2009

# A SEQUENTIAL ALGORITHM TO IDENTIFY THE MIXING ENDPOINTS IN LIQUIDS IN PHARMACEUTICAL APPLICATIONS

Akriti Saxena
*Virginia Commonwealth University*

Virginia Commonwealth University
School of Medicine

This is to certify that the thesis prepared by Akriti Saxena entitled A SEQUENTIAL ALGORITHM TO IDENTIFY THE MIXING ENDPOINTS IN LIQUIDS IN PHARMACEUTICAL APPLICATIONS has been approved by his or her committee as satisfactory completion of the thesis or dissertation requirement for the degree of Master of Science

V.Ramakrishnan, Ph.D. Virginia Commonwealth University

Charles W. Kish, Ph.D., Wyeth Consumer Healthcare

Carol Summitt, Ph.D., Wyeth Consumer Healthcare

Jessica McKinney Ketchum, Ph.D., Virginia Commonwealth University

Shumei S. Sun, Ph.D., Chair, Department of Biostatistics, Virginia Commonwealth University

Jerome F. Strauss, III, M.D, Ph.D., Dean of the School of Medicine, Virginia Commonwealth University

Dr. F. Douglas Boudinot, Dean of the School of Graduate Studies

[Click here and type the Month, Day and Year this page was signed.]

# A SEQUENTIAL ALGORITHM TO IDENTIFY THE MIXING ENDPOINTS IN LIQUIDS IN PHARMACEUTICAL APPLICATIONS

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Biostatistics at Virginia Commonwealth University.

by

AKRITI SAXENA
MS., Virginia Commonwealth University, 2005
Richmond, Virginia

Post Graduate Diploma, National Institute of Fashion Technology, 2002
New Delhi, India

B.S, University of Lucknow, 2000
Lucknow, India

Director
V. Ramakrishnan (Ramesh) Ph.D.
Associate Professor
Department of Biostatistics

Virginia Commonwealth University
Richmond, Virginia

August, 2009

## Acknowledgements

I feel overwhelmed as I write this part of my thesis. It's a pleasure to express my sincere gratitude to all those people who had been involved with me and this thesis all through this time. That being said, I make a humble approach to acknowledge those special people.

I would like to dedicate this work of mine to my beloved grandfather who although is not with us anymore but his blessings are always with me. I thank my biggest support system, my strength in all difficult times, my parents, Mr Anil Saxena and Mrs Abha Saxena for their unconditional love and support and most of all their faith in me. I thank my sister, Akanksha, for her love and concern for me and also for excusing me for not being with her when she really needed me due to my academics. Special thanks for my mother in law Mrs Kunti Behari for all her love and support to me. I also thank my sister (in-law) Ruchira for always being like an elder sister to me and helping me in all possible ways. Big thanks to Vishal for taking out his time for helping me in SAS programming whenever I needed. I greatly appreciate his efforts and patience in helping me through those complicated SAS codes. A warm thanks to Sudeep for his endless humor and optimism that always brought back the zeal of achieving my goal.

Thank you my little angel, Anoushka, my darling daughter for being my biggest inspiration. My baby, please forgive your mommy if she ever neglected your tiniest need due to her work.

My thanks to that special person in my life, my best friend and a loving husband, Varun for always loving me no matter what I do or don't. This was only possible due to his optimistic attitude, unconditional support and encouragement in every possible manner.

I have no words to express my gratitude for my advisor, mentor and one of the best things that happened to me, Dr. Ramesh for his continuous support and guidance. My thanks to him is just not only for this thesis but for being a teacher to me in a true sense. I sincerely thank him for all his support and encouragement that kept me motivated to finish this thesis against all the odds and challenges life had for me and my family. His efforts, patience and enthusiasm were the greatest support and I can never thank him enough for everything he did for me.

I extend heartfelt gratitude to one of the finest person I have ever met, Dr. Charlie Kish for all his guidance as a committee member and my manager during my internship at Wyeth Consumer Healthcare. I had a wonderful experience working with him during my internship. His experience, knowledge and most of all immense patience gave me a golden chance to refine my statistical knowledge and a practical approach of handling real life statistical problems. I thank Dr. Carol Summitt, my committee member who also

helped me in every big and small statistical problems I encountered at Wyeth. She had been my mentor and guide throughout my internship and helped me ease in the Wyeth family. I thank Jessica McKinney Ketchum for being my committee member and extending her useful inputs for refining this thesis.
A special thanks to Russell Boyle for every help he extended to me all through this time. My sincere thanks to him for his efforts for the internship opportunity I had at Wyeth.

