Spring 2019

Sensitivity Analyses for Tumor Growth Models

Ruchini Dilinika Mendis
Western Kentucky University, ruchini.mendis808@topper.wku.edu

SENSITIVITY ANALYSES FOR TUMOR GROWTH MODELS

A Thesis
Presented to
The Faculty of the Department of Mathematics
Western Kentucky University
Bowling Green, Kentucky

In Partial Fulfillment
Of the Requirements for the Degree
Master of Science

By
Ruchini D. Mendis

May 2019

SENSITIVITY ANALYSES FOR TUMOR GROWTH MODELS

Date Recommended 4/9/19

Ngoc Nguyen

Dr. Ngoc Nguyen, Director of Thesis

Ferhan Atici

Dr. Ferhan Atici

Dr. Richard Schugart

Cheryl O. Davis 4/18/19

Dean, Graduate Studies and Research Date

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor, Dr.Ngoc Nguyen for the great support that I received related to my research, for her understanding, patience, motivation and for all the guidance that she gave me through her valuable comments.

My deepest gratitude also goes to the rest of my thesis committee, Dr. Ferhan Atici and Dr. Richard Schugart for their insightful comments, advice and support.

I would also like to thank Dr. Saliya Kumara, Duc Nguyen and all my friends for helping me to make this succeed.

I am grateful to my parents and my brother who have been encouraging me in countless ways and for their great support. Also, I would like to express my sincere thanks to my husband, for bearing all the difficulties and helping me to succeed in this thesis and my life in general.

Finally, I am thankful to math faculty at WKU for investing your time in me and giving me an opportunity to get experience in teaching.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

SENSITIVITY ANALYSES FOR TUMOR GROWTH MODELS

Ruchini Mendis May 2019 59 Pages

Directed by: Dr. Ngoc Nguyen, Dr. Ferhan Atici and Dr. Richard Schugart

Department of Mathematics Western Kentucky University

This study consists of the sensitivity analysis for two previously developed tumor growth models: Gompertz model and quotient model. The two models are considered in both continuous and discrete time. In continuous time, model parameters are estimated using least-square method, while in discrete time, the partial-sum method is used. Moreover, frequentist and Bayesian methods are used to construct confidence intervals and credible intervals for the model parameters. We apply the Markov Chain Monte Carlo (MCMC) techniques with the Random Walk Metropolis algorithm with Non-informative Prior and the Delayed Rejection Adoptive Metropolis (DRAM) algorithm to construct parameters' posterior distributions and then obtain credible intervals.

Chapter 1

INTRODUCTION

Many mathematical models exhibit model errors as well as measurement errors which can affect the parameters associated with the model. Sensitivity analysis can be used to study those uncertainties of parameters in the model. This study consists of a sensitivity analysis using the frequentist and Bayesian analysis methods for two previously developed tumor growth models.

We begin with modeling unperturbed tumor growth using different models: Gompertz model and quotient model. The first tumor growth model [9], which we call quotient model, consists of two parameters: λ_0 for the exponential growth phase and λ_1 for the linear growth phase. We also consider the Gompertz model for the growth of tumor, which consists of two parameters α and K. This Gompertz model belongs to the family of sigmoidal curves.

The two tumor models are initially presented as an Initial Value Problem (IVP) of differential equations. We keep the time interval of these IVPs as continuous to do our study in the continuous setting. In a discrete setting, we discretize the time interval, and then obtain the difference equations for both tumor models. To obtain the parameter estimations in a continuous setting, we use a least-square method, while in a discrete setting, the partial-sum method is used.

After obtaining the estimated parameters, we use two different approaches to construct confidence intervals and credible intervals for the parameters of the models: the frequentist approach and the Bayesian approach. Frequentist inferences assume that the unknown parameters are fixed constants where probabilities are defined by using relative frequencies. Bayesian methods treat parameters as random variables where probabilities are subjective and one can make probability statements about

parameters.

The focus of this study is on the Bayesian techniques for sensitivity analysis. We apply the Markov Chain Monte Carlo (MCMC) techniques using the Random Walk Metropolis algorithm with Non-informative Prior and the Delayed Rejection Adoptive Metropolis algorithm. These Bayesian algorithms are used to obtain posterior distributions for the model parameters and to construct credible intervals. We use these algorithms for simulated data with constant variance. We also consider Gompertz model with non-constant variance noise component.

At the end of this study, we analyze the sensitivity of the parameters of perturbed tumor model using experimental data provided in the paper [9].

The study consists of six chapters. Chapter 1 includes an introduction of the sensitivity analysis with a summary of the models and methods. Chapter 2 provides necessary definitions and ideas of the frequentist and Bayesian approaches. Chapter 3 discusses the quotient model in both discrete and continuous times and provides the results of the sensitivity analysis for the model. Chapter 4 gives the results of the sensitivity analysis for the Gompertz model using simulated data with and without the constant variance assumption. Chapter 5 presents results of the sensitivity analysis for the perturbed tumor model using clinical data. Chapter 6 concludes the study.

Chapter 2

FREQUENTIST AND BAYESIAN INFERENCES

2.1 Frequentist Inferences

The frequentist approach is the classical way of making inferences. In the frequentist approach, probabilities are not updated as the experiment is repeated a large number of times. In other words, the unknown parameters are assumed to be fixed constants and they define probability by using limiting relative frequencies [11]. To start the sensitivity analysis for a model, we consider statistical models of the form

$$w_i = f_i(q) + \epsilon_i, \quad i = 1, ..., n, \tag{2.1.1}$$

where q denotes the parameter vector, $f_i(q)$ is the model response, and ϵ_i is the modeling and measurement errors [13]. We assume that ϵ_i follows a Normal distribution with mean zero and constant variance σ^2. When estimating parameters, the goal is to use the realized data w_i from the observations of the model and find a value of $\hat{q}$ which minimizes

$$SS(q) = \sum_{i=1}^{n} (f_i(q) - w_i)^2. \tag{2.1.2}$$

When determining this Ordinary Least Squares (OLS) estimator, numerical optimization techniques are often employed. The estimator $\hat{q}(W)$ is a random variable whose realized value depends on the observed data w_i of random variable W_i.

Sensitivity Matrix

Let $\chi(\hat{q})$ represent the $n \times p$ sensitivity matrix which is constructed by

$$\chi_{ik}(\hat{q}) = \frac{\partial f_i(\hat{q})}{\partial q_k}; \quad i = 1, ..., n; \quad k = 1, ..., p. \tag{2.1.3}$$

An estimate for the variance-covariance matrix is obtained by

$$\Sigma_0 \approx \Sigma(\hat{q}) = \sigma^2[\chi^\top(\hat{q})\chi(\hat{q})]^{-1}, \tag{2.1.4}$$

where σ^2 is estimated by

$$\sigma^2 \approx \hat{\sigma}^2 = \frac{1}{n-p}\sum_{i=1}^{n}(f_i(\hat{q}) - w_i)^2. \tag{2.1.5}$$

Sampling Distribution

The distribution of $\hat{q}(w)$ obtained through repeated sampling is called the sampling distribution. Under the assumptions which specify that errors are independent and identically distributed (iid) and $\epsilon_i \sim N(0, \sigma^2)$, as the sample size n approaches infinity, the sampling distribution of $\hat{q}(w)$ is approximated by

$$\hat{q} \sim N(q, \sigma^2[\chi^\top(q)\chi(q)]^{-1}), \tag{2.1.6}$$

where $\chi(q)$ is an $n \times p$ sensitivity matrix.

Confidence Intervals

If δ_k represents the k^{th} diagonal element of $[\chi^\top(q)\chi(q)]^{-1}$, then the $(1-\alpha)\times 100\%$ confidence interval is given by [13]

$$[q_k - t_{n-p,1-\frac{\alpha}{2}}\sigma\sqrt{\delta_k}, \quad q_k + t_{n-p,1-\frac{\alpha}{2}}\sigma\sqrt{\delta_k}], \tag{2.1.7}$$

where n is the number of observations, p is the number of parameters in the model and $n-p$ is the degree of freedom of the t-distribution. For implementation purposes, it can be simplified using the variance -covariance matrix as the following form [13].

$$[q_{OLS,k} - t_{n-p,1-\frac{\alpha}{2}}SE, \quad q_{OLS,k} + t_{n-p,1-\frac{\alpha}{2}}SE], \tag{2.1.8}$$

where $q_{OLS,k}$ is the k^{th} element of OLS estimator of q and the standard error is $SE \approx \sqrt{\Sigma_k}$. Here Σ_k is the k^{th} diagonal element of $\Sigma(\hat{q})$.

2.2 Bayesian Inferences

A Bayesian approach treats parameters as random variables and assumes probabilities are subjective that we can make probability statements about parameters [11]. A probability distribution for parameter q is denoted by $\pi_0(q)$, which is known as a prior distribution and contains knowledge about the parameters before acquiring observations w. If the prior knowledge is questionable or unknown, we use an improper uniform distribution as a non-informative prior. After the data is observed, the prior distribution of parameters are updated conditionally to reflect this new information based on sample values and generate posterior distribution of parameters.

The posterior density is calculated as

$$\pi(q|w) = \frac{\pi(w|q)\pi_0(q)}{\int \pi(w|q)\pi_0(q)dq} \tag{2.2.1}$$

where $\pi_0(q)$ is the prior distribution, $\pi(q|w)$ is the posterior distribution, q is the parameter values, w is the observed data of a random sample, and $\pi(w|q)$ is the likelihood function [13]. In Bayesian analysis, all inferences follow from the posterior distribution. In practice, one can use analytical methods to compute posterior distribution only if the problem or model is very basic. Thus most Bayesian analyses uses Markov Chain Monte Carlo (MCMC) techniques, which construct a Markov chain whose stationary distribution is the posterior distributions of the model parameters.

Random Walk Metropolis Algorithm

The MCMC method is considered as a general simulation method for computing posterior distributions. Markov chain is a sequence of random variables for which the random variable depends only on the previous value [11]. The Metropolis algorithm was founded by the American physicist and computer scientist Nicholas C. Metropolis and it consists of MCMC techniques that can be used to obtain random samples from any arbitrary target distribution. To use the Metropolis algorithm, we need to have an initial parameter value and a symmetric proposal distribution [11]. In this study, we consider the proposal distribution as a normal distribution with the mean equal to the previous value of q. The Metropolis-Hastings algorithm is used to decide which proposed value of q to accept or reject. The specific algorithm works as follows [13];

1. Let q_0 be an initial parameter vector and initialize the Markov Chain X as $X_0 = q_0$ and iterate M times.

2. Take the current chain realization to be $X_{k-1} = q^{k-1}$

3. In each iteration, propose a new value q^* which chosen from a Normal distribution that centers at the value q from the previous iteration and the variance-covariance matrix $\Sigma(\hat{q})$ where $\hat{q}$ is the ordinary least square estimates of the parameters.

