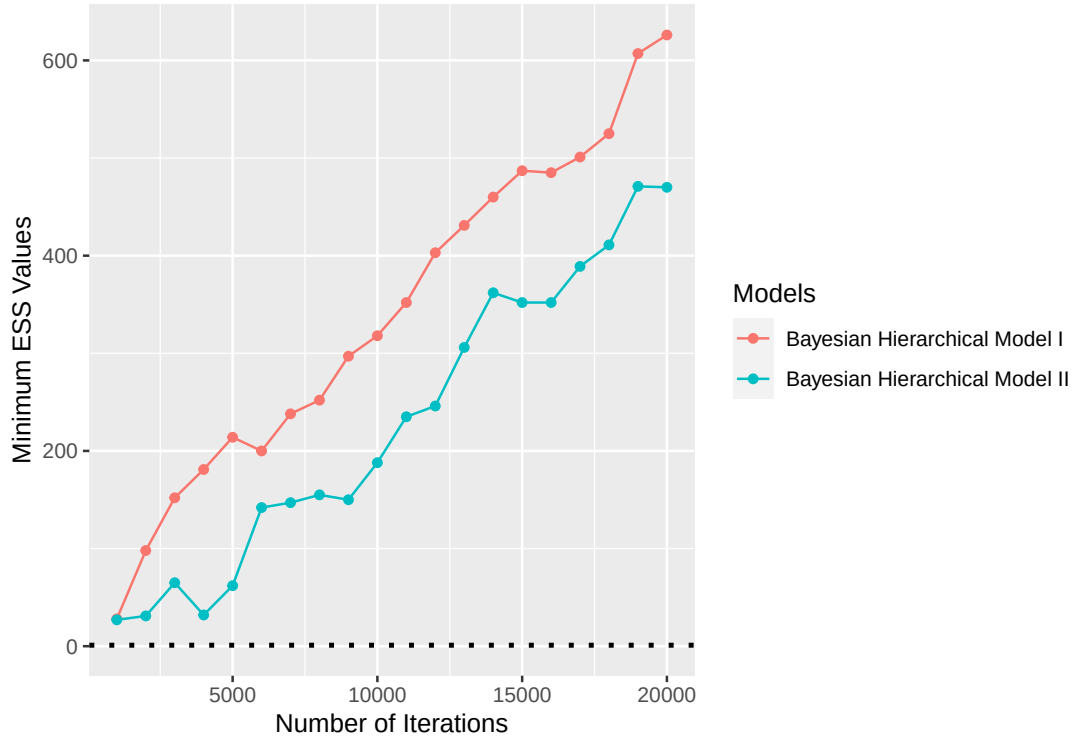

Figure 6.9: Minimum ESS values of real data runs

Since we have already obtained draws of unknown parameter β by running MCMC, replicated data y_{rep} can be generated as

$$^{(i)}y_{rep} \sim p(^{(i)}y_{rep}|^{(i)}\beta, X^o).$$

where $^{(i)}\beta$ is i^{th} draw comes from MCMC run.

Also, some test quantities can be measured to determine the discrepancy between the model and data by using y_{rep}. We generated 1000 replicated data for both models' y^o parts. In Figure 6.10, a posterior predictive comparison of the mean test quantity of two models is given. The vertical black line indicates the mean of y^o, and the histogram is plotted using each replicated data's mean. It shows that both models are capable of fitting the mean of real data. Similarly, in Figure 6.11 and 6.12, we compared their 5^{th} and 95^{th} percentiles. Both plots indicate that for tail test quantities, both models fit well. We conducted the same procedure for X_u^o. Although we modeled X_u^o under the name of different models, notice that their updates in both models are the same. Therefore, we did not compare models, and we only checked their test quantities. Figure 6.13 displays that our model is capable of fitting mean values. However, in terms of tail test quantities, as can be seen in Figure 6.14 and 6.15, model poorly fit to data. It might have resulted from our assumption that there is a linear relation between controllable and uncontrollable factors. When examining the synthetic data runs, we did not observe any poor fitting signs. However, it must be noted that we generated synthetic data by sticking to the linear relation assumption.

6.4.2 Model Comparison

Leave-one-out (LOO), a special case of k-fold cross-validation where k is equal to the number of samples, can be used to estimate a Bayesian model's pointwise out-of-sample prediction accuracy. Another alternative approach is widely applicable information criterion (WAIC), which is asymptotically similar to LOO. Both methods use log-likelihood evaluated at posterior simulation draws of parameters. Fast and robust estimation methods of both methods were proposed by Vehtari, Gelman and Gabry [19]. Those methods are already embedded in