I would like to thank all the faculty members because of whom I could earn this degree. I also thank our staff members Yvonne, Denise and all technology staff for all their help.

I feel very fortunate to be a part of Biostatistics, VCU family. Wherever I go and whatever I do in my career, I will always cherish the education, love and care this family has given me.

# Table of Contents

Page

## List of Tables

Page

## List of Figures

Page

# Abstract

A Sequential Algorithm to Identify the Mixing Endpoints in Liquids in Pharmaceutical Applications

By Akriti Saxena, M.S.

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Biostatistics at Virginia Commonwealth University.

Virginia Commonwealth University, 2009

Major Director: V. Ramakrishnan, Ph.D.
Associate professor
Department of Biostatistics

The objective of this thesis is to develop a sequential algorithm to determine accurately and quickly, at which point in time a product is well mixed or reaches a steady state plateau, in terms of the Refractive Index (RI). An algorithm using sequential non-linear model fitting and prediction is proposed. A simulation study representing typical scenarios in a liquid manufacturing process in pharmaceutical industries was performed to evaluate the proposed algorithm. The data simulated included autocorrelated normal errors and used the Gompertz model. A set of 27 different combinations of the parameters of the Gompertz function were considered. The results from the simulation study suggest that the algorithm

is insensitive to the functional form and achieves the goal consistently with least number of time points.

# CHAPTER 1

# Introduction

In this thesis estimation of a measure called refractive index (RI), used in food and beverage industry, chemical, petrochemical and medical/pharmaceutical manufacturing is of interest. In this chapter the RI is defined and some of its applications are discussed.

## 1.1 Refractive Index

Light travels in different mediums and its speed within any medium depends on the density of that medium. When a light enters a medium, the change in the speed of light causes the ray to be deflected or bent. This phenomenon is called refraction. A measure of this is refractive Index (RI) and it is defined as the ratio of the velocity of a light in air to the velocity of light in a substance (USP/NF, 2006). The RI denoted by $\eta$ is given by

$$\eta = c / v ,$$

where $c$ is the speed of light in air ($3\times10^8$ meters/sec) and $v$ is the velocity of radiation of a specific frequency in a specific material .

The RI could also be defined as the ratio of the sine of the angle of incidence to the sine of the angle of refraction (Figure 1.1). This relationship between the angles of incidence and refraction and the indices of refraction of the two mediums is known as Snell's Law.

$$n_1 \sin\theta_1 = n_2 \sin\theta_2 ,$$

where $\theta_1$ is the angle of incidence, $\theta_2$ is the angle of refraction $n_1$ is index of refraction of the incident medium and $n_2$ is the index of refraction of the refractive medium. The Figure 1.1 shows the phenomenon of refraction as defined by Snells law.

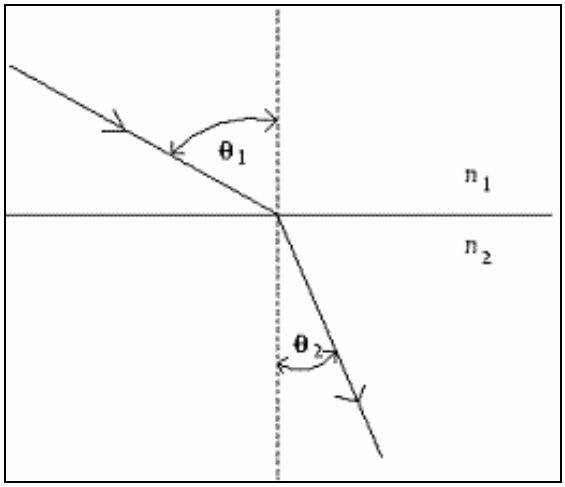



**Figure 1.1: Diagram showing Refraction by Snell's law**

Since RI is a ratio, it is a unitless number. For most of the compounds, the value for RI is between 1.3000 and 1.7000. (It is generally reported to four or five digit precision.) The RI of some common materials is shown in Table 1.1.