   Then compute the ratio

$$\alpha(q^*|q^{k-1}) = \frac{\pi(q^*|w)}{\pi(q^{k-1}|w)} = \frac{\pi(w|q^*)\pi_0(q^*)}{\pi(w|q^{k-1})\pi_0(q^{k-1})}. \tag{2.2.2}$$

4. With probability determined by the likelihood function and prior density, either accept $q^* : X_k = q^*$ or otherwise, take $X_k = q^{k-1}$.

5. Establish that the posterior density is the stationary distribution for the chain.

Since the construction of q^* depends on the previous value q^{k-1}, this is termed as random walk or local Metropolis algorithm. Methods of constructing the proposal distribution and the probability of acceptance may vary depending on different models and algorithms.

With regular assumptions for the error terms and with fixed σ^2, the likelihood function is given as follows [13]

$$\pi(w|q) = \frac{1}{(2\pi\sigma^2)^{n/2}} e^{-SS(q)/2\sigma^2}, \tag{2.2.3}$$

where $SS(q)$ is the sum of square errors as in Equation (2.1.2).

For implementation purposes we can use this likelihood function to establish another version to the ratio $\alpha(q^*|q^{k-1})$ [13]

$$\alpha(q^*|q^{k-1}) = \frac{\pi(w|q^*)}{\pi(w|q^{k-1})} = \frac{e^{-SS(q^*)/2\sigma^2}}{e^{-SS(q^{k-1})/2\sigma^2}} = e^{-[SS(q^*)-SS(q^{k-1})]/2\sigma^2}. \tag{2.2.4}$$

The new candidate q^* is accepted with probability

$$\alpha(q^*|q^{k-1}) = min(1, \frac{\pi(q^*|w)}{\pi(q^{k-1}|w)}),$$

and if rejected, the prior chain value q^{k-1} will be retained [13]. Then the initial variance estimate given in Equation (2.1.5) is updated using an inverse gamma distribution [13].

$$\sigma^2|(w,q) \sim Inv - gamma(0.5(n_s+n), 0.5(n_s\sigma_s^2 + SS(q))), \tag{2.2.5}$$

where n_s is the number of observations that provides the information encoded in the

prior, σ_s^2 is the mean square error of the observations and n is the total number of observations [13, 5]. In practice n_s can be taken from 0.01 to 1 [5].

Delayed Rejection Adaptive Metropolis (DRAM)

DRAM consists of two main algorithms: a Delayed Rejection algorithm and an Adaptive Metropolis algorithm. In a nonadaptive period of length k_0, initial variance-covariance matrix is used to compute chain values. When adaptation begins, the chain covariance matrix is updated at the k^{th} step as [13]

$$\Sigma_k = s_p cov(q^0, q^1, ..., q^{k-1}) + \epsilon I_p, \tag{2.2.6}$$

where s_p is a design parameter, $\epsilon > 0$, and I_p is the p-dimensional identity matrix. One has to choose ϵ very small constant in order to make sure Σ_k is positive definite [6]. In practice we take the scaling parameter to be $s_p = 2.38^2/p$ [6], as it is proven in Gelman et al.(1995) that, in a certain sense, this choice optimizes the mixing properties.

When making decisions with simulation-based Bayesian inferences, it is important to check whether the Markov chain has converged to its desired posterior distribution [11]. In practice, we use trace plots of chain values as the most direct method of assessing convergence where we can visually or statistically monitor the marginal paths associated with each parameter. We can observe trace plots and determine whether the chain is mixing well or not. A chain that mixes well travels across its posterior space rapidly, and it can jump from one remote region of the posterior to another in relatively few steps [11]. The initial period that a chain must be run to converge to its stationary distribution is called burn-in time [13]. When making inferences one should exclude these values because they are not sampled from the stationary distribution. In this study we take the burn-in period as the first 5000

iterations [13].

With the standard Random Walk Metropolis algorithm, whether the new candidate q^* is either accepted or not depends on whether it is more likely compared to the previous value of q as regards to the posterior distribution. Once a candidate q^* is rejected, the Delayed Rejection algorithm starts and construct a second stage candidate q^{*2} using the proposal function [13]

$$J_2(q^{*2}|q^{k-1}, q^*) = N(q^{k-1}, \gamma_2^2 \Sigma_k).$$

Here the scale factor γ_2 can be freely chosen and as discussed in [6], a good choice would be $\gamma_2 = 0.01$ which creates a second stage proposal with a small variance.

Then, we construct the probability of accepting second stage candidate α_2, having started at q^{k-1} and rejected q^*:

$$\alpha_2(q^{*2}|q^{k-1}, q^*) = min\left(1, \frac{\pi(q^{*2}|w)J(q^*|q^{*2})[1-\alpha(q^*|q^{*2})]}{\pi(q^{k-1}|w)J(q^*|q^{k-1})[1-\alpha(q^*|q^{k-1})]}\right). \tag{2.2.7}$$

If a second stage candidate is rejected, then a third candidate is not selected. However, another algorithm could be written to select three or more stage candidates if the first two are rejected.

Credible Intervals

The Bayesian set estimates are called credible intervals which is analogous to the concept of confidence intervals use in frequentist approach [11]. A $(1-\alpha)\times 100\%$ equal tail interval can be found using the $(\frac{\alpha}{2})\times 100^{th}$ and $(1-\frac{\alpha}{2})\times 100^{th}$ percentiles of the posterior distribution. In this study we construct the 90% credible intervals using the quantile function in R and compute the 5^{th} and 95^{th} percentiles of the posterior distribution obtained by different Bayesian algorithms.

Chapter 3

QUOTIENT MODEL

In this study, we apply the sensitivity analysis in the problem of modeling tumor growth. The first model we consider was used in [9]. We call it quotient model.

According to Koch et al.(2009), the experimental data of tumor growth in untreated xenograft mice suggested that the tumor growth process proceeds in two distinctively different phases: an initial exponential growth phase followed by a linear growth phase. The tumor growth model produced a monotone growth curve, with a biological support which could explain the curve. On account of having plenty of nutrition and oxygen, the tumor grows exponentially, but this increased tumor itself limits the nutrition and oxygen supply. Thus it slows down the growth of the tumor [9].

There are two main parameters in this unperturbed tumor growth model: λ_0 for the exponential growth phase and λ_1 for the linear growth phase. Initial tumor weight is represented by $x_1(0) = w_0$. The nonlinear tumor growth function [9] was adjusted to obtain the corresponding differential equation,

$$x'(t) = f(x(t)) = \frac{2\lambda_0\lambda_1 x(t)}{\lambda_1 + 2\lambda_0 x(t)}, \quad x(0) = w_0. \tag{3.0.1}$$

In their paper, Koch et al.(2009) stated that this non linear tumor growth model shows a smooth transition between exponential and linear phases. We consider this differential equation (3.0.1) in two different settings: discrete and continuous. We apply both frequentist and Bayesian techniques to analyze the sensitivity of quotient tumor growth model in each setting.

3.1 Discrete Quotient Model

Discrete difference equations allow us to compute the values of a function recursively from a given set of values. We can obtain a difference equation by discretizing a differential equation using the limit definition of the derivative and taking $h = 1$ to get the discrete derivative.

$$x' = \frac{2\lambda_0\lambda_1 x(t)}{\lambda_1 + 2\lambda_0 x(t)}$$

$$\lim_{h\to 0} \frac{x(t+h) - x(t)}{h} = \frac{2\lambda_0\lambda_1 x(t)}{\lambda_1 + 2\lambda_0 x(t)}$$

The term discrete derivative is used to describe the derivative for a function whose domain is discrete. We can define the discrete derivative of a function using forward difference operator Δ as in [8].

Definition 3.1. *Let $x(t)$ be a function of a real or complex variable t, the difference operator Δ is defined by*

$$\Delta x(t) = x(t+1) - x(t). \tag{3.1.1}$$

We apply the forward difference operator to the quotient model to treat it in discrete setup.

$$\Delta x(t) = \frac{2\lambda_0\lambda_1 x(t)}{\lambda_1 + 2\lambda_0 x(t)}$$

$$x(t+1) - x(t) = \frac{2\lambda_0\lambda_1 x(t)}{\lambda_1 + 2\lambda_0 x(t)},$$

$$x(t+1) = \frac{2\lambda_0\lambda_1 x(t)}{\lambda_1 + 2\lambda_0 x(t)} + x(t). \tag{3.1.2}$$

We use this recurrence relation to simulate data with an added noise of a Normal distribution $\epsilon_i \sim N(0, 0.09)$ and with parameter values $\lambda_0 = 0.3$ and $\lambda_1 = 1.5$. Initial tumor weight is taken as $x(0) = 0.5$. The R code for simulating data is given in the appendix.

	Data Points
1	1.091252
2	1.495264
3	1.980548
4	2.878643
5	4.027253
6	4.789723
7	5.762161
8	6.835694
9	7.662608
10	8.686173
11	9.594758
12	9.910701
13	10.555348
14	11.371302
15	12.375328

Table 3.1.1: Simulated data points for the quotient model.

Partial Sum Method for Parameter Estimation

This method was introduced in the paper [1]. In order to determine parameter estimates which fits the data in optimal sense, we first need to obtain pairs of parameter values (λ_0 ,λ_1). We use the partial sum method to break the recurrence relation equation at some index and sum up both sides so that we can get two simultaneous equations for each index. By solving the simultaneous equations we can obtain λ_0 and λ_1. We start with

$$\Delta x(t) = \frac{2\lambda_0\lambda_1 x(t)}{\lambda_1 + 2\lambda_0 x(t)} \tag{3.1.3}$$

$$\Delta x(t)\lambda_1 + \Delta x(t)2\lambda_0 x(t) = 2\lambda_0\lambda_1 x(t),$$

$$[2x(t)]\lambda_0\lambda_1 = [\Delta x(t)]\lambda_1 + [2x(t)\Delta x(t)]\lambda_0.$$

Now we can take the sum of both sides changing the index which determines the break point.

$$[\sum_{t=1}^{index} 2x(t)]\lambda_0\lambda_1 = [\sum_{t=1}^{index} \Delta x(t)]\lambda_1 + [\sum_{t=1}^{index} 2x(t)\Delta x(t)]\lambda_0,$$

and

$$[\sum_{t=index+1}^{15} 2x(t)]\lambda_0\lambda_1 = [\sum_{t=index+1}^{15} \Delta x(t)]\lambda_1 + [\sum_{t=index+1}^{15} 2x(t)\Delta x(t)]\lambda_0.$$

Then, we can get two simultaneous equations of λ_0 and λ_1 as follows

$$A_1\lambda_0\lambda_1 = B_1\lambda_1 + C_1\lambda_0 \tag{3.1.4}$$

$$A_2\lambda_0\lambda_1 = B_2\lambda_1 + C_2\lambda_0, \tag{3.1.5}$$

where $A_1 = \sum_{t=1}^{index} 2x(t)$, $B_1 = \sum_{t=1}^{index} \Delta x(t)$, $C_1 = \sum_{t=1}^{index} 2x(t)\Delta x(t)$,

and $A_2 = \sum_{t=index+1}^{15} 2x(t)$, $B_2 = \sum_{t=index+1}^{15} \Delta x(t)$, $C_2 = \sum_{t=index+1}^{15} 2x(t)\Delta x(t)$.