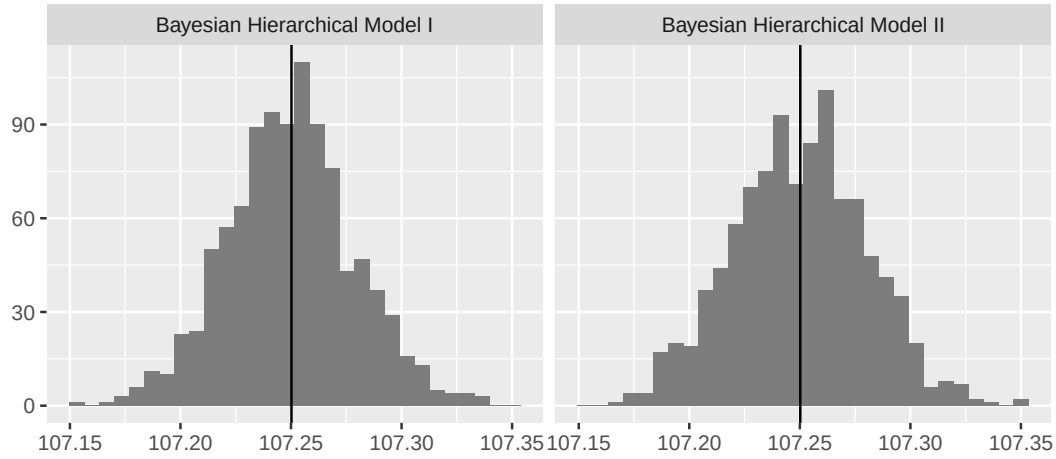

Figure 6.10: Posterior predictive comparison on means

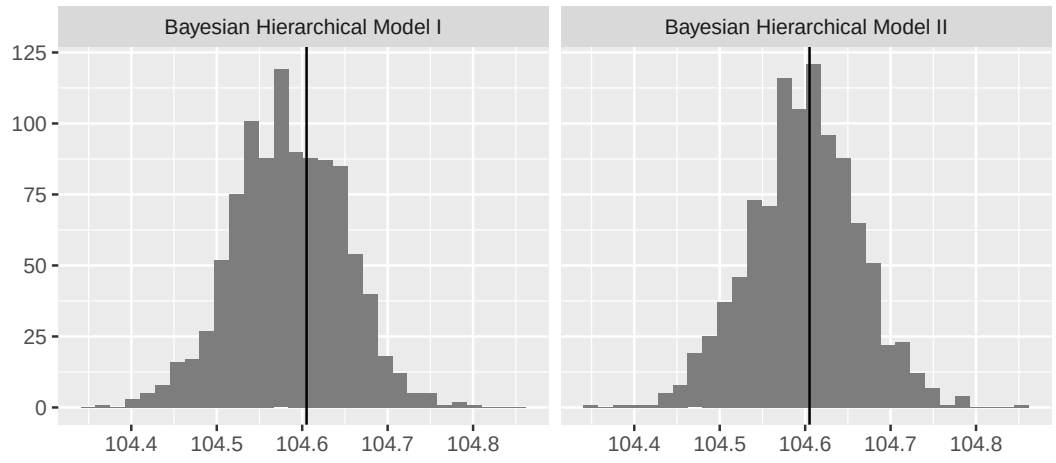

Figure 6.11: Posterior predictive comparison on percentile .05

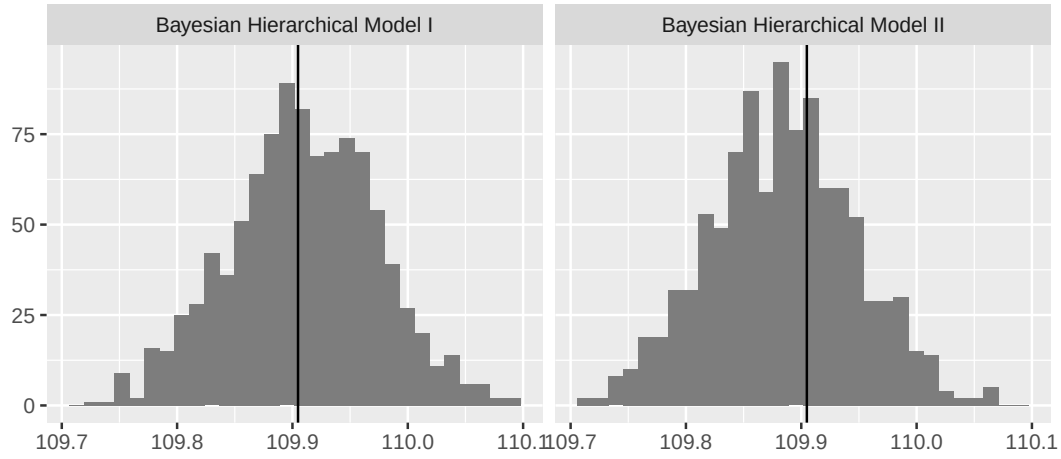

Figure 6.12: Posterior predictive comparison on percentile .95

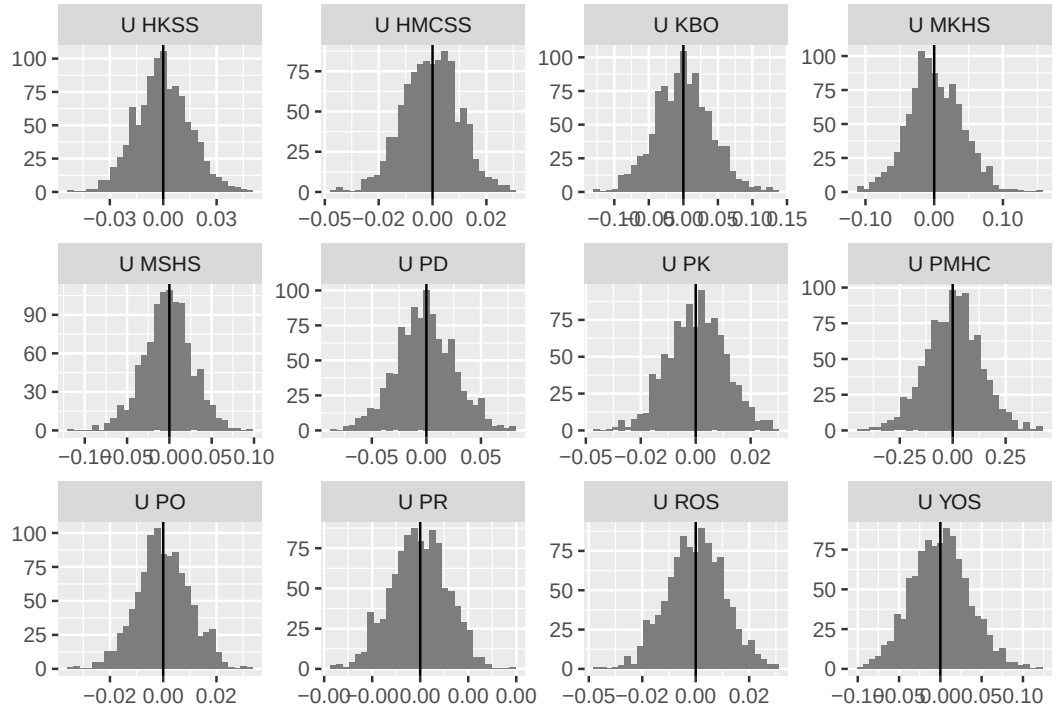

Figure 6.13: Posterior predictive checking on means

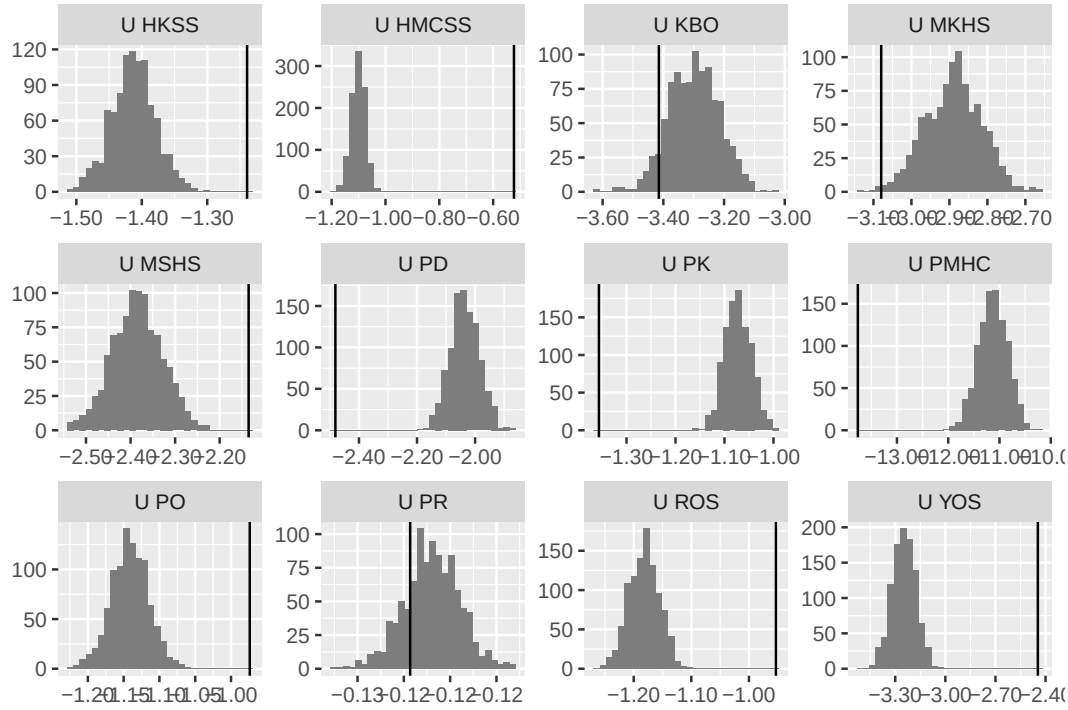

Figure 6.14: Posterior predictive checking on percentile .05

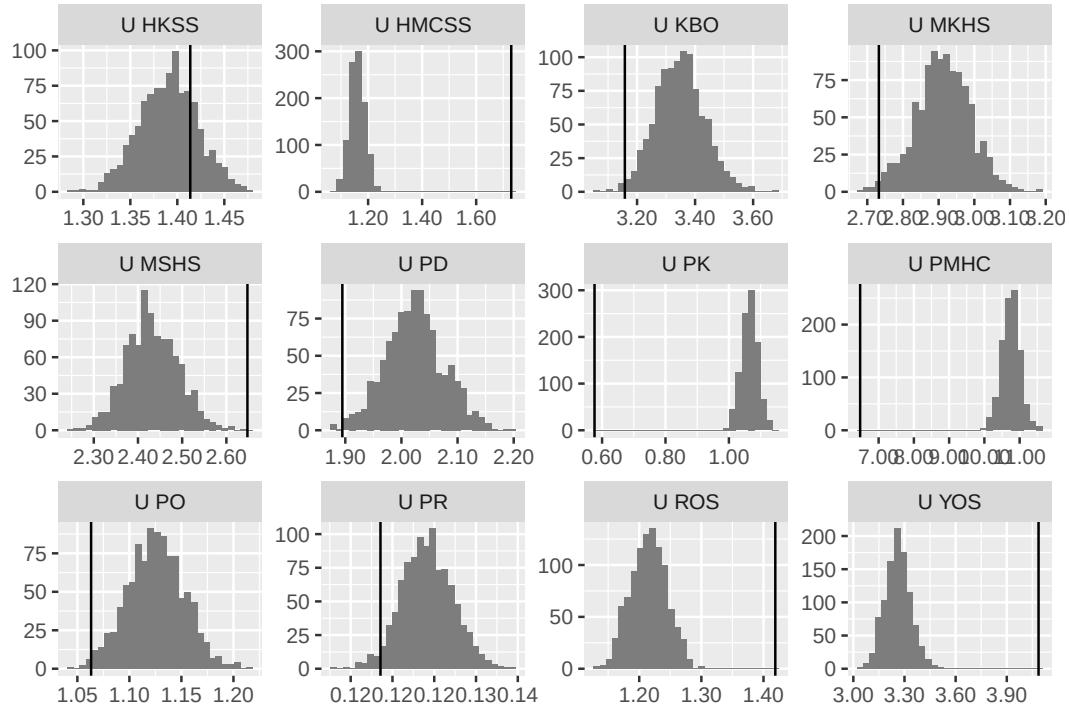

Figure 6.15: Posterior predictive checking on percentile .95

Table 6.1: LOO and WAIC of both models

Model	LOO	WAIC
Bayesian Hierarchical Model I	-6264.66	-6263.24
Bayesian Hierarchical Model II	-6261.26	-6260.29

Table 6.2: LOO comparison of models

Model	Difference	Standard error of difference
Bayesian Hierarchical Model II	0.0	0.00
Bayesian Hierarchical Model I	-3.4	5.41

the R package "loo"[20]. We compared Bayesian Hierarchical Model I and II in terms of only y^o since we do not have different models for X_u^o. LOO and WAIC values are given in Table 6.1. It shows that Bayesian Hierarchical Model II overperforms Model I since a lower value is better for both metrics. However, since both metrics are estimated, the standard error of comparison should also be considered. LOO and WAIC comparison of models are given in Table 6.2 and 6.3. Both comparisons show that the difference between both metrics is insufficient to conclude that Bayesian Hierarchical Model II is statistically better since the standard error of pairwise comparison differences is larger than the differences. However, considering its performance on synthetic data runs, we selected it as a final model.