**Table 1.1: Refractive Index of common materials**

| **Material** | **Refractive Index ($\eta$)** |
|---|---|
| Vacuum | 1.0000 |
| Water | 1.3300 |
| Glass | 1.4500 - 1.4800 |

The RI is a fundamental property of a substance. Since RI of a substance is constant at a given pressure and temperature, it is often used to identify a particular substance and dispersion of solids and liquids in it, confirm its purity and measure the concentration of solutes in water based solutions .The RI is very sensitive to small amounts of impurities. Even a small change in the concentrations of the fluid can be detected by measuring small changes in the index of refraction.

The RI is measured by an instrument called refractometer. It is used to assess the purity of a particular substance, or to determine the concentration of one substance dissolved in another. To measure the RI of a mixture, an inline process refractometer is incorporated in piping of manufacturing plants, liquid mixing devices, and washing apparatuses to measure the concentration of various liquids. An inline process refractometer is designed for the continuous measurement of concentration of dissolved solids or water soluble liquids flowing through a pipe or inside a tank. It has a sensor placed inline with the fluid flow along with a control box which provides the temperature compensated reading and relay outputs for controlling pumps and valves. Inline process refractometer provides 24-hour monitoring of quality without the time involved in periodic manual sampling and therefore has extensive applications in the food, beverage, sugar, chemical and metalworking industries. The mixing is completed when the mixture achieves a specified RI, so the mixing process is closely monitored and the RI is measured very frequently making the data volume very large. Figure 1.2 shows an example of inline process refractometer.

.


**Figure 1.2: Inline Process Refractometer (Unitech USA)**

## 1.2 Applications of RI

The main objective of measuring RI in various industrial applications is to maintain an optimum product consistency during manufacturing. Therefore the industries routinely monitor RI throughout the manufacturing process. This benefits industries in many ways such as achieving consistency in the end product, maximizing energy efficiency and increasing production capacity. Some of the industries where RI plays a significant role are listed below.

### 1.2.1 Food and Beverage Industry

The RI is widely used in food and beverage industries in the manufacturing of tomato products, applesauce, baby food, fruit syrups, various jams and jellies, gelatin, milk products, syrups, corn, various beverages (e.g., coffee, malt). Different products have different specifications of required RI. For example, the minimum acceptable RI specified for tomato sauce is 1.3455 (USDA, 1994).

### 1.2.2 Chemical and Petrochemical Industry

The RI is used to characterize and identify different solvents, distillation products, organic solution, and organic polymers in chemical industries. It also has application in many petrochemical products like oils, fats, waxes, naphthalene, paints and so on (Kenkel, 2002). For example RI of ethanol (at $20^0$ C) is 1.361 and for glycerol it is 1.4731 ( Hech, 2003)

### 1.2.3 Medical and Health industry

The RI is widely used in medical industry specifically in analyzing blood protein concentration, salinity and specific gravity of urine or serum. Any shifts in their RI indicate a cause for concern in the sample being tested.

### 1.2.4 Environmental Application

In environmental applications, it is of interest in measuring the concentration of detergent in protein detergent mixtures to study their impact on health. (Strop and Brunger, 2005). There are several ways of measuring this concentration one of which is using the RI. Apparently unlike some of the other methods, the RI quantification is not limited to detergents containing sugar (Urbani and Warne 2005). Further the RI does not require high concentrations of detergents (daCosta and Baenziger 2003; Eriks et al.2003).

## 1.3 Introduction

The US Food and Drug Administration(FDA) encourages the use of process analytical technology (PAT) to increase efficiency, cost-effectiveness, and quality of manufacturing in the pharmaceutical industry (FDA, 2002, 2004). The FDA describes PAT, in broad sense, as “a system for designing, analyzing, and controlling manufacturing through timely measurements (i.e., during processing) of critical quality and performance attributes of raw and in-process materials and processes with the goal of ensuring final product quality.”(FDA, 2002, 2004)
In pharmaceutical industries, the objective of PAT is to understand and control the manufacturing processes to have consistency with the current drug quality system.
While many other industries have used PAT, the pharmaceutical industries have recently begun to implement PAT and are attempting to match the other industries in technology, manufacturing requirements, and methods of data evaluation. In the PAT framework, one of the tools that enable scientific, risk-managed pharmaceutical development, manufacture, and quality assurance is process and endpoint monitoring and control tools.