By changing the index from 1 to 14 and solving Equations (3.1.4) and (3.1.5) we can obtain 14 different estimates for (λ_0, λ_1). For the simulated data from $t = 1$ to $t = 15$ (15 data points) one can calculate the above coefficients for different indices. Simulated data from $t = 1$ to $t = 15$ is shown in Table (3.1.1).

Finding coefficients for the first index ($index = 1$) can be shown as follows:

$\sum_{t=1}^{1} 2x(t) = A_1 = 1$

$\sum_{t=1}^{1} \Delta x(t) = B_1 = 0.5913$

$\sum_{t=1}^{1} 2x(t)\Delta x(t) = C_1 = 0.5913$

$\sum_{t=2}^{15} 2x(t) = A_2 = 173.283$

$\sum_{t=2}^{15} \Delta x(t) = B_2 = 11.2841$

$\sum_{t=2}^{15} 2x(t)\Delta x(t) = C_2 = 142.008$

Since it takes significant amount of time to do it by hand for each index, we use Mathematica to compute the coefficients for each index, and then numerically solve the two simultaneous equations. After we get all the estimated pairs, we use the discrete slope method to choose the optimal value for λ_0 and λ_1.

Discrete Slope Method

We use the algorithm stated in [1] to compute the point estimates. As mentioned in [1], we replace data values with discrete slopes (Δx) in the ordinary least squares (OLS) estimation and compute the minimum sum of squared error to obtain the optimum value λ_{opt} for the model parameters.

$$\lambda_{opt} = min \sum_{t=1}^{15} (f_t(\hat{q}) - \Delta x_t)^2$$

The "Discrete Slope" computes Δx_t as $\Delta x_1 = x_2 - x_1, \Delta x_2 = x_3 - x_2$, etc. For each pair of estimated parameters, $f_t(\hat{q})$ is calculated using Equation (3.0.1) with observations $x(t)$. Results from the Mathematica programming codes [1] are given in Table (3.1.2).

Index	λ_0	λ_1	**SSR**
1	1.95406	0.847765	0.771967
2	0.631306	0.917199	0.801652
3	0.475956	0.954909	0.860548
4	0.831216	0.891264	0.7703774
5	-21.6313	0.815634	0.93499
6	6.1647	0.827348	0.84092
7	-2.10582	0.792559	1.49092
8	-0.589544	0.733435	15.744
9	-0.550349	0.728063	11.3863
10	-0.282471	0.659192	106.911
11	-0.195319	0.606591	272.206
12	2.14456	0.845053	0.776944
13	0.296863	1.06195	1.04185
14	0.173774	1.34597	1.43606

Table 3.1.2: Results of parameter estimation using the discrete slope method.

Optimal $\hat{\lambda}$ pair $(0.831216, 0.891264)$ is at index 4 with the minimum sum of squared residuals (SSR) of 0.7703774. The Figure (3.1.1) displays a good fit for the fitted values and the data points of the model.

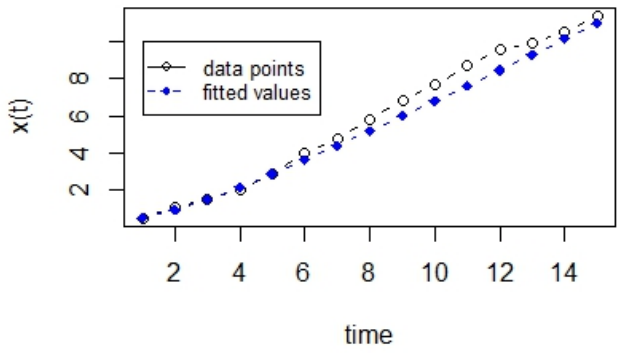
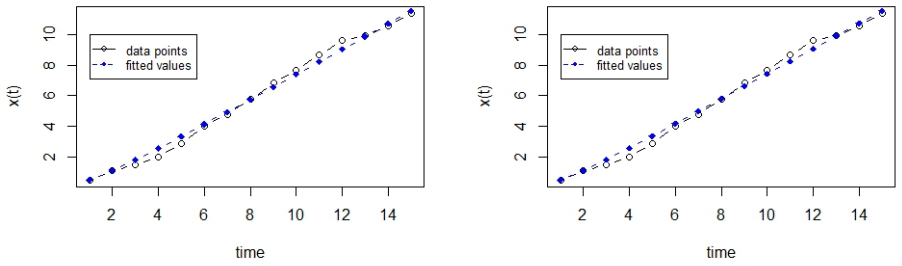


Figure 3.1.1: The plot of data values and fitted values for the discrete quotient model.

Since we get a quite closer SSR values at index 1 and index 12, we plot the fitted curves using the parameter pairs at those indexes.



(a) For index 1 **(b)** For index 12

Figure 3.1.2: The plot of fitted values with the parameter pairs obtained at index 1 and 12 using partial sum method.

Both plots displays good fit. This suggests that we cannot uniquely determine at least one of the parameters. Thus, the model parameters are not practically identifiable. We choose the parameter pair with minimum SSR (at index 4), but one can use the above mentioned parameter pairs and check whether the results will change or not.

Frequentist Approach to Discrete Quotient Model

Since we use $\Delta x(t)$ when computing parameter estimates, we modify the $SS(q)$ function stated in Equation (2.1.2) by replacing data values (w_i) with the difference of data values $\Delta x(t)$. We also update the equation to compute the estimate for the variance of the error terms in the same way:

$$SS(q) = \sum_{t=1}^{n} (f_t(q) - \Delta x(t))^2, \tag{3.1.6}$$

$$\sigma^2 \approx \hat{\sigma}^2 = \frac{1}{n-p} \sum_{t=1}^{n} (f_t(\hat{q}) - \Delta x(t))^2. \tag{3.1.7}$$

After calculating the point estimates $(0.831216, 0.891264)$, we then construct the error variance estimate using Equation (3.1.7) and obtain $\hat{\sigma}^2 = 0.6114875$. The sensitivity matrix $\chi(\hat{q})$ was found by taking the derivative with respective to two parameters and evaluating them at the estimated parameter values.

$$\chi(\hat{q}) = \begin{bmatrix} \frac{\partial \Delta x(1)}{\partial \lambda_0}\Big|_{\lambda_{opt}} & \frac{\partial \Delta x(1)}{\partial \lambda_1}\Big|_{\lambda_{opt}} \\ \vdots & \vdots \\ \frac{\partial \Delta x(14)}{\partial \lambda_0}\Big|_{\lambda_{opt}} & \frac{\partial \Delta x(14)}{\partial \lambda_1}\Big|_{\lambda_{opt}} \end{bmatrix}, \tag{3.1.8}$$

where $\Delta x(t)$ compute by using Equation (3.1.2) and taking the difference of $x(t+1)$ and $x(t)$ from $t = 1, ..., 14$. The Matlab code for computing the sensitivity matrix is listed in the appendix. Then using Equation (2.1.4) we find the initial variance covariance matrix estimate

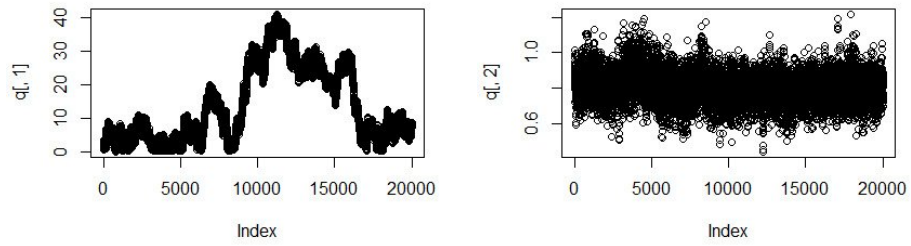
$$\Sigma(\hat{q}) = \begin{bmatrix} 0.3244461 & -0.05151480 \\ -0.0515148 & 0.01482847 \end{bmatrix}.$$

Confidence intervals for the discrete quotient model's parameters are then found using

Equation (2.1.8) with $n = 14$ and $p = 2$.

Bayesian Techniques for the Discrete Quotient Model

First, we use the Random Walk Metropolis algorithm to construct the posterior distribution and then compute the credible intervals. We consider chains of length $M = 20,000$. The R codes for the algorithm is provided in the appendix. We choose $n_s = 0.01$ in Equation (2.2.5) because it gives the narrower credible intervals than other values.

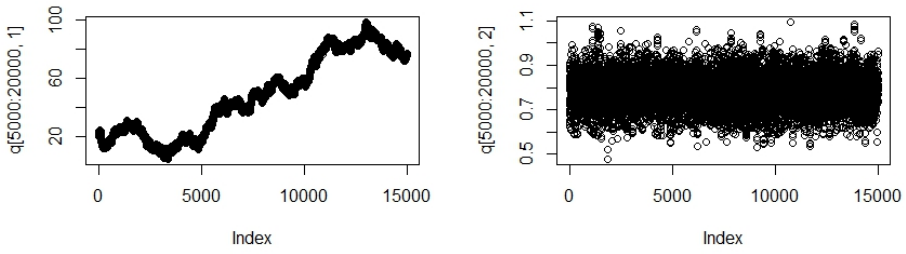


(a) λ_0 **(b)** λ_1

Figure 3.1.3: Trace plots of λ_0 and λ_1 from the Random walk Metropolis algorithm for discrete quotient model.

As shown in Figure (3.1.3) (a), the trace plot for λ_0 displays a chain with a slow mixing. Since one parameter has bad mixing we applied the Delayed Rejection (DR) algorithm to estimate the posterior distribution.

For the discrete quotient model we only use the DR algorithm because when we try to use the Adaptive Metropolis step, it results in singular matrix.

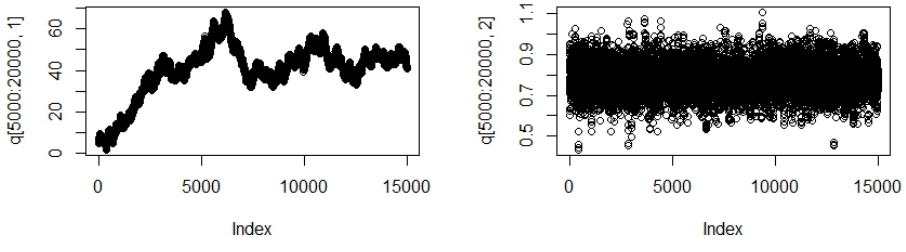


(a) λ_0 **(b)** λ_1

Figure 3.1.4: Trace plots of λ_0 and λ_1 from the Delayed Rejection algorithm for the discrete quotient model.