Table 6.3: WAIC comparison of models

Model	Difference	Standard error of difference
Bayesian Hierarchical Model II	0.00	0.00
Bayesian Hierarchical Model I	-2.95	5.38

Chapter 7

Fully Bayesian Optimal Experimental Design Problem

In this chapter, we examined the fully Bayesian optimal experimental design problem, which was defined in (2.3). The solution to the problem does not have a closed form if it is considered under Bayesian Hierarchical Model II. Thus, numerical approximations or stochastic methods are required to solve it. We discussed solution methods and proposed an update to the hierarchical model.

7.1 Solution Approach and Discussion

For a fixed design X_c^d, loss of the experiment $L(X_c^d, D)$ was defined as

$$L(X_c^d, D) = \int_{\mathcal{X}_u^d} \int_{Y^d} L(y^d, X_u^d, X_c^d, D) p(y^d, X_u^d | X_c^d, D) dy^d dX_u^d.$$

If one is able to sample from $p(y^d, X_u^d | X_c^d, D)$, the expression above can be approximated by Monte Carlo integration as

$$L(X_c^d, D) \approx \frac{1}{N_{out}} \sum_{i=1}^{N_{out}} L({}^{(i)}y^d, {}^{(i)}X_u^d, X_c^d, D).$$

where $\{^{(i)}y^d, ^{(i)}X_u^d\}_{i=1}^{N_{out}}$ denote the samples from $p(y^d, X_u^d|X_c^d, D)$ for $i = 1, ..., N_{out}$.

An alternative approach is separating $p(y^d, X_u^d|X_c^d, D)$ into $p(y^d|X_u^d, X_c^d, D)$ and $p(X_u^d|X_c^d, D)$ and then sampling from $p(X_u^d|X_c^d, D)$. Let $\{^{(i)}X_u^d\}_{i=1}^{N_{out}}$ denote samples from $p(X_u^d|X_c^d, D)$. In addition to that, in this case, we have to find $p(^{(i)}y^d|^{(i)}X_u^d, X_c^d, D)$ which is defined below as

$$p(^{(i)}y^d|^{(i)}X_u^d, X_c^d, D) = \int_{\mathcal{B}} p(y^d|^{(i)}X_u^d, X_c^d, D, \beta)p(\beta|X_c^d, D)d\beta.$$

In the literature, it is common to assume that design X_c^d and β are independent. If it is assumed, sampling $\{^{(j)}\beta\}_{j=1}^{N_{in}}$ from $p(\beta|D)$ is enough to approximate $p(^{(i)}y^d|^{(i)}X_u^d, X_c^d, D)$ as

$$p(^{(i)}y^d|^{(i)}X_u^d, X_c^d, D) \approx \frac{1}{N_{in}} \sum_{j=1}^{N_{in}} p(y^d|^{(i)}X_u^d, X_c^d, D, ^{(j)}\beta).$$

However, we are not able to directly sample from $p(X_u^d|X_c^d, D)$ without considering y^d, since X_u^d is connected to y^d in our model. Therefore, $p(X_u^d|X_c^d, D)$ must be approximated by another Monte Carlo integration as

$$p(X_u^d|X_c^d, D) \approx \frac{1}{N_{in}} \sum_{j=1}^{N_{in}} p(^{(i)}X_u^d|X_c^d, D, ^{(j)}\theta).$$

where $^{(j)}\theta$'s are MCMC draws of $p(\theta|D)$ under the assumption of unknown parameters are independent from design X_c^d.

As readers may notice, in this approach, many approximations are required. Also, posteriors of β and θ are taken as independent from each other and design, which might be a wrong approach if they are dependent. As a result, we decided to sample directly from $p(y^d, X_u^d|X_c^d, D)$ by updating Bayesian Hierarchical Model II. It is assumed that the duration of the experiment covers only the next month G+1, and it will last as d_t. Model II's full conditional distributions can be updated easily by increasing the observation number from n to $n + d_t$ and increasing the number of groups from G to $G + 1$. DAG of updated model is given in Figure 7.1.

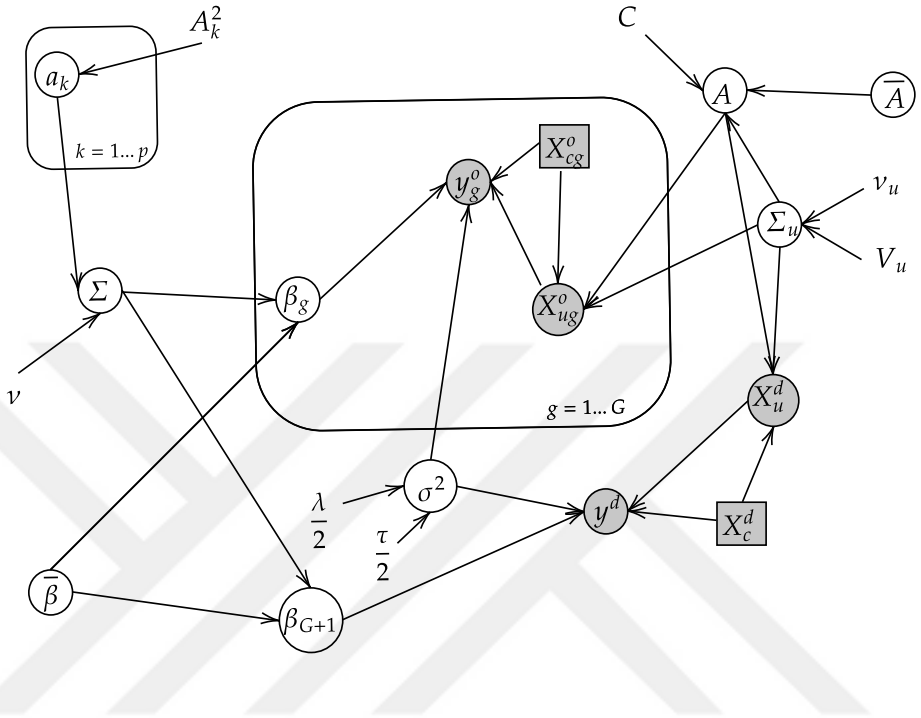

Figure 7.1: DAG of Bayesian Hierarchical Model II with design

Even if we approximate the loss function for a fixed X_c^d, a searching method is needed to travel over the operational space S to find X_c^{d*}. We testified whether we could sample when X_c^d is chosen from different locations of S. We determined d_t as 96 to set the duration of the experiment as one day. We specified four different design settings X_c^d. Each controllable element of ${}^{(1)}X_c^d$ and ${}^{(2)}X_c^d$ were chosen randomly from either its 20^{th} or 80^{th} percentile. On the other hand, ${}^{(mean)}X_c^d$ and ${}^{(median)}X_c^d$ are filled with their means and medians, respectively. Runs were continued until either converge is met or 50000^{th} iteration is reached. Maximum $\hat{R}$ values of each design are given in Figure 7.2. It shows that when design X_c^d is chosen from the edge of operational space S, Gibbs samplers do not converge in a reasonable time. Also, Figure 7.3 displays that we cannot effectively sample in those design settings. If we examine those problematic design settings in terms of y^d and X_u^d, as can be seen in Figure 7.4 and 7.5, the reason for low ESS and high $\hat{R}$ is due to y^d. Otherwise, we might have used X_u^d and y^d draws by assuming those reached the equilibrium.