Following the general lead of the food and beverage industries, the pharmaceutical industry is beginning to implement in-line RI to objectively determine mixing endpoint for the liquid products. When liquids are well mixed, the RI appears to "level-off" or reach a "steady-state plateau". It is of interest to estimate when this phenomenon occurs in the pharmaceutical products accurately, quickly and dynamically. In this thesis, using a non linear statistical model fit, a sequential algorithm is proposed for determining the moment at which the RI data reach a steady state or plateau. In Chapter 2 some of the methods currently used to measure the plateau using RI are discussed. In Chapter 3, the algorithm proposed in this thesis is presented. The

results from simulation study are also presented in this chapter. In Chapter 4 problems for further research are discussed.

# CHAPTER 2

## Background

There are several statistical methods that are widely used to identify the steady state plateau using the RI data and determine mixing endpoints in liquids. In this chapter, some of these methods are described. When the process has reached a steady state plateau it is also considered to be 'in statistical control.' Therefore some of the methods that use control charts could also be used for finding the steady state plateau for RI data. Two such methods (Shewhart chart and Exponentially Weighted Moving Average and Variance) are described in section 2.1 to 2.2.

### 2.1 Shewhart chart

A process is said to be in statistical control when the variability in the process is inherent to the process, i.e. random or chance caused and not due to assignable causes such as operator, machine or a certain batch or a material. A control chart known as Shewhart chart is a tool widely used in manufacturing process to determine whether a process is in statistical control. If the chart indicates that the process is currently under control, at a specified confidence level, it could then be used to predict the future performance of the process. If the chart indicates that the process is not in control, it can provide information on the source of variation so that it could be eliminated to bring the process back in control. The Shewhart chart is based on run charts, which plot the process characteristics in chronological sequence. For the Shewhart chart samples from the process are taken at regular intervals and simple performance statistics such as mean, range, variance, number of defects, etc., are calculated and graphed over time. The graph is checked to

see if the process is within specific control limits. The formula for control limits depends upon the distribution of the data being monitored. For Normally distributed data, these control limits simply be $\pm 3\sigma$ limits. That is,

$$\text{Upper Control Limit (UCL)} = \mu + 3\sigma,$$

$$\text{Lower Control Limit (LCL)} = \mu - 3\sigma,$$

where $\mu$ is overall process mean and $\sigma$ is process standard deviation. In the control charts, lines through $\mu$ (central line), and the two limits (UCL and LCL) are shown. The control charts may also show upper and lower 'warning limits' set typically at $2\sigma$ above and below the overall process mean. (Keats, J.B. and Montgomery, D.C, (1991)).

Once the control limits are defined, rules for deeming a process to be in or out of control need to be specified. One such set of rules, used in conjunction with the Shewhart charts, is the Western Electric rules (Montgomery, D. (2001). The Figure 2.1 shows an example of a Shewhart chart where the points marked 'X' denote the out of control data points with the Western Electric Rule. If any one of the X's occurs the process would be deemed out of control.
These rules are as follows.

1. At least one point is out of the $\mu \pm 3\sigma$ zone. (In Figure 2.1, point sixteen is above the UCL.)

2. At least two out of three consecutive points are outside of the warning limits and are on the same side of the central line. (In Figure 2.1, two consecutive points, 3 and 4, are outside the -2$\sigma$ limit.)
3. At least four out of five consecutives points are outside of $\mu \pm \sigma$ limits and are on the same side of the central line (mean). (In Figure 2.1, among the consecutive points, 7 to 11, the four points, 7, 8, 10 and 11 are outside the $+\sigma$ limit.)
4. Eight consecutive points are on the same side of the central line. (In Figure 2.1, consecutive points, 14 to 21 are above the centerline.)
5. Eight consecutive points are monotonically increasing or decreasing, whether or not they are outside of any limits. (This is not shown in Figure 2.1.)