The trace plot shown in Figure (3.1.4) (a) also displays a poor chain mixing for the parameter λ_0. We then try the DR algorithm with a second stage candidate chosen from a multivariate t distribution.

$$J(q^{*2}|q^{k-1}, q^*) = t_\nu(q^{k-1}, \gamma_2^2 \Sigma_k).$$



(a) λ_0 **(b)** λ_1

Figure 3.1.5: Trace plots of λ_0 and λ_1 from the Delayed Rejection algorithm with a multivariate t distribution for α_2.

As shown in Figure (3.1.5) (a), the trace plot for parameter λ_0 displays an improved

mixing than the previous but not quite good as Figure (3.1.5) (b). The summarized sensitivity analysis for the discrete quotient model is shown in Table (3.1.3).

Parameter	λ_0	λ_1
Point Estimate	0.831216	0.891264
Frequentist approach	(0.05871231, 1.60371969)	(0.7261144, 1.6637677)
RW Metropolis algorithm	(1.192166, 34.318927)	(0.6690686, 0.9445369)
DR algorithm with normal distribution	(11.66595, 87.93513)	(0.6620907, 0.9036670)
DR algorithm with multivariate t distribution	(10.23299, 57.00860)	(0.6641669, 0.9059454)

Table 3.1.3: Results of sensitivity analysis for the discrete quotient model.

The point estimates for both parameters are significantly different from the true values of the parameters: $\lambda_0 = 0.3$ and $\lambda_1 = 1.5$. The 90% credible interval from the Random Walk Metropolis algorithm is very wide for the parameter λ_0, which does not include the point estimate (0.831216) inside the interval. Even though the 90% credible interval obtain from the Delayed Rejection is narrower than the interval from the RW algorithm, it still does not include the point estimate for the parameter λ_0. We try to get an improved credible interval using the multivariate t distribution to obtain α_2 as in Equation (2.2.7), but it still does not give a narrower credible interval which includes the point estimate for the parameter λ_0.

When comparing the two approaches, the frequentist approach gives better results than Bayesian techniques for the discrete quotient model.

3.2 Continuous Quotient Model

In this section, we consider the original version of the model stated in Equation (3.0.1) and treat it as an Ordinary Differential Equation (ODE) and as an initial value problem to solve the ODE. We use the same 15 data points given in Table (3.1.1) for the continuous quotient model. We solve the differential equation (3.0.1) using Matlab codes [6, 7] and obtain the parameter estimations using the least square method. The detailed Matlab code is listed in the appendix. After obtaining the point estimates $(\lambda_0, \lambda_1) = (0.9447, 0.9084)$, we use interpolation to compute fitted values at each time value (i.e. t=1,...15). Then we plot the data points and the fitted values in the same plot and Figure (3.2.1) exhibit a good fit.

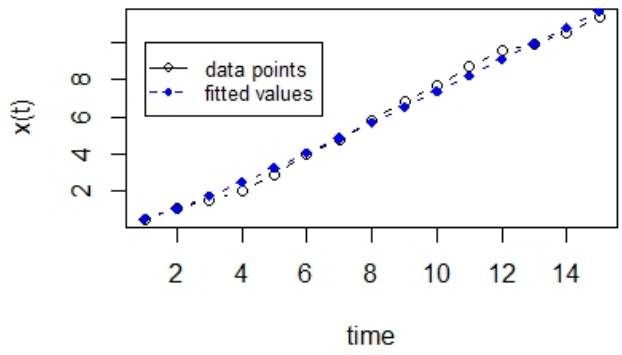


Figure 3.2.1: The plot of data values and fitted values for the continuous quotient model.

Frequentist Approach to Continuous Quotient Model

We obtain an estimate for the variance of the error terms σ^2 using Equation (3.1.7) as $\hat{\sigma}^2 = 0.06039719$. In order to construct the sensitivity matrix for the con-

tinuous quotient model we use the sensitivity equations [13]. As stated in Dr. Ralph Smith's book Uncertainty Quantification (2014), we differentiate the evolution equation $\frac{dx}{dt} = f(x(t), \lambda_0, \lambda_1)$ with respect to the two parameters λ_0 and λ_1, and switch the order of integration to obtain Equations (3.2.1) and (3.2.2).

$$x'(t) = \frac{dx(t)}{dt} = \frac{2\lambda_0\lambda_1 x(t)}{\lambda_1 + 2\lambda_0 x(t)}$$

$$\frac{\partial}{\partial\lambda_0}\Big(\frac{dx(t)}{dt}\Big) = \frac{d}{dt}\Big(\frac{\partial x(t)}{\partial\lambda_0}\Big)$$

$$\frac{dv_0}{dt} = v_0\frac{df}{dx} + \frac{df}{d\lambda_0}, \quad where \quad v_0 = \frac{dx(t)}{d\lambda_0} \tag{3.2.1}$$

$$\frac{dv_1}{dt} = v_1\frac{df}{dx} + \frac{df}{d\lambda_1}, \quad where \quad v_1 = \frac{dx(t)}{d\lambda_1} \tag{3.2.2}$$

Then we solve these two differential equations for v_0 and v_1 numerically using Matlab codes at t=1,...15, and compute the difference between rows of the matrix to obtain the sensitivity matrix shown in (3.1.8). We use Equation (2.1.4) to compute the variance covariance matrix estimate

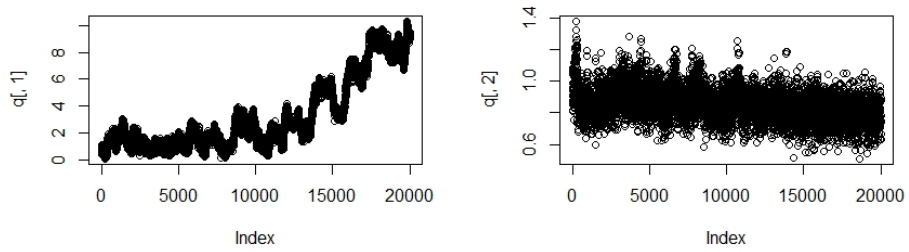
$$\Sigma(\hat{q}) = \begin{bmatrix} 0.1047427 & 0.06871060 \\ 0.0687106 & 0.05052778 \end{bmatrix}.$$

The 90% confidence intervals for the parameters of the continuous quotient model are calculated using Equation (2.1.8). The result is shown in Table (3.2.1).

Bayesian Techniques for Continuous Quotient Model

We consider chains of length M=20,000 and construct the posterior distribution

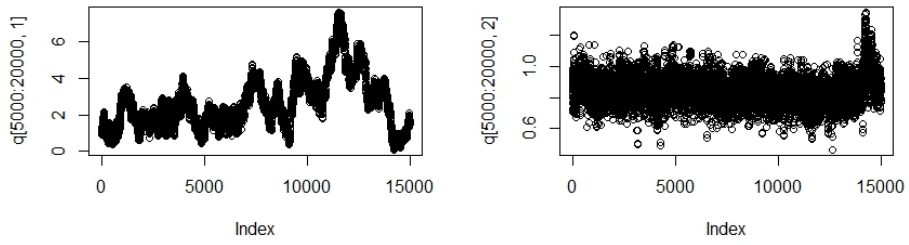
using the Random Walk Metropolis algorithm and obtain the 95% credible intervals.



(a) λ_0 **(b)** λ_1

Figure 3.2.2: Trace plots of λ_0 and λ_1 from the Random Walk Metropolis algorithm for continuous quotient model.

Even though Figure (3.2.2) (b) exhibits good mixing for λ_1, the other parameter λ_0 displays a bad chain mixing. Since it is not very accurate to make inferences with one poor mixing chain, we then try the Delayed Rejection algorithm to construct the posterior distribution.



(a) λ_0 **(b)** λ_1

Figure 3.2.3: Trace plots of λ_0 and λ_1 from the Delayed Rejection algorithm for continuous quotient model.

The trace plot for parameter λ_0 displays slightly improved chain mixing after

using the Delayed Rejection algorithm. Summarized sensitivity analysis for the continuous quotient model is shown in the Table (3.2.1).

Parameter	λ_0	λ_1
Point Estimate	0.9447	0.9084
Frequentist approach	(0.5057741, 1.3836259)	(0.6035442, 1.3473259)
RW Metropolis algorithm	(0.5802713, 8.8497801)	(0.709937, 1.023375)
DR algorithm	(0.7436584, 5.3774320)	(0.7055840, 0.9856311)

Table 3.2.1: Results of sensitivity analysis for the continuous quotient model.

The point estimates for both parameters are significantly different from the true values of the parameters. Both point estimates are included in the 90% credible intervals. The interval width gets narrower when using the DR algorithm rather than using the RW Metropolis algorithm. Both credible intervals constructed using the Bayesian approach are narrower than the frequentist confidence interval. This shows that when the chain converges, the Bayesian approach does give more precise interval estimate than the frequentist approach. For the continuous quotient model, the frequentist approach might be preferable comparing to the Bayesian approach.

Comparing the Results of Discrete and Continuous Quotient Models

Trace plots for λ_1 display good mixing in both discrete and continuous quotient model. But trace plots for the parameter λ_0 exhibit slow mixing compared to λ_1 in both discrete and continuous setting.

The 90% confidence interval obtained by the frequentist approach for the parameter λ_1 in the discrete quotient model has a narrower width than that in the continuous quotient model. On the other hand, we get a wider confidence interval for

the parameter λ_0 in discrete quotient model than that of in the continuous quotient model.

All the 90% credible intervals obtained using the Bayesian techniques for the parameter λ_1 in the discrete quotient model display a narrower width than those in the continuous quotient model. For the parameter λ_0, the continuous quotient model produces narrower 90% credible intervals than the discrete quotient model.

Chapter 4

GOMPERTZ MODEL

Generally, the quantification of tumor growth kinetics follows a sigmoidal shape which is an increasing curve with one inflection point that asymptotically converges to a maximal volume, the carrying capacity [3]. The Gompertz curve or Gompertz function is widely used for tumor growth modeling. This function was first introduced by a British self-educated mathematician Benjamin Gompertz (1779-1865). In 1964, A.K. Laird came up with a modified Gompertz function which was fitted for tumor growth data [10]. Tumor cells can be represented as cellular populations growing in a restricted area or a volume with limited nutrients.

Gompertz curve models tumor size $x(t)$ as follows:

$$x(t) = Kexp(log\tfrac{(x(0)}{K})exp(-\alpha t))$$

$$x(t) = K(\frac{x(0)}{K})^{exp(-\alpha t)}, \tag{4.0.1}$$

where $x(0)$ is the initial tumor size (at $t = 0$) and K is the carrying capacity (maximum size that can be reached with the available nutrients). The second parameter α is related to the proliferation ability of the cells. And $log()$ refers to natural log.

The Gompertz differential equation is:

$$x'(t) = \alpha log(\frac{K}{x(t)})x(t). \tag{4.0.2}$$

Similar to the quotient model, we treat this differential equation in two different cases: discrete setting and continuous setting.