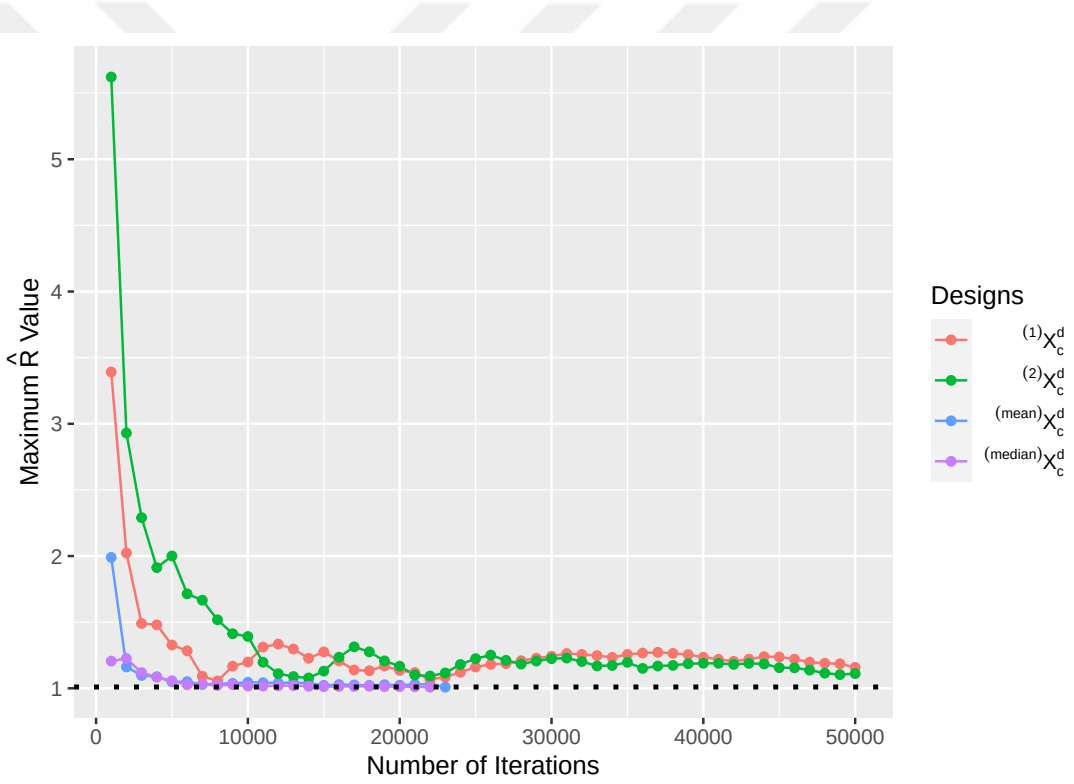

Figure 7.2: Maximum $\hat{R}$ values of different design settings

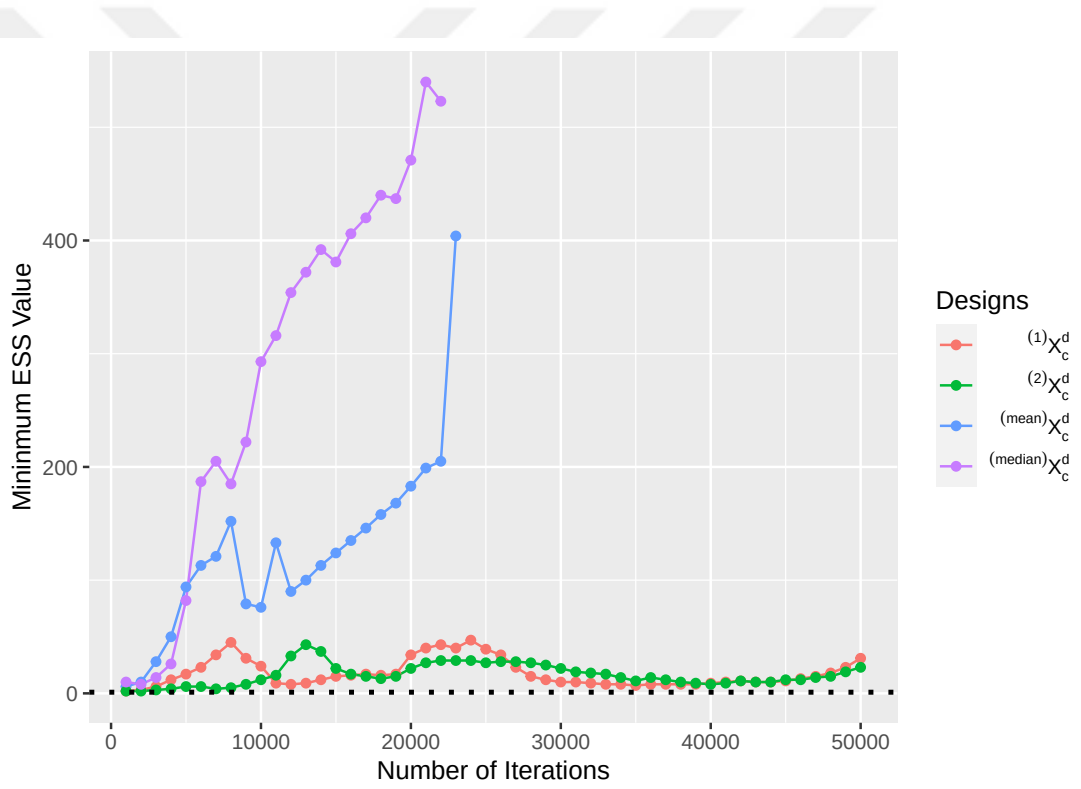

Figure 7.3: Minimum ESS values of different design settings

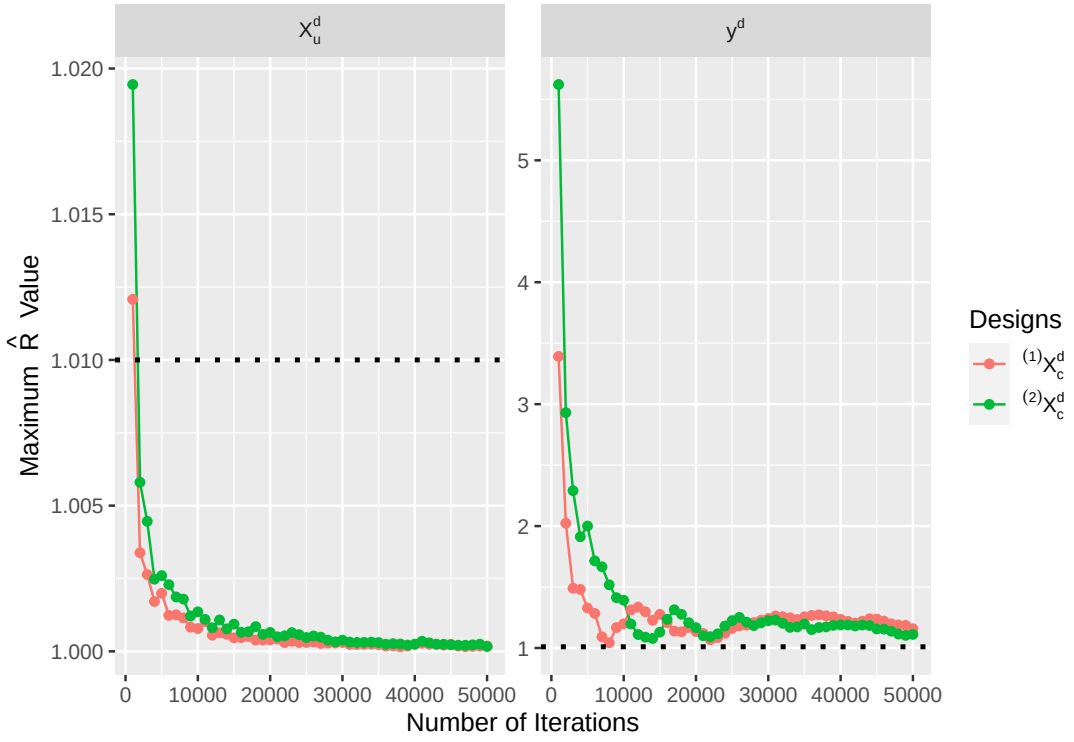

Figure 7.4: Maximum $\hat{R}$ values of X_u^d and y^d

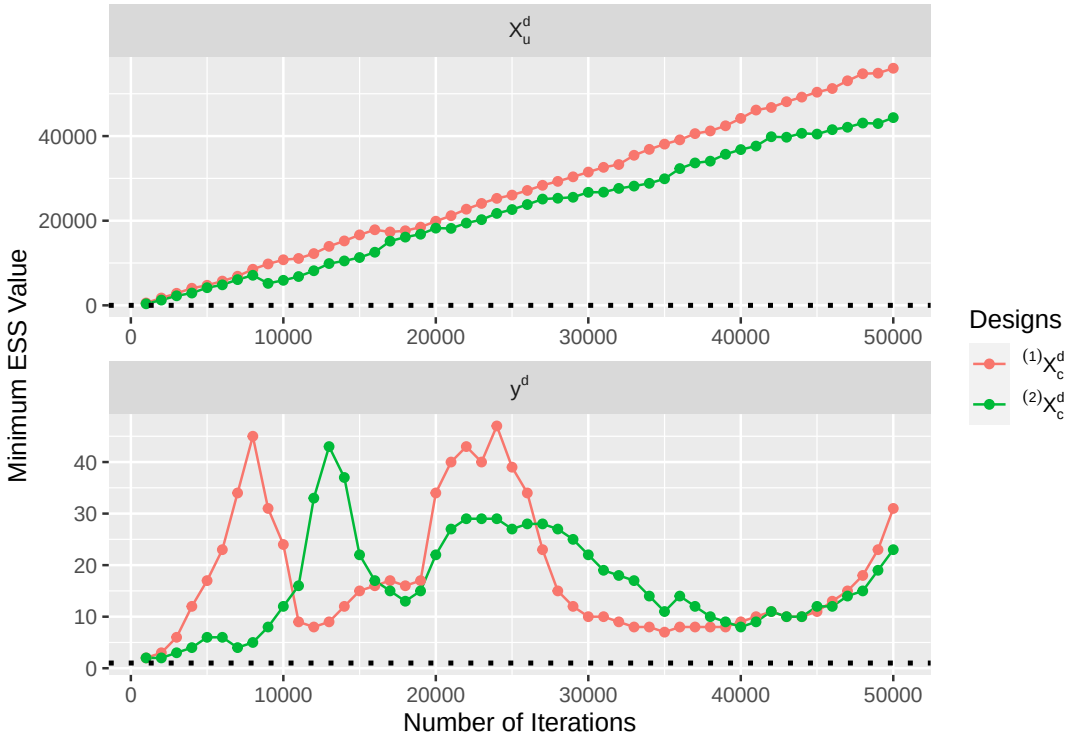

Figure 7.5: Minimum ESS values of X_u^d and y^d

Even though those sampling procedures were realized, $L({}^{(i)}y^d, {}^{(i)}X_u^d, X_c^d, D)$ would have to be solved for each realization of ${}^{(i)}y^d, {}^{(i)}X_u^d$ pair. Its objective function and constraint are extended as