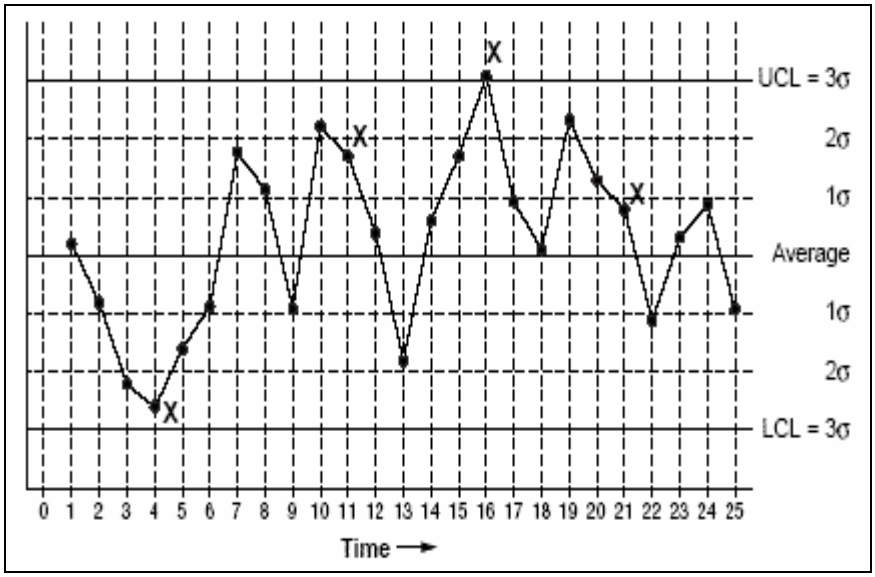



**Figure 2.1: Shewharts chart showing out of control signals(X) under Western Electric rules, Nancy R. Tague, 2004.**

For the RI data which has a high sampling frequency and low inherent variability, a plateau is reached when the process is in statistical control. That is, none of the above five rules are true for a moving window of a specified number of consecutive points (e.g., 20 consecutive points).

**2.2 Exponentially Weighted Moving Average and Variance (EWMA and EWMV)**

The Exponentially Weighted Moving Average (EWMA) and Variance (EWMV) control charts are proposed for monitoring various types of continuous process variation. They are particularly ideal for individual observations across time where no estimates of variability are available from replicates and/or the observations are autocorrelated. These charts plot weighted moving average values over time. A weighting factor is chosen by the user to determine how previous data points affect the mean value compared to more recent ones. For the EWMA control technique, the decision depends on the EWMA statistic, which is an exponentially weighted average of all prior data, including the most recent measurement. The EWMV charts are useful for augmenting control charts where no estimates of variability is available from replicates (J.F Macgregor, T.J Harris, 1993).

The EWMA and EWMV chart uses information from all samples and therefore is a better alternative to the Shewhart control chart when detecting small shifts. They also handle correlated data in some situations. They also allow different targets across batches and are used widely in accounting and chemical processes.

### 2.2.1 EWMA calculations

The EWMA requires specification of design parameters ($\lambda$, $\sigma_0$), where $\lambda$ is the EWMA weight such that $0 < \lambda \leq 1$ and $\sigma_0$ is the standard deviation of the historical data when the process was under control. At any time *i* EWMA is defined

$$\text{EWMA}_i = \lambda y_i + (1-\lambda)\,\text{EWMA}_{i\text{-}1} \tag{2.1}$$

for $i = 1$ to $n$, where $y_i$ is the RI at time $i$ and $n$ is the number of observations to be monitored (Robert,1959). Initially, at time 1 ($i = 1$), the value of $\text{EWMA}_0$ has to be specified. The recommended value for this is the mean of the historical data or the process target.

The parameter $\lambda$ determines the rate at which previous data enter into the calculations of the EWMA statistic. If $\lambda = 1$ only the most recent measurement influences the EWMA and the EWMA converges to Shewhart chart. Usually the choice of weighting factor is arbitrary and through $\lambda$, EWMA control procedure can be made sensitive to a small or gradual drift in the process, whereas the Shewhart control procedure can only react when the last data point is outside a control limit.

For independent observations and large *n,* the estimated variance of the EWMA statistic is

$$s^2_{EWMA} = (\lambda / (2-\lambda))s^2 \ , \tag{2.2}$$

where s is the estimated standard deviation possibly obtained from the historical data.

The control limits are calculated by $EWMA_0 \pm 3\sigma_0\sqrt{\lambda/(2-\lambda)}$ for independent observations (Hunter, 1986).

The EWMA procedure depends on a historical data that are truly representative of the process. Once the mean value and standard deviation have been calculated from this dataset, the process

can enter the monitoring stage, provided the process was in control when the data were collected. Once EWMA estimates are available, the process could be deemed in or out of control using the Western Electric rules as described under Shewhart chart.

### 2.2.2 EWMV calculations

The EWMV will respond to both the changes in mean and variance. Therefore it is useful to compute EWMV about some estimate of process mean.