4.1 Discrete Gompertz Model

We apply the forward difference operator (3.1.1) to the Gompertz differential equation to compute $\Delta x(t)$, an approximation of $x'(t)$

$$\Delta x(t) = \alpha log(\frac{K}{x(t)})x(t) \tag{4.1.1}$$

$$x(t+1) - x(t) = \alpha log(\frac{K}{x(t)})x(t)$$

$$x(t+1) = \alpha log(\frac{K}{x(t)})x(t) + x(t).$$

We use this recurrence relation to simulate data with an added noise of a Normal distribution $\epsilon_i \sim N(0, 0.09)$ and with parameter values $\alpha = 0.3$ and $K = 15$. Initial tumor weight is taken as $x(0) = 0.2$. Simulated data points for discrete Gompertz model are shown in Table (4.1.1).

	Data Points
1	0.2000000
2	0.6347079
3	1.4497536
4	2.4332356
5	3.6248777
6	5.3510916
7	6.4603955
8	8.2820181
9	9.6749278
10	10.8624527
11	11.6383742
12	12.4894447
13	13.7209321
14	14.1989947
15	14.5888278

Table 4.1.1: Simulated data points for the Gompertz model.

Using the partial sum method for parameter estimation [1], we obtain the point estimates for the two parameters α and K as $(\hat{\alpha}, \hat{K}) = (0.282224, 15.6497)$. Note that the partial sum method gives good estimates for the parameters because they are quite closed to the true parameter values. Figure (4.1.1) displays a good fit for the data points.

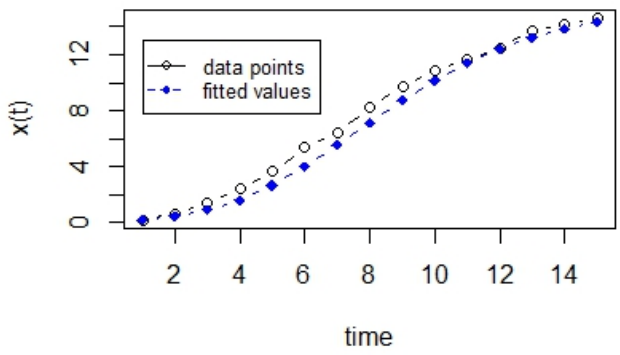


Figure 4.1.1: The plot of data values and fitted values for the discrete Gompertz model.

Frequentist Approach to Discrete Gompertz Model

We construct the $SS(q)$ function as mentioned in Equation (3.1.6) and obtain an estimate for the variance of the error terms using Equation (3.1.7) as $\hat{\sigma}^2 = 0.09933246$, where $n = 14$ and $p = 2$. Then we compute the sensitivity matrix as in Equation (4.1.2) using Matlab codes.

$$\chi(\hat{q}) = \begin{bmatrix} \frac{\partial \Delta x(1)}{\partial \alpha}\Big|_{(\hat{\alpha},\hat{K})} & \frac{\partial \Delta x(1)}{\partial K}\Big|_{(\hat{\alpha},\hat{K})} \\ \vdots & \vdots \\ \frac{\partial \Delta x(14)}{\partial \alpha}\Big|_{(\hat{\alpha},\hat{K})} & \frac{\partial \Delta x(14)}{\partial K}\Big|_{(\hat{\alpha},\hat{K})} \end{bmatrix}. \tag{4.1.2}$$

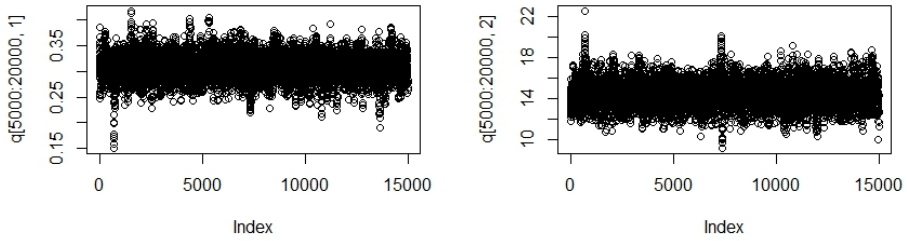
We use Equation (2.1.4) to compute the initial variance covariance matrix $\Sigma(\hat{q})$.

$$\Sigma(\hat{q}) = \begin{bmatrix} 0.0003335633 & -0.006025905 \\ -0.006025905 & 1.467419933 \end{bmatrix}.$$

Then we find the 90% confidence intervals using Equation (2.1.8). The result is listed in Table (4.1.2).

Bayesian Techniques for Discrete Gompertz Model

We use the Random Walk Metropolis algorithm with chains of length M=20,000 and construct the posterior distributions.



(a) α **(b)** K

Figure 4.1.2: Trace plots of α and K from the Random Walk Metropolis algorithm for discrete Gompertz model.

The trace plots for parameters α and K both exhibit chains which mix rapidly. The center of two chains in Figure (4.1.2) appears to be around the values of point estimates $(0.282224, 15.6497)$ for the parameters α and K, respectively. Using the posterior distributions, we construct the 90% credible intervals for both α and K.

Parameter	α	K
Point Estimate	0.282224	15.6497
Frequentist approach	(0.2485003, 0.3159477)	(13.45314, 15.68342)
RW Metropolis algorithm	(0.2681337, 0.3158182)	(13.47489, 15.43927)

Table 4.1.2: Results of sensitivity analysis for the discrete Gompertz model.

In Bayesian approach, the 90% credible intervals computed from the Random Walk Metropolis algorithm are narrower than the confidence intervals obtained from

the frequentist approach. Also, both credible intervals include the point estimates for the two parameters.

4.2 Continuous Gompertz Model

In this section, we consider the Gompertz differential equation (4.0.2) with the analytical solution given in Equation (4.0.1). We simulate data using Equation (4.0.1) with an added noise of a Normal distribution $\epsilon_i \sim N(0, 0.09)$ and with the parameter values $\alpha = 0.3$, $K = 15$, and $x(0) = 0.2$. Then we construct the function $SS(q)$ as stated in Equation (2.1.2). Using a built-in R code for non linear minimization (nlm function) with the $SS(q)$ function and starting parameter values ($\alpha = 2$ and $K = 15$), we get the point estimates $(\hat{\alpha}, \hat{K}) = (0.2939088, 15.2249845)$. Then we plot the data points and fitted values for the model in the same graph as shown in the Figure (4.2.1). The Point estimates obtained give quite good fit for the data points.

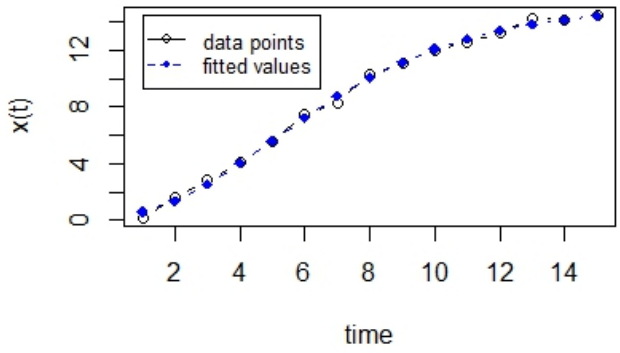


Figure 4.2.1: The plot of data values and fitted values for the continuous Gompertz model.

Frequentist Approach to Continuous Gompertz Model

We compute an estimate of the variance of the error terms $\hat{\sigma}^2 = 0.07372391$ using Equation (2.1.5) with $n = 15$ and $p = 2$. The sensitivity matrix is calculated using Equation (2.1.3) and we get

$$\chi(\hat{q}) = \begin{bmatrix} \frac{\partial x(1)}{\partial \alpha}\Big|_{(\hat{\alpha},\hat{K})} & \frac{\partial x(1)}{\partial K}\Big|_{(\hat{\alpha},\hat{K})} \\ \vdots & \vdots \\ \frac{\partial x(14)}{\partial \alpha}\Big|_{(\hat{\alpha},\hat{K})} & \frac{\partial x(14)}{\partial K}\Big|_{(\hat{\alpha},\hat{K})} \end{bmatrix}, \tag{4.2.1}$$

where

$$\frac{\partial x}{\partial \alpha} = (-)K * t * e^{-t\alpha} log(\frac{x(0)}{K}) * \Big(\frac{x(0)}{K}\Big)^{e^{-t\alpha}}, \tag{4.2.2}$$

and

$$\frac{\partial x}{\partial K} = e^{-t\alpha} * e^{t\alpha - 1} * \Big(\frac{x(0)}{K}\Big)^{e^{-t\alpha}}. \tag{4.2.3}$$

Then we construct the initial estimated variance covariance matrix $\Sigma(\hat{q})$ using Equation (2.1.4).

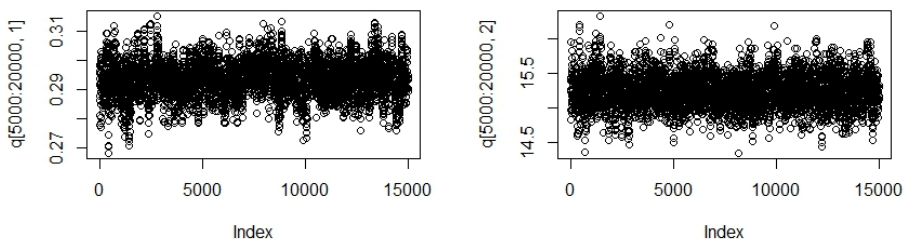
$$\Sigma(\hat{q}) = \begin{bmatrix} 3.581765e^{-05} & -0.002976695 \\ -0.002976695 & 0.331307301 \end{bmatrix}.$$

The 90% confidence intervals are calculated using Equation (2.1.8).

Bayesian Techniques for the Continuous Gompertz Model

We use the RW Metropolis algorithm to obtain posterior distributions and then construct the 90% credible intervals for two parameters.

The trace plots shown in Figure (4.2.2) display chains with good mixing for both parameters α and K. These plots also appear to fluctuate around the estimated parameters. We then compute the 90% credible intervals from the posterior distributions obtained by the Random Walk Metropolis algorithm.



(a) α **(b)** K

Figure 4.2.2: Trace plots of α and K from the Random Walk Metropolis algorithm for continuous Gompertz model.

Parameter	α	K
Point Estimate	0.2939088	15.2249845
Frequentist approach	(0.2833101, 0.3045075)	(14.20565,16.24432)
RW Metropolis algorithm	(0.284014, 0.303288)	(14.87758, 15.60539)

Table 4.2.1: Results of sensitivity analysis for the continuous Gompertz model.

The 90% credible intervals obtained by the RW Metropolis algorithm include the point estimates for both parameters. The width of the RW Metropolis intervals for both parameters are narrower than those obtained by the frequentist approach.

Comparing the Results of Gompertz and Quotient Models

In the continuous Gompertz model, we have a closed form solution for the differential equation. On the other hand, in the continuous quotient model, we have to use numerical methods to solve the differential equation. As a result, the point estimates obtained for parameters of the continuous Gompertz model are much better than those in the continuous quotient model.