$$E[y^F|{}^{(i)}y^d, {}^{(i)}X_u^d, X_c^d, D, x_c^F] = \int_{Y^F} y^F p(y^F|{}^{(i)}y^d, {}^{(i)}X_u^d, X_c^d, D, x_c^F)dy^F$$

$$= \int_{Y^F}\int_{x_u^F} y^F p(y^F, x^F|{}^{(i)}y^d, {}^{(i)}X_u^d, X_c^d, D, x_c^F)dx_u^F dy^F,$$

$$P\Big(y^F \leq t|{}^{(i)}y^d, {}^{(i)}X_u^d, X_c^d, D, x_c^F\Big) = \int_{-\infty}^{t} p(y^F|{}^{(i)}y^d, {}^{(i)}X_u^d, X_c^d, D, x_c^F)dy^F$$

$$= \int_{-\infty}^{t}\int_{x_u^F} p(y^F, x^F|{}^{(i)}y^d, {}^{(i)}X_u^d, X_c^d, D, x_c^F)dx_u^F dy^F.$$

which led us to a similar problem as we encountered while approximating $L(X_c^d, D)$.

In conclusion, we could not solve our specified experimental design problem. Having two different probabilistic models forced us to build hierarchical models. As it is said before, the case of building independent models for each observable variable and independently combining them might have been a wrong approach. Although there is a more advanced MCMC method, the Hamiltonian Monte Carlo algorithm, embedded in Stan[21], we were unable to take advantage of it. Since we have multiple responses (X_u^d, X_u^o) with a high number of observations, it required more than one day to get a sample, and it was supposed to be rerun for each X_c^d. In addition, to solve our complex loss function, we had to approximate those samples for each realization of y^d and X_u^d as well.

Chapter 8

Conclusion

In this study, we provided a linear response approximator with a close performance compared to more advanced prediction models for predicting product weights. Since uncontrollable factors are also correlated with controllable factors, an additional probabilistic model is built. Using them, we minimized product weights that are statistically related to uncontrollable and controllable factors by considering the process optimization problem in both classical and Bayesian approaches under different assumptions. In the classical approach, we neglected correlations between uncontrollable factors and their uncertainties, and their pointwise predictions are used at the process optimization step. Results show that the company may decrease the mean of the product weights by 4.27 percent, reducing the production cost. Furthermore, as a result of the problem that was solved in the Bayesian approach, it is observed that correlations and uncertainties of uncontrollable factors may affect the result of the optimization process.

We examined a fully Bayesian experimental design problem that aims directly to improve the result of the process optimization problem. Since two different probabilistic models were required (one for product weight and one for uncontrollable factors) and group effects were considered, we provided a hierarchical model which is suitable for Gibbs sampling. While verifying the hierarchical

model on generated synthetic data, we observed that the Inverse Wishart distribution, which is conjugate prior to the covariance matrix, could be problematic when the number of groups is low and the parameter stays at a low level. Without breaking full conditional conjugacy, some problems were eliminated using Hierarchical Half-t prior. However, both models performed similarly on real data. Later, the selected model was updated to solve the experimental design problem. We observed that candidate design settings might have an important impact on the convergence and effectiveness of Gibbs sampling. Therefore, we recommend computationally cheaper posterior approximation methods which are robust to candidate designs for high dimensional problems in the case of fully Bayesian experimental design whose utility function is also expensive to solve.

Appendix A

Bayesian Hiearchical Model I Full Conditional Distributions

A.1 Distributions

- $y_g^o|\beta_g, \sigma^2, X_o^g \sim N(X_g^o\beta_g, \sigma^2 I_{n_g}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-1}{2\sigma^2}(y_g^o - X_g^o\beta_g)^\top(y_g^o - X_g^o\beta_g)\right)$
- $\beta_g|\bar{\beta}, \Sigma \sim N(\bar{\beta}, \Sigma) = |2\pi\Sigma|^{-1/2} \exp\left(\frac{-1}{2}(\beta_g - \bar{\beta})^\top\Sigma^{-1}(\beta_g - \bar{\beta})\right)\ p(\bar{\beta}) \propto 1$
- $\sigma^2|\frac{\tau}{2}, \frac{\lambda}{2} \sim IG(\frac{\tau}{2}, \frac{\lambda}{2}) = \frac{\frac{\lambda}{2}^{\frac{\tau}{2}}}{\Gamma(\frac{\tau}{2})}(\sigma^2)^{-\frac{\tau}{2}-1} \exp\left(-\frac{\frac{\lambda}{2}}{\sigma^2}\right)$
- $\Sigma|\nu, V \sim IW(\nu+p-1, V) = \frac{|V|^{\frac{\nu+p-1}{2}}}{2^{p(\nu+p-1)/2}\Gamma_p(\frac{\nu+p-1}{2})}|\Sigma|^{-(\nu+2p)/2} \exp\left(\frac{-1}{2}tr(V\Sigma^{-1})\right)$
- $\Sigma_u|\nu_u, V_u \sim IW(\nu_u+p_u-1, V_u) = \frac{|V_u|^{\frac{\nu_u+p_u-1}{2}}}{2^{p_u(\nu_u+p_u-1)/2}\Gamma_p(\frac{\nu_u+p_u-1}{2})}|\Sigma_u|^{-(\nu_u+2p_u)/2} \exp\left(\frac{-1}{2}tr(V_u\Sigma_u^{-1})\right)$
- $vec(A)|\Sigma_u, C, \bar{A} \sim N(vec(\bar{A}), \Sigma_u \otimes C^{-1}) = |2\pi\Sigma_u \otimes C^{-1}|^{-1/2} \exp\left(\frac{-1}{2}(vec(A) - vec(\bar{A}))^\top(\Sigma_u \otimes C^{-1})^{-1}(vec(A) - vec(\bar{A}))\right)$
- $vec(X_{ug}^o)|X_{cg}^o, A, \Sigma_u \sim N(vec(X_{cg}^o A), \Sigma_u \otimes I_{n_g}) = |2\pi\Sigma_u \otimes I_{n_g}|^{-1/2} \exp\left(\frac{-1}{2}(vec(X_{ug}^o) - vec(vec(X_{cg}^o))^\top(\Sigma_u \otimes I_{n_g})^{-1}(vec(X_{ug}^o) - vec(vec(X_{cg}^o))\right)$
- $p(\bar{A}) \propto 1$

A.2 Full Conditional Distributions

A.2.1 $y_g^o|$*everything else*

$\implies y_g^o|\text{everything else} \sim N(X_g^o\beta_g, \sigma^2 I_{n_g})$

A.2.2 $\beta_g|$*everything else*

$$\propto (\sigma^2)^{\frac{-n_g}{2}} \exp\left(\tfrac{-1}{2\sigma^2}(y_g^o - X_g^o\beta_g)^\top(y_g^o - X_g^o\beta_g)\right)|\Sigma|^{\frac{-1}{2}} \exp\left(\tfrac{-1}{2}(\beta_g - \bar{\beta})^\top\Sigma^{-1}(\beta_g - \bar{\beta})\right)$$

$$\propto \exp\left(\tfrac{-1}{2\sigma^2}(y_g^o - X_g^o\beta_g)^\top(y_g^o - X_g^o\beta_g)\right) \exp\left(\tfrac{-1}{2}(\beta_g - \bar{\beta})^\top\Sigma^{-1}(\beta_g - \bar{\beta})\right)$$

Let $U_0^\top U_0$ be Cholesky root of Σ^{-1}

Let $v = \begin{bmatrix} \frac{y_g^o}{\sigma} \\ U_0\bar{\beta} \end{bmatrix}$ and $W = \begin{bmatrix} \frac{X_g^o}{\sigma} \\ U_0 \end{bmatrix}$

$$(y_g^o - X_g^o\beta_g)^\top(y_g^o - X_g^o\beta_g) + (\beta_g - \bar{\beta})^\top U_0^\top U_0(\beta_g - \bar{\beta}) = (v - W\beta_g)^\top(v - W\beta_g)$$