The EWMV also requires specification of design parameters ($\lambda$, $\sigma_0$), where $\lambda$ is the EWMA weight such that $0 < \lambda \leq 1$ and $\sigma_0$ is the standard deviation of the historical data when the process was under control. At any time *i*, EWMV is defined

$$\text{EWMV}_i = \lambda(y_i - EWMA_i)^2 + (1-\lambda)EWMV_{i-1}, \tag{2.4}$$

For i = 1 to n, where $y_i$ is the RI at time *i,* n is the number of observations to be monitored and $\sigma_0$ is the standard error of the historical data when process was under control (Robert, 1959). As in EWMA calculations, initially at time 1 ($i$ = 1), the value of $\text{EWMA}_0$ has to be specified. Also the initial value for EWMV is

$$\text{EWMV}_0 = \sigma_0^2 \tag{2.5}$$

The upper and lower control limits for EWMV charts are given by ( $\sigma_0 - C_7\sigma_{0,}\sigma_0 + C_8\sigma_0$ ) where $C_7$ and $C_8$ are square roots of the critical values based on the value of $\lambda$ or by other methods such as Johnson Box approximation (Sweet,1986).

# CHAPTER 3

# Method

In this chapter an algorithm to estimate the mixing endpoints (steady state plateau) using a sequential prediction from a non linear fit to the RI data is proposed. In Section 3.1 typical non linear functions are described. In Section 3.2 fitting non linear model is briefly summarized. In Section 3.3 steps for determining the plateau are presented. In Section 3.4 the application of the proposed method on a simulated data is described.

## 3.1 Non Linear Functions

Let $f(y \mid x, \boldsymbol{\theta})$ represent a non linear function for an outcome $y$ in terms of an independent variable $x$, where vector $\boldsymbol{\theta}$ represents the unknown parameters. In this section some examples of non linear functions that have plateau(s) are presented.

### 3.1.1 Exponential Function

An exponential function is used in physical sciences to model growth where the growth rate of an organism is proportional to the function's current value. This rate of growth could be in either the positive or negative direction. A representation of the exponential function is,

$$y = \theta_1 - \exp\left(-\theta_2 x\right), \tag{3.1}$$

where exp denotes exponential $\theta_1$ is the asymptote (as $x$ tends to $\infty$), and $\theta_2$ is the rate of increase in the log scale with respect to $x$. A plot of an exponential function for $\theta_1 = 1.38$ and $\theta_2 = 1$ (chosen to represent the RI data used in this thesis) is shown in Figure 3.1.

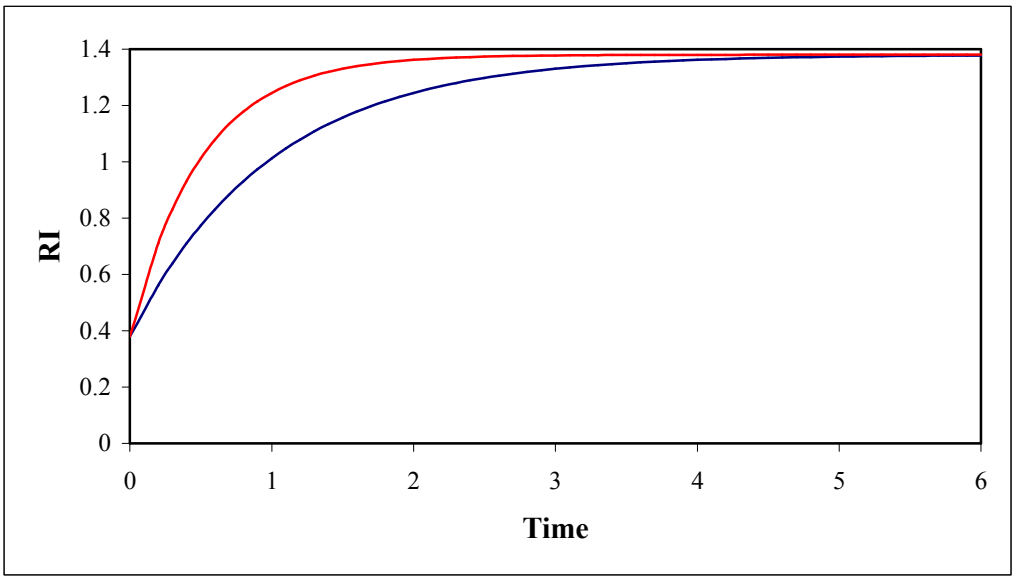



**Figure 3.1: Exponential Function:** $\theta_1 = 1.38$ and $\theta_2 = 1$ (blue) or 2 (Red)

Exponential models arise whenever quantities grow or shrink by a constant factor, such as in radioactive decay or population growth. Some of the examples of exponential growth are microorganisms in a culture dish that grow exponentially until the nutritients are exhausted. Another application is in the growth of virus that initially spread quickly in the absence of any artificial immunization and plateau once the virus is contained.