Trace plots obtained in Gompertz model display chains with much better mixing than those in the quotient model. Since we obtain better mixing in posterior distributions in the case of Gompertz model, the resulting credible intervals include the point estimates of the parameters and the Bayesian credible intervals are narrower than the frequentist confidence intervals. In the quotient model, since the chains display poor mixing, which suggests non-convergence, the frequentist approach might be preferable than the Bayesian approach.

4.3 Continuous Gompertz Model with Non-Constant Variance Noise

Even though we simulate data with an added noise of constant variance as in Section 4.2, the variance in noise may not be constant when it comes to real data. Hence we try to modify our study of continuous Gompertz model using simulated data which contain non-constant noise component of a Normal distributions $\epsilon_i \sim N(0, \frac{t}{10} * 0.09)$, where the variance changes with respect to time values. Then we apply both frequentist and Bayesian techniques to analyze sensitivity of the parameters.

Frequentist Approach

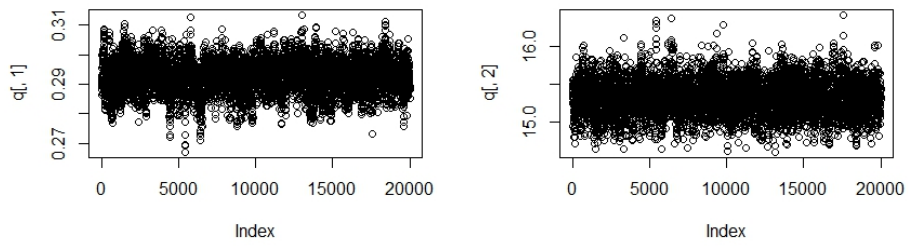
We obtain the point estimates $(\hat{\alpha}, \hat{K}) = (0.2925534, 15.3141003)$ by minimizing the $SS(q)$ function in Equation (2.1.2) and apply the frequentist techniques as mentioned in Section 4.2 to construct the initial estimated variance covariance matrix $\Sigma(\hat{q})$.

$$\Sigma(\hat{q}) = \begin{bmatrix} 3.203252e^{-05} & -0.002710358 \\ -0.002710358 & 0.305919100 \end{bmatrix}.$$

The 90% confidence intervals for two parameters are calculated using Equation (2.1.8).

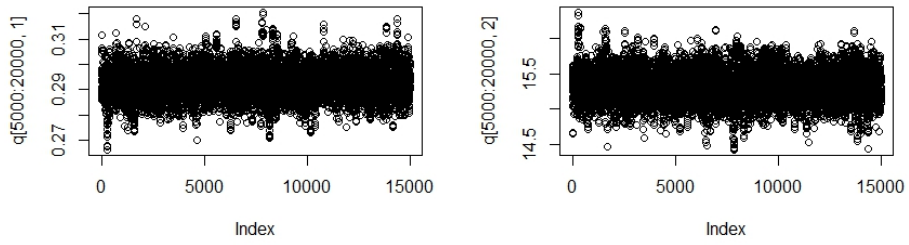
Bayesian Approach

We use both the Random Walk Metropolis algorithm and the Delayed Rejection Adoptive Metropolis algorithm to construct posterior distributions and then obtain the 90% credible intervals for both parameters.



(a) α **(b)** K

Figure 4.3.1: Trace plots of α and K from the Random Walk Metropolis algorithm for continuous Gompertz model with non-constant variance noise.



(a) α **(b)** K

Figure 4.3.2: Trace plots of α and K from the DRAM algorithm for continuous Gompertz model with non-constant variance noise.

The trace plots for α and K from the Random Walk Metropolis algorithm and the DRAM algorithm display chains with good mixing. We construct the 90% credible intervals for the parameters α and K using posterior distributions obtained by both the Random Walk Metropolis algorithm and the DRAM algorithm. The results are given in Table (4.3.1).

Parameter	α	K
Point Estimate	0.2925534	15.3141003
Frequentist approach	(0.2825304, 0.3025764)	(14.3346, 16.2936)
RW Metropolis algorithm	(0.2837232, 0.3014521)	(14.98159, 15.67266)
DRAM algorithm	(0.2839949, 0.3011581)	(14.97724, 15.67279)

Table 4.3.1: Results of sensitivity analysis for the continuous Gompertz model with non-constant variance noise.

Both 90% credible and confidence intervals include point estimates and the Bayesian credible intervals are narrower than the confidence intervals obtain using the frequentist technique. The goal of this section is to show that a narrower credible intervals are obtained from the DRAM algorithm since in the DRAM algorithm the initial variance-covariance matrix gets updated based on previous accepted chain values. As shown in the Table (4.3.1), the 90% credible intervals obtained from the DRAM algorithm are narrower than the credible intervals obtained from the Random Walk Metropolis algorithm.

4.4 Another Form of Discrete Gompertz Tumor Model

The Gompertz differential equation also has the following form [1]:

$$x'(t) = (a - blnx(t))x(t), \tag{4.4.1}$$

where we can redefine the parameters of the Equation (4.0.2) as $a = \alpha lnK$ and $b = \alpha$. As mentioned in the paper [1], we divide each side of the Equation (4.4.1) by $x(t)$ and then using the substitution $u(t) = lnx(t)$ and obtain the following differential equation

$$u'(t) = a - blnu(t).$$

Then, using the forward difference operator we obtain the discrete unperturbed Gompertz tumor growth model.

$$u(t) = lnx(t) \tag{4.4.2a}$$

$$\Delta u(t) = a - blnu(t) \tag{4.4.2b}$$

We use the same data points mentioned in the Table (4.1.1). Figure (4.4.1) displays the plot of $u(t)$ versus time for the model. We use the partial sum method to obtain the point estimates for the model parameters a and b as $(0.68922, 0.254632)$.

Using the relations $a = \alpha lnK$ and $b = \alpha$ we calculate the values of α and K as $(0.254632, 14.98020573)$. This gives quite close values for the point estimates obtained in section 4.1. We then plot the data points and the fitted values in the same graph as in Figure (4.4.1) and it exhibits a good fit.

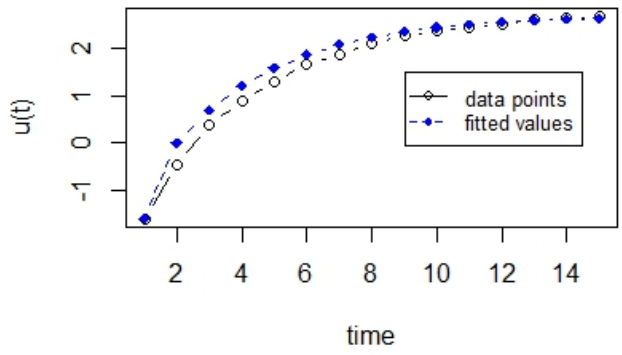


Figure 4.4.1: The plot of data values and fitted values for the unperturbed discrete Gompertz model.

Chapter 5

THE PERTURBED TUMOR MODEL

Due to the effect of anticancer drug, it is assumed that the speed of growth of tumor cells decreases with a rate depending on the drug concentration in plasma [9]. Thus a loss term is added to the unperturbed Gompertz tumor growth model [2]. In this chapter we consider the perturbed tumor model described in the paper [2]. The perturbed tumor model is given by,

$$u(t) = lnx_1(t), \tag{5.0.1a}$$

$$\Delta u(t) = a - bu(t) - k_2c(t), \qquad x_1(0) = w_0, \quad u(0) = ln(w_0) \tag{5.0.1b}$$

$$\Delta x_2(t) = k_2c(t)x_1(t) - k_1x_2(t), \qquad x_2(0) = 0, \tag{5.0.1c}$$

$$w(t) = x_1(t) + x_2(t). \tag{5.0.1d}$$

Parameters of the Gompertz differential equation (4.0.2) is redefined as $a = \alpha ln(K)$ and $b = \alpha$. In addition to the two parameters in unperturbed tumor growth model a and b, two drug related parameters are introduced: k_1 for transit rate between the compartments of the non-proliferating cells and k_2 for the potency factor of the drug [9]. In this model, $x_1(t)$ refers to the tumor weight of proliferating cells that are not affected by the drug which contribute to the growth of tumor. Tumor weight of non-proliferating cells that is affected by drug action is modeled using $x_2(t)$. $c(t)$ represents the concentration of the drug in plasma. The total tumor weight $w(t)$ is the sum of both proliferating and non-proliferating cells.

We use clinical data from Koch et al.(2009) and apply both frequentist and Bayesian techniques to analyze the sensitivity of perturbed tumor model. Total of 30 data points are used in this study. For the first fifteen data points no drug is given, and for the next fifteen data points drug is given and total tumor weight $w(t)$

is recorded.

Frequentist Approach to Perturbed Tumor Model

We first consider the nine data points where no drug is given and use Equations (5.0.1a) and (5.0.1b) to estimate parameters a and b using a built-in R function for non linear minimization. Then we construct

$$\chi(\hat{q}) = \begin{bmatrix} \frac{\partial w(1)}{\partial a}\Big|_{(\hat{a},\hat{b})} & \frac{\partial w(1)}{\partial b}\Big|_{(\hat{a},\hat{b})} \\ \vdots & \vdots \\ \frac{\partial w(15)}{\partial a}\Big|_{(\hat{a},\hat{b})} & \frac{\partial w(15)}{\partial b}\Big|_{(\hat{a},\hat{b})} \end{bmatrix}, \tag{5.0.2}$$

and add two zero valued columns because there are no parameters k_1 and k_2 with the absence of drug concentration $c(t)$. Then we use the perturbed model for the fifteen data points with the presence of drug to obtain parameter estimation for a, b, k_1, k_2 and construct the matrix,

$$\chi(\hat{q}) = \begin{bmatrix} \frac{\partial w(16)}{\partial a}\Big|_{(\hat{a},\hat{b},\hat{k}_1,\hat{k}_2)} & \frac{\partial w(16)}{\partial b}\Big|_{(\hat{a},\hat{b},\hat{k}_1,\hat{k}_2)} & \frac{\partial w(16)}{\partial k_1}\Big|_{(\hat{a},\hat{b},\hat{k}_1,\hat{k}_2)} & \frac{\partial w(16)}{\partial k_2}\Big|_{(\hat{a},\hat{b},\hat{k}_1,\hat{k}_2)} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial w(30)}{\partial a}\Big|_{(\hat{a},\hat{b},\hat{k}_1,\hat{k}_2)} & \frac{\partial w(30)}{\partial b}\Big|_{(\hat{a},\hat{b},\hat{k}_1,\hat{k}_2)} & \frac{\partial w(30)}{\partial k_1}\Big|_{(\hat{a},\hat{b},\hat{k}_1,\hat{k}_2)} & \frac{\partial w(30)}{\partial k_2}\Big|_{(\hat{a},\hat{b},\hat{k}_1,\hat{k}_2)} \end{bmatrix}. \tag{5.0.3}$$

We combine these two matrices to obtain the 30×4 sensitivity matrix. We use all 30 data points to construct the function $SS(q)$ as in Equation (2.1.2) and using non linear minimization we obtain the point estimates for all four parameters $(\hat{a}, \hat{b}, \hat{k}_1, \hat{k}_2) = (0.07018510, 0.02331440, 0.88257986, 0.01427439)$. The Figure (5.0.1) displays a good fit for the data points and the fitted values.