Let $\beta_{0j} = (W^\top W)^{-1}W^\top v = (\frac{X_g^{o\top}X_g^o}{\sigma^2} + \Sigma^{-1})^{-1}(\frac{X_g^{o\top}y_g^o}{\sigma^2} + \Sigma^{-1}\bar{\beta})$

$$\implies \beta_g|\text{everything else} \propto \exp\left(\tfrac{-1}{2}(\beta_g - \beta_{0j})^\top(\tfrac{X_g^{o\top}X_g^o}{\sigma^2} + \Sigma^{-1})(\beta_g - \beta_{0j})\right)$$

$$\implies \beta_g|\text{everything else} \sim N\left(\beta_{0j}, (\tfrac{X_g^{o\top}X_g^o}{\sigma^2} + \Sigma^{-1})^{-1}\right)$$

$$\implies \beta_g|\text{everything else} \sim N\left((\tfrac{X_g^{o\top}X_g^o}{\sigma^2} + \Sigma^{-1})^{-1}(\tfrac{X_g^{o\top}y_g^o}{\sigma^2} + \Sigma^{-1}\bar{\beta}), (\tfrac{X_g^{o\top}X_g^o}{\sigma^2} + \Sigma^{-1})^{-1}\right)$$

A.2.3 $\bar{\beta}|$*everything else*

$$\propto \Pi_{g=1}^{G}\Bigg(p(\beta_g|\Sigma,\bar{\beta})\Bigg)p(\bar{\beta})$$

$$\propto \Pi_{g=1}^{G}\Bigg(|\Sigma|^{\frac{-1}{2}}\exp\left(\tfrac{-1}{2}(\beta_g-\bar{\beta})^\top\Sigma^{-1}(\beta_g-\bar{\beta})\right)\Bigg)$$

$$\propto \exp\Bigg(\sum_{g=1}^{G}\left(\tfrac{-1}{2}(\beta_g-\bar{\beta})^\top\Sigma^{-1}(\beta_g-\bar{\beta})\right)\Bigg)$$

Let $U_0^\top U_0$ be cholesky root of Σ^{-1}

Let $z=\begin{bmatrix}U_0\beta_1\\ \dots\\ U_0\beta_m\end{bmatrix}$ and $T=\begin{bmatrix}U_0\\ \dots\\ U_0\end{bmatrix}$

$$\sum_{g=1}^{G}\Bigg((\beta_g-\bar{\beta})^\top U_0^\top U_0(\beta_g-\bar{\beta})\Bigg)=(z-T\bar{\beta})^\top(z-T\bar{\beta})$$

$$\text{Let } \beta_b=(T^\top T)^{-1}T^\top z=\Bigg(\sum_{g=1}^{G}(\Sigma^{-1})\Bigg)^{-1}\Bigg(\sum_{g=1}^{G}(\Sigma^{-1}\beta_g)\Bigg)$$

$$T^\top T=\Bigg(\sum_{g=1}^{G}(\Sigma^{-1})\Bigg)$$

$$\implies \propto \exp\Bigg(\tfrac{-1}{2}(\bar{\beta}-\beta_b)^\top\Bigg(\sum_{g=1}^{G}(\Sigma^{-1})\Bigg)(\bar{\beta}-\beta_b)\Bigg)$$

$$\implies \bar{\beta}|\text{everything else}\sim N\Bigg(\beta_b,\Bigg(\sum_{g=1}^{G}(\Sigma^{-1})\Bigg)^{-1}\Bigg)$$

$$\implies \bar{\beta}|\text{everything else}\sim N\Bigg(\Bigg(\sum_{g=1}^{G}(\Sigma^{-1})\Bigg)^{-1}\Bigg(\sum_{g=1}^{G}(\Sigma^{-1}\beta_g)\Bigg),\Bigg(\sum_{g=1}^{G}(\Sigma^{-1})\Bigg)^{-1}\Bigg)$$

A.2.4 $\sigma^2|$*everything else*

$$\propto \Pi_{g=1}^{G}\Bigg(p(y_g^o|\sigma^2,\beta_g)\Bigg)p(\sigma^2|\tfrac{\tau}{2},\tfrac{\lambda}{2})$$

$$\propto \Pi_{g=1}^{G}\Bigg((\sigma^2)^{\frac{-n_g}{2}}\exp\left(\tfrac{-1}{2\sigma^2}(y_g^o-X_g^o\beta_g)^\top(y_g^o-X_g^o\beta_g)\right)\Bigg)\tfrac{\frac{\lambda}{2}^{\frac{\tau}{2}}}{\Gamma(\frac{\tau}{2})}(\sigma^2)^{-\frac{\tau}{2}-1}\exp\left(-\tfrac{\frac{\lambda}{2}}{\sigma^2}\right)$$

$$\propto \Pi_{g=1}^{G}\Bigg((\sigma^2)^{\frac{-n_g}{2}}\exp\left(\tfrac{-1}{2\sigma^2}(y_g^o-X_g^o\beta_g)^\top(y_g^o-X_g^o\beta_g)\right)\Bigg)(\sigma^2)^{-\frac{\tau}{2}-1}\exp\left(-\tfrac{\frac{\lambda}{2}}{\sigma^2}\right)$$

$$\propto (\sigma^2)^{\frac{-n}{2}}\exp\left(\sum_{g=1}^{G}\left(\tfrac{-1}{2\sigma^2}(y_g^o-X_g^o\beta_g)^\top(y_g^o-X_g^o\beta_g)\right)\right)(\sigma^2)^{-\frac{\tau}{2}-1}\exp\left(-\tfrac{\frac{\lambda}{2}}{\sigma^2}\right)$$

$$\implies \sigma^2|\text{everything else} \sim IG(\tfrac{\tau_n}{2},\tfrac{\lambda_n}{2})$$

where

$$\tau_n = \tau + n$$

$$\lambda_n = \lambda + \sum_{g=1}^{G}\left((y_g^o-X_g^o\beta_g)^\top(y_g^o-X_g^o\beta_g)\right)$$

A.2.5 $\Sigma|$*everything else*

$$\Sigma|\text{everything else} \propto \Pi_{g=1}^{G}\Bigg(p(\beta_g|\Sigma,\bar{\beta})\Bigg)p(\Sigma|\nu,V))$$

$$\propto \Pi_{g=1}^{G}\Bigg(|\Sigma|^{\frac{-1}{2}}\exp\left(\tfrac{-1}{2}(\beta_g-\bar{\beta})^\top\Sigma^{-1}(\beta_g-\bar{\beta})\right)\Bigg)|\Sigma|^{-(v+2p)/2}\exp\left(\tfrac{-1}{2}tr(V\Sigma^{-1})\right)$$

$$\propto |\Sigma|^{\frac{-G}{2}}\exp\left(\sum_{g=1}^{G}\left(\tfrac{-1}{2}(\beta_g-\bar{\beta})^\top\Sigma^{-1}(\beta_g-\bar{\beta})\right)\right)|\Sigma|^{-(v+2p)/2}\exp\left(\tfrac{-1}{2}tr(V\Sigma^{-1})\right)$$

$\exp\left(\sum_{g=1}^{G}\left(\tfrac{-1}{2}(\beta_g-\bar{\beta})^\top\Sigma^{-1}(\beta_g-\bar{\beta})\right)\right)$ can be written as $\exp\left(\tfrac{-1}{2}tr(S\Sigma^{-1})\right)$

where $S = \sum_{g=1}^{G} \left((\beta_g - \bar{\beta})(\beta_g - \bar{\beta})^\top \right)$

Then the expression is

$$\propto |\Sigma|^{\frac{-J}{2}} \exp\left(\tfrac{-1}{2} tr(S\Sigma^{-1}) \right) |\Sigma|^{-(v+2p)/2} \exp\left(\tfrac{-1}{2} tr(V\Sigma^{-1}) \right)$$

$$\implies \Sigma|\text{everything else} \sim IW(\nu + p - 1 + G, V + S)$$

A.2.6 $vec(A)|$*everything else*

$$A|\text{everything else} \propto p(X_u^o|X_c^o, A, \Sigma_u) p(A|\Sigma_u, C, \bar{A})$$

$$\propto |\Sigma_u|^{-n/2} \exp tr\left(\tfrac{-1}{2} (X_u^o - X_c^o A)^\top (X_u^o - X_c^o A) \Sigma_u^{-1} \right) |\Sigma_u|^{-p_u/2} |C|^{-p_c/2} \exp tr\left(\tfrac{-1}{2} (A - \bar{A})^\top C (A - \bar{A}) \Sigma_u^{-1} \right)$$

$$\propto \exp tr\left(\tfrac{-1}{2} (X_u^o - X_c^o A)^\top (X_u^o - X_c^o A) \Sigma_u^{-1} \right) \exp tr\left(\tfrac{-1}{2} (A - \bar{A})^\top C (A - \bar{A}) \Sigma_u^{-1} \right)$$

Notice that,

$$(X_u^o - X_c^o A)^\top (X_u^o - X_c^o A) + (A - \bar{A})^\top C (A - \bar{A}) = (Z - MA)^\top (Z - MA)$$

where $Z = \begin{bmatrix} X_u^o \\ U_2 \bar{A} \end{bmatrix}$, $M = \begin{bmatrix} X_c^o \\ U_2 \end{bmatrix}$ and $U_2^\top U_2 = C$