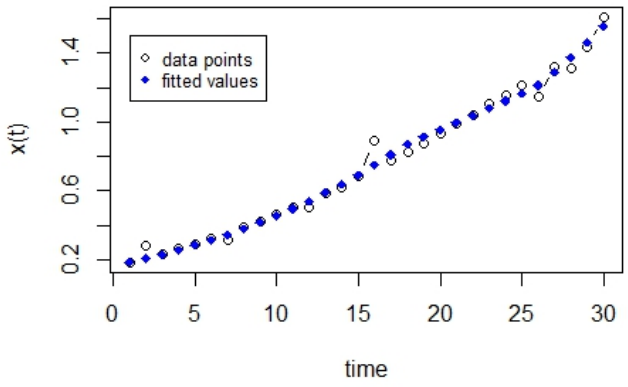


Figure 5.0.1: The plot of data values and fitted values for the perturbed tumor model.

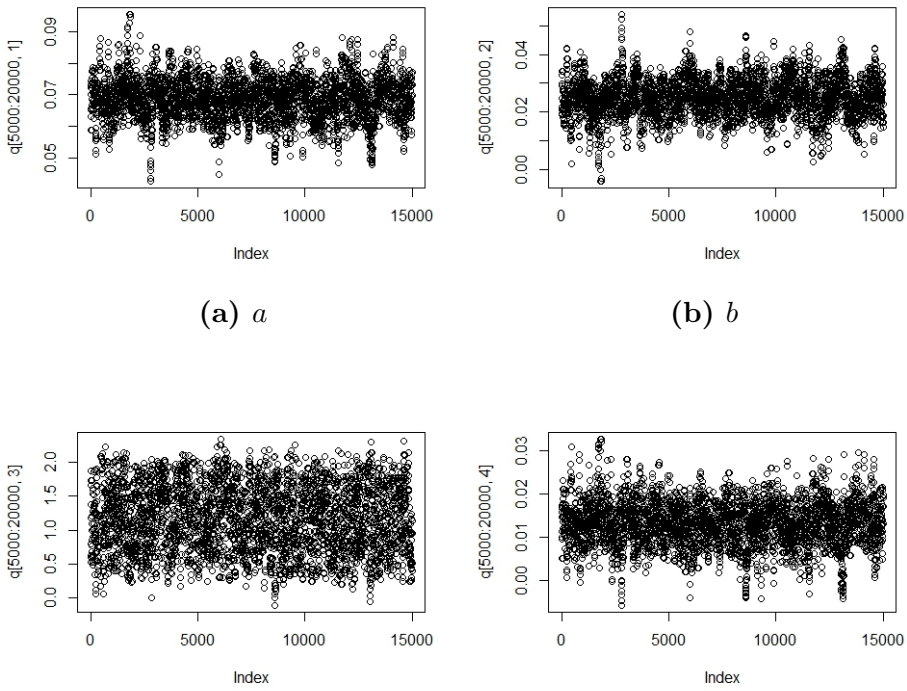
The initial estimated variance covariance matrix $\Sigma(\hat{q})$ is obtained using Equation (2.1.4) and the estimate of the variance of the error terms is $\hat{\sigma}^2 = 0.001912274$.

$$\Sigma(\hat{q}) = \begin{bmatrix} 3.749842e^{-05} & -3.221253e^{-05} & -0.001749090 & 3.260231e^{-05} \\ -3.221253e^{-05} & 3.382281e^{-05} & 0.001676255 & -2.881328e^{-05} \\ -1.749090e^{-03} & 1.676255e^{-03} & 0.307592154 & -1.813584e^{-03} \\ 3.260231e^{05} & -2.881328e^{-05} & -0.001813584 & 2.934787e^{-05} \end{bmatrix}.$$

Then we calculate the 90% confidence intervals using Equation (2.1.8).

Bayesian Approach to Perturbed Tumor Model

We use the Random Walk Metropolis algorithm to construct posterior distributions and then calculate the 90% credible intervals for all four parameters in the perturbed model.

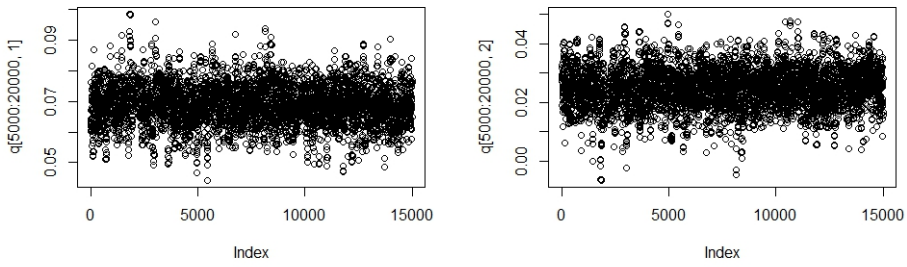


(a) a **(b)** b

(c) k_1 **(d)** k_2

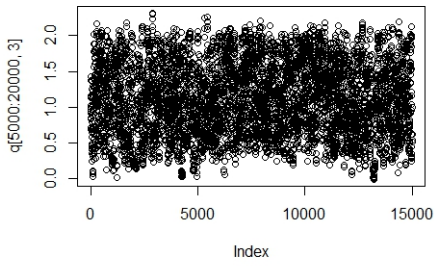
Figure 5.0.2: Trace plots of a, b, k_1 and k_2 from the Random Walk Metropolis algorithm for perturbed tumor model.

The trace plots for all parameters exhibit chains with good mixing. We try to improve the mixing by applying the Delayed Rejection Adoptive Metropolis algorithm to construct posterior distributions.

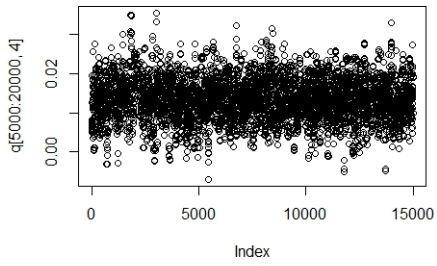


(a) a

(b) b



(c) k_1



(d) k_2

Figure 5.0.3: Trace plots of a, b, k_1 and k_2 from the Delayed Rejection Adoptive Metropolis algorithm for perturbed tumor model.

The trace plots shown in Figure (5.0.3) display improved chain mixing for all four parameters. These plots appear to be fluctuating around the point estimates. We then compute 90% credible intervals from the posterior distributions obtained by both the Random Walk Metropolis algorithm and the Delayed Rejection Adoptive Metropolis algorithm. All the confidence intervals and credible intervals for the perturbed model are listed in Table (5.0.1).

Parameter	a	b	k_1	k_2
Point Estimate	0.07018510	0.02331440	0.88257986	0.01427439
Frequentist approach	(0.05974059, 0.08062961)	(0.01339497, 0.03323383)	(-0.06337274, 1.82853246)	(0.00503443, 0.023514349)
RWM algorithm	(0.05893384, 0.07948058)	(0.01352790, 0.0348804)	(0.3824376, 1.9255008)	(0.005320269, 0.021947253)
DRAM algorithm	(0.058597, 0.07901574)	(0.01393455, 0.03332443)	(0.380086, 1.9215447)	(0.005362433, 0.022109217)

Table 5.0.1: Results of sensitivity analysis for the perturbed model.

The 90% confidence intervals contain the point estimates of all four parameters. For the parameter k_1, the frequentist confidence interval contains zero which shows that k_1 is not statistically significant. On the other hand, the 90% credible intervals obtained by the RW Metropolis algorithm and the DRAM algorithm do not contain zero. By using Bayesian algorithm to construct credible intervals, it improves the results as the credible intervals for all four parameters are narrower than the confidence intervals obtained from the frequentist approach.

Chapter 6
CONCLUSION AND FUTURE WORK

Sensitivity analysis can assess the uncertainty of model parameters, which occurs due to noise and measurement errors. In this study, we discuss the sensitivity of model parameters for previously developed tumor models by constructing confidence intervals and credible intervals using the frequentist approach and the Bayesian techniques, respectively.

In chapter 3, we analyze the sensitivity of the parameters of the quotient model. Trace plots for λ_0 obtained in both discrete and continuous settings display chains with slow mixing. The parameter λ_1 exhibits chains with good mixing in both discrete and continuous settings. We obtain an improved confidence interval and an improved credible interval for the parameter λ_1 in the discrete quotient model than that of in the continuous quotient model. Since we get chains with slow mixing for the posterior distributions for the parameter λ_0 in the discrete setting, the credible intervals did not include the point estimates. For the parameter λ_1, we obtain improved credible intervals by using Bayesian techniques in both discrete and continuous quotient models. In both discrete and continuous settings, the point estimates show a significant difference than their true values. Thus, the frequentist approach gives better results in the quotient model.

In chapter 4, we analyze the sensitivity of parameters of the Gompertz model. Trace plots for both parameters display chains with very good mixing in both discrete and continuous settings. In both discrete Gompertz model and continuous Gompertz model, the credible intervals include the point estimates for both parameters and the width of the intervals obtained from the Bayesian techniques are narrower than that of the frequentist approach.

We obtain improved credible intervals when using the DRAM algorithm for

the continuous Gompertz model with non-constant variance noise component. When comparing the trace plots for parameters in the quotient model with those in the Gompertz model, trace plots obtained in the Gompertz model exhibits improved mixing than those in the quotient model.

In chapter 5, we analyze the perturbed tumor model. Trace plots obtained from the DRAM algorithm display chains with improved mixing than the trace plots obtained from the Random Walk Metropolis algorithm. We obtain narrower credible intervals from Bayesian techniques compared to the frequentist approach for all four parameters.

Future Work

In this study we add the noise from a symmetric Normal distribution. One can add noise from a skewed distribution and study the uncertainty of the parameters of these models. Different data sets could be obtained by changing the initial parameter values and the variance in the simulation and could use those data sets to check whether the results would change or not. Also, the Bayesian algorithms could be constructed with a skewed proposal distribution. These tumor models could be used to study the geometry of parameter space. Also one can analyze the parameter identifiability for the models.

Bibliography

[1] Atici F. M., Atici, M., Belcher, M. and Marshall, D. (2017). A new approach for modeling with discrete fractional Equations, *Fundamenta Informaticae*, 151(1-4), 313-324.

[2] Atici, F. M., Atici, M., Nguyen, N., Zhoroev, T. and Koch, G. (2019). A study on discrete and discrete fractional pharmacokinetics-pharmacodynamics models for tumor growth and anti-cancer effects. (To be published)

[3] Benzekry, S., Lamont, C., Beheshti, A., Tracz, A., Ebos, J. M. L., Hlatky, L. and Hahnfeldt, P. (2014). Classical mathematical models for description and prediction of experimental tumor growth, *PLoS Computational Biology*, 10(8): e1003800, URL https://doi.org/10.1371/journal.pcbi.1003800.