Let $A_0 = (M^\top M)^{-1} M^\top Z = (X_c^\top X_c + C)^{-1} (X_c^{o\top} X_u^o + C\bar{A})$, then

$$(X_u^o - X_c^o A)^\top (X_u^o - X_c^o A) + (A - \bar{A})^\top C (A - \bar{A}) = (A - A_0) M^\top M (A - A_0) + ..$$

where $M^\top M = (X_c^{o\top} X_c^o + C)$, then the expression is

$$\propto \exp tr\left(\tfrac{-1}{2} (A - A_0) M^\top M (A - A_0) \Sigma_u^{-1} \right)$$

$$\propto \exp tr\left(\frac{-1}{2}(A-A_0)(X_c^{o\top}X_c^o+C)(A-A_0)\Sigma_u^{-1}\right)$$

$$\implies vec(A)|\text{everything else} \sim N(vec(A_0), \Sigma_u \otimes (X_c^{o\top}X_c^o+C)^{-1})$$

$$\implies vec(A)|\text{everything else} \sim N\left(vec\left((X_c^{o\top}X_c^o+C)^{-1}(X_c^\top X_u^o+C\bar{A})\right), \Sigma_u \otimes (X_c^{o\top}X_c^o+C)^{-1}\right)$$

A.2.7 $vec(\bar{A})|$*everything else*

$$\bar{A}|\text{everything else} \propto p(A|\Sigma_u, C, \bar{A})p(\bar{A})$$

$$\propto |\Sigma_u|^{-p_u/2}|C|^{-p_c/2}\exp tr\left(\frac{-1}{2}(A-\bar{A})^\top C(A-\bar{A})\Sigma_u^{-1}\right)$$

$$\implies vec(\bar{A})|\text{everything else} \sim N\left(vec(A), \Sigma_u \otimes C^{-1}\right)$$

A.2.8 $\Sigma_u|$*everything else*

$$\Sigma_u|\text{everything else} \propto p(X_u^o|X_c, A, \Sigma_u)p(A|\Sigma_u, C, \bar{A})p(\Sigma_u|\nu_u, V_u)$$

$$\propto |\Sigma_u|^{-n/2}\exp tr\left(\frac{-1}{2}(X_u^o-X_c^oA)^\top(X_u^o-X_c^oA)\Sigma_u^{-1}\right)|\Sigma_u|^{-p_u/2}\exp tr\left(\frac{-1}{2}(A-\bar{A})^\top C(A-\bar{A})\Sigma_u^{-1}\right)|\Sigma_u|^{-(\nu_u+2p_u)/2}\exp\left(\frac{-1}{2}tr(V_u\Sigma_u^{-1})\right)$$

Let $K = (X_u^o-X_c^oA)^\top(X_u^o-X_c^oA)+(A-\bar{A})^\top C(A-\bar{A})$

Then expression can be written as

$$\propto |\Sigma_u|^{-n/2}|\Sigma_u|^{-p_u/2}|\Sigma_u|^{-(\nu_u+2p_u)/2}\exp\left(\frac{-1}{2}tr((K+V_u)\Sigma_u^{-1})\right)$$

$$\implies \Sigma_u|\text{everything else} \sim IW(\nu_u+p_u+n-1, K+V_u)$$

A.2.9 $X^o_{ug}|$*everything else*

$X^o_{ug}|$everything else $\propto p(y^o_g|\sigma^2, \beta_g, X^o_g)p(X^o_{ug}|X^o_{cg}, A, \Sigma_u)$

$\propto \exp\left(\frac{-1}{2\sigma^2}(y^o_g - X^o_g\beta_g)^\top(y^o_g - X^o_g\beta_g)\right)|\Sigma_u|^{-n/2}\exp tr\left(\frac{-1}{2}(X^o_{ug} - X^o_{cg}A)^\top(X^o_{ug} - X^o_{cg}A)\Sigma_u^{-1}\right)$

$\propto \exp\left(\frac{-1}{2\sigma^2}(y^o_g - X^o_g\beta_g)^\top(y^o_g - X^o_g\beta_g)\right)\exp tr\left(\frac{-1}{2}(X^o_{ug} - X^o_{cg}A)^\top(X^o_{ug} - X^o_{cg}A)\Sigma_u^{-1}\right)$

Recall that $X^o_g = [X^o_{cg} \quad X^o_{ug}]$ and $\beta_g^\top = [\beta_{cg}^\top \quad \beta_{ug}^\top]$

Then the expression can be written as

$\propto \exp\left(\frac{-1}{2\sigma^2}(y^o_g - X^o_{cg}\beta_{cg} - X^o_{ug}\beta_{ug})^\top(y^o_g - X^o_{cg}\beta_{cg} - X^o_{ug}\beta_{ug})\right)\exp tr\left(\frac{-1}{2}(X^o_{ug} - X^o_{cg}A)^\top(X^o_{ug} - X^o_{cg}A)\Sigma_u^{-1}\right)$

Let tilde operator denotes the operation $\tilde{a} = \frac{a}{\sigma}$. Then,

$\propto \exp\left(\frac{-1}{2}(\tilde{y}^o_g - X^o_{cg}\tilde{\beta}_{cg} - X^o_{ug}\tilde{\beta}_{ug})^\top(\tilde{y}^o_g - X^o_{cg}\tilde{\beta}_{cg} - X^o_{ug}\tilde{\beta}_{ug})\right)\exp tr\left(\frac{-1}{2}(X^o_{ug} - X^o_{cg}A)^\top(X^o_{ug} - X^o_{cg}A)\Sigma_u^{-1}\right)$

Use vec operator to write the expression as a familiar form,

$(\tilde{y}^o_g - X^o_{cg}\tilde{\beta}_{cg} - vec(X^o_{ug}\tilde{\beta}_{ug}))^\top(\tilde{y}^o_g - X^o_{cg}\tilde{\beta}_{cg} - vec(X^o_{ug}\tilde{\beta}_{ug})) = (\tilde{y}^o_g - X^o_{cg}\tilde{\beta_{cg}} - (\tilde{\beta}_{ug}^\top \otimes I_{n_g})vec(X^o_{ug}))^\top(\tilde{y}^o_g - X^o_{cg}\tilde{\beta_{cg}} - (\tilde{\beta}_{ug}^\top \otimes I_{n_g})vec(X^o_{ug}))$

and,

$tr\left((X^o_{ug} - X^o_{cg}A)^\top(X^o_{ug} - X^o_{cg}A)\Sigma_u^{-1}\right) = (vec(X^o_{cg}A) - vec(X^o_{ug}))^\top(\Sigma_u^{-1} \otimes I_{n_g})(vec(X^o_{cg}A) - vec(X^o_{ug}))$

Let $U_4^\top U_4$ Cholesky root of $\left(\Sigma_u^{-1} \otimes I_{n_g}\right)$

and $s = \begin{bmatrix} \tilde{y}_g^o - X_{cg}^o \tilde{\beta_{cg}} \\ U_4 vec(X_{cg}^o A) \end{bmatrix}$ and $N = \begin{bmatrix} (\tilde{\beta}_{ug}^\top \otimes I_{n_g}) \\ U_4 \end{bmatrix}$

Then,

$(\tilde{y}_g^o - X_{cg}^o \tilde{\beta_{cg}} - (\tilde{\beta}_{ug}^\top \otimes I_{n_g}) vec(X_{ug}^o))^\top (\tilde{y}_g^o - X_{cg}^o \tilde{\beta_{cg}} - (\tilde{\beta}_{ug}^\top \otimes I_{n_g}) vec(X_{ug}^o) + (vec(X_{cg}^o A) - vec(X_{ug}^o))^\top (\Sigma_u^{-1} \otimes I_{n_g})(vec(X_{cg}^o A) - vec(X_{ug}^o)) = (s - N vec(X_{ug}^o))^\top (s - N vec(X_{ug}^o))$

Let

$t_0 = (N^\top N)^{-1} N^\top s = \left((\tilde{\beta}_{ug}^\top \otimes I_{n_g})^\top (\tilde{\beta}_{ug}^\top \otimes I_{n_g}) + (\Sigma_u^{-1} \otimes I_{n_g})\right)^{-1} \left((\tilde{\beta}_{ug}^\top \otimes I_{n_g})^\top (\tilde{y}_g^o - X_{cg}^o \tilde{\beta_{cg}}) + (\Sigma_u^{-1} \otimes I_{n_g}) vec(X_{cg}^o A)\right)$

$= \left((\tilde{\beta}_{ug} \tilde{\beta}_{ug}^\top \otimes I_{n_g}) + (\Sigma_u^{-1} \otimes I_{n_g})\right)^{-1} \left((\tilde{\beta}_{ug}^\top \otimes I_{n_g})^\top (\tilde{y}_g^o - X_{cg}^o \tilde{\beta_{cg}}) + (\Sigma_u^{-1} \otimes I_{n_g}) vec(X_{cg}^o A)\right)$

$= \left((\tilde{\beta}_{ug} \tilde{\beta}_{ug}^\top + \Sigma_u^{-1}) \otimes I_{n_g}\right)^{-1} \left((\tilde{\beta}_{ug}^\top \otimes I_{n_g})^\top (\tilde{y}_g^o - X_{cg}^o \tilde{\beta_{cg}}) + (\Sigma_u^{-1} \otimes I_{n_g}) vec(X_{cg}^o A)\right)$