[4] Brooks, S. P. and Roberts, G. O. (1998). Convergence Assessment Techniques for Markov Chain Monte Carlo, *Statistics and Computing*, 8(4), 319-335.

[5] Gelman, A., Carlin, J. B., Stern, H. S. and Rubin, D. B. (2004). *Bayesian Data Analysis*, Second Edition, Boca Raton, FL: Chapman and Hall/CRC.

[6] Haario, H., Laine, M., Mira, A. and Saksman, E. (2006). DRAM: Efficient Adaptive MCMC, *Statistics and Computing*, 16(4), 339-354.

[7] Jeffrey, C. L, James, A. R, Margaret, H. W, and Paul, E. W. (1998). Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions, *SIAM Journal of Optimization*, 9(1), 112-147.

[8] Kelly, W. G. and Peterson, A. C. (2001). *Difference Equations: An Introduction with Applications*, San Diego, CA: Academic Press.

[9] Koch, G., Walz, A., Lahu, G., and Schropp, J. (2009). Modeling of Tumor Growth and Anticancer Effects of Combination Therapy, *Journal of Pharmacokinetics and Pharmacodynamics*, 36(2), 179-197.

[10] Laird, A. K. (1964). Dynamics of Tumor Growth, *British Journal of Cancer*, 13(3), 490-502.

[11] SAS Institute, Inc. (2009). SAS/STAT ® 9.2 User's Guide, Second Edition. Cary, NC: SAS Institute Inc.

[12] Simeoni, M., Magni, P., Cammia, C., Nicolao, D. G, Croci, V., Pesenti, E., Germani, M., Poggesi, I., and Rocchetti, M. (2004). Predictive pharmacokinetic phamrmacodynamic modeling of tumor growth kinetics in xenograft models after administration of anticancer agents, *Cancer Research*, 64, 1094–1101.

[13] Smith, R. C. (2014). *Uncertainty Quantification: Theory, Implementation, and Applications*, Philadelphia: SIAM.

APPENDIX

A.1 R Codes

This section gives R codes written for implementing the methods in this thesis. The array z gives the simulated data points.

```
data<-array(0,15)
z=data
set.seed(12345)
z[1]=w0
# w0 is the initial tumor weight
for (i in 1:14)
{z[i+1]= ((2*L0*L1*z[i])/(L1+2*L0*z[i]) + z[i])+rnorm
   (1,0,0.3)}
# Here L0 and L1 should be initial parameter guesses
# Following codes are for constructing SS(q) function
    in discrete setting
ds<-array(0,14)
for(i in 1:14)
{ds[i]=z[i+1]-z[i]}
ssq=function(x)
{fitteddata<- array(0,15)
fittedslope <-array(0,14)
fitteddata[1]=0.5
for(i in 1:14)
{fitteddata[i+1]= ((2*x[1]*x[2]*fitteddata[i])/(x
   [2]+2*x[1]*fitteddata[i]))+fitteddata[i]
```

```
fittedslope[i]=fitteddata[i+1]-fitteddata[i]}
sum((ds - fittedslope)^2)}
ss=array(0,20000)
s2=ss
ss[1]= ssq(c(L0 ,L1)) # L0,L1 are the point
    estimates obtained from either Mathematica codes
    for discrete setting or built-in nlm code in R.
s2[1]=ss[1]/12
chimat<- matrix(c(y,nrow=14,ncol=2)
# y is the result obtain for chi matrix by matlab
    code and written as an array
A<-t(chimat)%*%chimat
det(A)
solve(A)
k<-solve(A)
covmat<-k*s2[1]
# If it is for the continuous setting, we use the
    code
nlm(ssq,c(L0,L1))
#where L0 and L1 are the parameter guesses.
# The Random Walk Metropolis algorithm begins from
    here
M=20000
q=matrix(0,20000,2,byrow=TRUE)
q[1,]= c(L0 ,L1)
for(j in 2:20000)
```

```
{qstar=rmvnorm(1,q[j-1,],covmat)
ualpha <- runif(1,0,1)
ssqstar=ssq(qstar)
alpha<-min(1,exp(-(ssqstar-ss[j-1])/(2*s2[j-1])))
if(ualpha<alpha){
q[j,]=qstar
ss[j]=ssqstar}
else{
q[j,]=q[j-1,]
ss[j]=ss[j-1]}
aval=0.5*(0.01+14)
bval=0.5*((0.01*s2[1])+ss[j] )
s2[j]=rinvgamma(n=1,aval,scale= bval)
}
# R codes for the DRAM algorithm are shown below.
M=20000
q=matrix(0,20000,2,byrow=TRUE)
q[1,]= c(L0 ,L1)
gamma = 0.01
sp=((2.38)^2)/2
for(j in 2:20000)
{qstar=rmvnorm(1,q[j-1,],covmat)
ualpha <- runif(1,0,1)
ssqstar=ssq(qstar)
alpha<-min(1,exp(-(ssqstar-ss[j-1])/(2*s2[j-1])))
if(ualpha<alpha){
```

```
q[j,]=qstar
ss[j]=ssqstar}
else
{
qstar2=rmvnorm(1,q[j-1,],gamma*covmat)
ualpha <- runif(1,0,1)
ssqstar2=ssq(qstar2)
r1= t(qstar-qstar2)%*%(qstar-qstar2)
r2= t(qstar-q[j-1,])%*%(qstar-q[j-1,])
vinv=solve(covmat)
J=exp(-vinv*(r1-r2)/2)
alph_star<- min(1,exp(-(ssqstar-ssqstar2)/(2*s2[j-1]))
   ))
alpha2<-min(1,exp(-(ssqstar2-ss[j-1])/(2*s2[j-1]))*J
   *((1-alph_star)/(1-alpha)))
# If the second candidate is chosen from a
   multivariate t distribution, we use following
   codes to construct qstar2
qstar2= rmvt (1, gamma*sig ,df=14,delta=q[j-1,])
ualpha <- runif(1,0,1)
ssqstar2=ssq(qstar2)
r1=((1/14)* (qstar-q[j-1,])%*%(solve(sig))%*%t (qstar
   -q[j-1,]))
r2= ((1/14)* (qstar- qstar2)%*%(solve(sig))%*%t(qstar
   - qstar2))
J=((1+r1)/(1+r2))^8
```

```
if(ualpha<alpha2){
q[j,]=qstar2
ss[j]=ssqstar2}
else{
q[j,]=q[j-1,]
ss[j]=ss[j-1]}
}
aval=0.5*(0.01+14)
bval=0.5*((0.01*s2[1])+ss[j] )
s2[j]=rinvgamma(n=1,aval,scale= bval)
if (j%%100==1){
covmat=sp*(cov(q[1:j,]))}}
```

A.2 Matlab Codes

In this section,we use the Matlab codes from [6] to construct the sensitivity matrix for the continuous quotient model and modify the codes to obtain parameter estimations [7].

Parameter estimation for continuous quotient model

```
function [params,J] = trial_func_main_fminsearch_QC

t = 0:1:15;
u1_dat = [..];% dataset
param_guess = [L0g,L1g];% 2 parameter guesses L0gand L1g
```

```
[params,J] = trial_func_min(param_guess,t,u1_dat);
params
u = trial_func(params,t);
u1 = u(:,1);
J
function [params,J] = trial_func_min(param_guess,t,u1_dat)
options = optimset('TolFun',1e-12,'TolX',1e-12,'MaxFunEvals
    ',1e5','MaxIter',1e5);
[params,J] = fminsearch(@trial_func_least,param_guess,options
    ,t,u1_dat);
function J = trial_func_least(param_guess,t,u1_dat)
u = trial_func(param_guess,t);
u1 = u(:,1);
diffJ = (abs(u1 - u1_dat)).^2;
J = sum(diffJ);
function u = trial_func(param_guess,t)
ic = 0.5;
options = odeset('Stats','on','RelTol',1e-4,'AbsTol',1e-4);
[T1,u] = ode15s(@trial_func_ode,t,ic,options,param_guess);
function dy = trial_func_ode(t,y,param_guess)
k1 = param_guess(1);
k2 = param_guess(2);
dy = (2*k1*k2*y(1))/(k2+(2*k1*y(1)));
```

Sensitivity matrix for continuous quotient model

```
function sensitivitymatrix=quotientcontsensitivitycode
tic
```

```
t1 = 0:1:15;
t2 = 0:1:15;
L0 = L0sub;
L1 = L1sub;
params=[L0;L1];%L0sub and L1 sub are the point estimates for
    the model parameters.
ic = w0;%this is the initial tumor weight;
ic2 = [0,0];
options = odeset('Stats','on','RelTol',1e-4,'AbsTol',1e-4);
options2 = odeset('Stats','on','RelTol',1e-4,'AbsTol',[1e-4,1
    e-4]);
[T1,u] = ode15s(@quotientcont_ode,t1,ic,options,params);
[T4,sensitivities] = ode15s(@sensitivities_ode,t2,ic2,
    options2,params,t1,u);
sensitivitymatrix = [sensitivities(:,1),sensitivities(:,2)];
function dy = quotientcont_ode(t,y,params)
L0 = params(1);
L1 = params(2);
dy = (2*L0*L1*y(1))/(L1+(2*L0*y(1)));
function dy = sensitivities_ode(t,y,params,tt,u)
L0 = params(1);
L1 = params(2);
u1 = u(:,1);
u11 = interp1(tt,u1,t,'cubic');
dy = [(2*L0*L1^2*y(1)+2*L1^2*u11+4*L0*L1*u11-4*L0*L1*u11^2)/(
    L1+2*L0*u11)^2;(2*L0*L1^2*y(2)+4*L0^2*u11^2)/(L1+2*L0*u11)
```

```
^2];
```

Sensitivity matrix in discrete setting

We use following Matlab codes to obtain the $\chi(q)$ matrix explained in Equation (2.1.3) for the discrete setting of both quotient model and Gompertz model. The code shown below is for the discrete quotient model, but one can modify it according to the model. Also we modified it accordingly to obtain the sensitivity matrix for the perturbed tumor model as well.

```
clear; clc;
syms L0 L1 x;
x(1) = w0;
% here w0 is the initial tumor weight
L0_sub = L0sub;
L1_sub =L1sub;
% L0sub and L1sub are the poit estimates of model parameters
answer = [];
for i = 1:14
      x(i+1) = 2*L0*L1*x(i)/(L1+2*L0*x(i)) + x(i);
    delx(i)=x(i+1)-x(i);
     x_L0 = diff(delx(i), L0);
     x_L1 = diff(delx(i), L1);
     answer(1, i) = double(subs(x_L0, {L0, L1}, [L0_sub,
        L1_sub]));
     answer(2, i) = double(subs(x_L1, {L0, L1}, [L0_sub,
        L1_sub]));
end
disp(answer);
```