Notice that $(N^\top N) = \left((\tilde{\beta}_{ug} \tilde{\beta}_{ug}^\top + \Sigma_u^{-1}) \otimes I_{n_g}\right)$

Then,

$(s - N vec(X_{ug}^o))^\top (s - N vec(X_{ug}^o)) = (vec(X_{ug}^o) - t_0)^\top \left((\tilde{\beta}_{ug} \tilde{\beta}_{ug}^\top + \Sigma_u^{-1}) \otimes I_{n_g}\right)(vec(X_{ug}^o) - t_0) + ..$

Thus, $X_{ug}^o|$everything else $\propto \exp\left(\frac{-1}{2}(vec(X_{ug}^o) - t_0)^\top \left((\tilde{\beta}_{ug} \tilde{\beta}_{ug}^\top + \Sigma_u^{-1}) \otimes\right.\right.$

$$I_{n_g}\Big)(vec(X_{ug}^o) - t_0)\Big)$$

$$vec(X_{ug}^o)|\text{everything else} \sim N\Big(t_0, ((\tilde{\beta}_{ug}\tilde{\beta}_{ug}^\top + \Sigma_u^{-1})^{-1} \otimes I_{n_g})\Big)$$

Appendix B

Synthetic Data Generation Process

- Let u_f's be random vectors with compatible sizes and each element is uniformly distributed between 0.95 and 1.05
- Let d_f's be random vectors of Dirichlet distribution with compatible sizes
- Let LKJ_f's be LKJ random matrices having parameter 1 with compatible sizes
- Let $\bar{A}$ be the OLS estimator of the coefficient matrix of the fitted multivariate linear regression between uncontrollable factors and controllable factors. Let S be OLS estimator of standard deviation of this model.
- Set $\bar{A}$ as $diag(u_1)\bar{A}diag(u_2)$
- Set S as $diag(S)diag(u_3)$.
- Set Σ_u as $S(LKJ_1)S$
- Set C as Identity matrix
- Generate A from $N(vec(\bar{A}), \Sigma_u \otimes C^{-1})$
- Generate Errors from $N(0, \Sigma_u)$
- Set X_u as X_cA+Errors
- Set X as $[X_cX_u]$
- For each group of product weight fit a linear regression and keep their OLs estimators in $\bar{\beta_g}$ and σ_g.

- Set s as standard deviation of $\bar{\beta}_g$'s
- Set $\bar{\beta}$ as sum of $\bar{\beta}_g d_1$
- Set s as $diag(s)u_4/2$
- Set Σ as $s(LKJ_2)s$
- Set σ as sum of $\sigma_g d_2$
- Generate errors from $N(0, \sigma^2)$
- Generate β_g's from $N(\bar{\beta}, \Sigma)$
- Generate y_g's from corresponding $X_g\beta_g$ and add errors

Appendix C

Bayesian Hiearchical Model II Full Conditional Distribution Updates

C.1 Equality

- $V_a = diag(1/a_1, ...1/a_p)$

C.2 Distributions

- $\sigma^2|\frac{1}{2}, \frac{\lambda}{2} \sim IG(\frac{1}{2}, \frac{1}{A_k^2}) = \frac{(A_k^2)^{\frac{-1}{2}}}{\Gamma(\frac{1}{2})}(\sigma^2)^{-\frac{1}{2}-1} \exp\left(-\frac{A_k^{-2}}{\sigma^2}\right)$

C.3 Full Conditional Distributions

C.3.1 $a_k|$*everything else*

$a_k|$everything else $\propto p(\Sigma|\nu, a_1, ..a_p))p(a_k|A_k^2)$

$\propto \frac{|2\nu V_a|^{\frac{\nu+p-1}{2}}}{2^{(\nu+p-1)p/2}\Gamma_p(\frac{\nu+p-1}{2})}|\Sigma|^{-(\nu+2p)/2}\exp\left(\frac{-1}{2}tr(2\nu V_a\Sigma^{-1})\right)\frac{A_k^{-1}}{\Gamma(1/2)}a_k^{-3/2}\exp\left(\frac{-1}{A_k^2 a_k}\right)$

$\propto |2\nu V_a|^{\frac{\nu+p-1}{2}}\exp\left(\frac{-1}{2}tr(2\nu V_a\Sigma^{-1})\right)a_k^{-3/2}\exp\left(\frac{-1}{A_k^2 a_k}\right)$

$\propto \Pi_{k=1}^p(2\nu/a_k)^{\frac{\nu+p-1}{2}}\exp\left(\sum_{j=1}^p(\frac{-\nu\Sigma_{kk}^{-1}}{a_k})\right)a_k^{-3/2}\exp\left(\frac{-1}{A_k^2 a_k}\right)$ where Σ_{kk}^{-1} denotes (k,k) element of Σ^{-1}

$\propto a_k^{\frac{-\nu-p+1}{2}}\exp\left(\frac{-v\Sigma_{kk}^{-1}}{a_k})\right)a_k^{-3/2}\exp\left(\frac{-1}{A_k^2 a_k}\right) \implies$

$a_k \sim IG(\frac{v+p}{2}, v\Sigma_{kk}^{-1} + 1/A_k^2)$

Bibliography

[1] D. V. Lindley, *Bayesian statistics: A review*. SIAM, 1972.

[2] M. Ryan, Drovandi and Pettitt, "Fully bayesian optimal experimental design: A review," 2016.

[3] K. Chaloner and I. Verdinelli, "Bayesian experimental design: A review," *Statistical Science*, pp. 273–304, 1995.

[4] R. H. Myers, D. C. Montgomery, and C. M. Anderson-Cook, *Response surface methodology: process and product optimization using designed experiments*. John Wiley & Sons, 2016.

[5] E. Del Castillo, *Process optimization: a statistical approach*, vol. 105. Springer Science & Business Media, 2007.

[6] E. G. Ryan, C. C. Drovandi, and A. N. Pettitt, "Fully bayesian experimental design for pharmacokinetic studies," *Entropy*, vol. 17, no. 3, pp. 1063–1089, 2015.

[7] A. Gelman, J. Carlin, H. Stern, D. Dunson, A. Vehtari, and D. Rubin, *Bayesian Data Analysis, Third Edition*. Chapman & Hall/CRC Texts in Statistical Science, Taylor & Francis, 2013.

[8] N. Chopin, "A sequential particle filter method for static models," *Biometrika*, vol. 89, no. 3, pp. 539–551, 2002.

[9] X. Huan and Y. M. Marzouk, "Simulation-based optimal bayesian experimental design for nonlinear systems," *Journal of Computational Physics*, vol. 232, pp. 288–317, jan 2013.

[10] P. Müller and G. Parmigiani, "Optimal design via curve fitting of monte carlo experiments," *Journal of the American Statistical Association*, vol. 90, no. 432, pp. 1322–1330, 1995.

[11] C. Bielza, P. Müller, and D. R. Insua, "Decision analysis by augmented probability simulation," *Management Science*, vol. 45, no. 7, pp. 995–1007, 1999.

[12] E. LeDell, N. Gill, S. Aiello, A. Fu, A. Candel, C. Click, T. Kraljevic, T. Nykodym, P. Aboyoun, M. Kurka, and M. Malohlava, *h2o: R Interface for the 'H2O' Scalable Machine Learning Platform*, 2022. R package version 3.36.0.3.

[13] J. Ypma, "Introduction to nloptr: an r interface to nlopt," *R Package*, vol. 2, 2014.

[14] K. Triantafyllopoulos, "Moments and cumulants of the multivariate real and complex gaussian distributions," 01 2002.

[15] A. Gelman, "Prior distributions for variance parameters in hierarchical models (comment on article by browne and draper)," *Bayesian analysis*, vol. 1, no. 3, pp. 515–534, 2006.

[16] A. Gelman and J. Hill, *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Analytical Methods for Social Research, Cambridge University Press, 2006.

[17] I. Alvarez, J. Niemi, and M. Simpson, "Bayesian inference for a covariance matrix," *arXiv preprint arXiv:1408.4050*, 2014.

[18] A. Huang and M. P. Wand, "Simple marginally noninformative prior distributions for covariance matrices," *Bayesian Analysis*, vol. 8, no. 2, pp. 439–452, 2013.

[19] A. Vehtari, A. Gelman, and J. Gabry, "Practical bayesian model evaluation using leave-one-out cross-validation and WAIC," *Statistics and Computing*, vol. 27, pp. 1413–1432, aug 2016.

[20] A. Vehtari, J. Gabry, M. Magnusson, Y. Yao, P.-C. Bürkner, T. Paananen, and A. Gelman, "loo: Efficient leave-one-out cross-validation and waic for bayesian models," 2022. R package version 2.5.0.

[21] B. Carpenter, A. Gelman, M. D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. Brubaker, J. Guo, P. Li, and A. Riddell, "Stan: A probabilistic programming language," *Journal of statistical software*, vol. 76, no. 1, 2017.