### 3.1.2 Gompertz Function

Gompertz curve is a generalized form of the exponential model and is used generally where growth is slowest at the start and end of the growing period leading to an S shaped curve. A representation of the Gompertz function is,

$$y = \theta_1 \exp\left(\theta_2 \exp(-\theta_3 x)\right) , \tag{3.2}$$

where exp is the exponential , $\theta_1$ is the asymptote, $\theta_2$ represents the early growth , $\theta_3$ represents the linear phase of the growth (growth spurt).

An alternative representation of the Gompertz function, which is commonly used, is obtained by taking the natural logarithm of (3.2). That is,

$$y' = \alpha(1 - \exp(-\beta - \gamma x)) , \tag{3.3}$$

where $y'$ is the log of $y$ , $\alpha = \ln(\theta_1), \beta = -\ln(\theta_2 / \ln(\theta_1))$ and $\gamma = -\theta_3$ . Example plots of Gompertz curves showing the effect of varying $\theta_2$ and $\theta_3$ are presented in Figure 3.2.

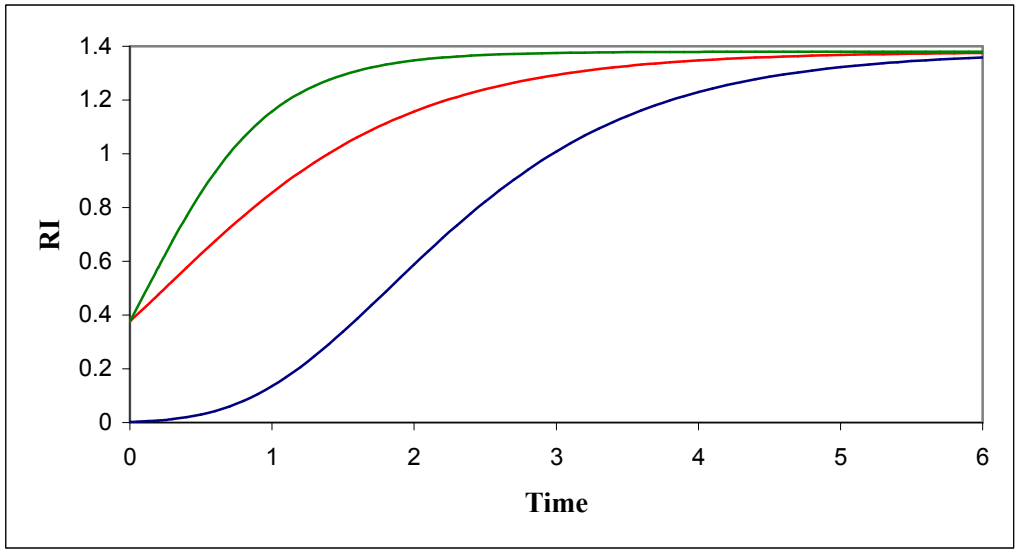



**Figure 3.2: Gompertz curve:** $\theta_1$ =1.38, $\theta_2$ = 1.3, $\theta_3$ = -1 (Blue), or $\theta_2$ = 1.3, $\theta_3$ = -2 (Green) or $\theta_2$ = 6.3, $\theta_3$ = -1 (Red)

The main applications of the Gompertz function are in population studies and animal growth, especially when the growth is not necessarily symmetrical about the point of inflection.

### 3.1.3 Logistic Function

The functional form of the logistic function (Katsunori Fujii, 2006) is,

$$y = f(x \mid \boldsymbol{\theta}) = \frac{\theta_1}{1 + \exp(\theta_2 - \theta_3 x)}. \tag{3.3}$$

The logistic function also describes a Sigmoid shaped process. It is primarily used to model biological population growth in species which have grown large and are nearing their saturation. It has applications in other fields such as neural networks where it is used to introduce non-linearity in the process. The decline of demand for a product as a function of increasing price can be modeled in marketing by a logistic function. In statistics, the logistic function is applied with categorical data and also in regression to model the probability of an event as a function of explanatory variables. Example plots of this function are shown in figure 3.3.

Here again, $\theta_1$ represents the asymptote, $\theta_2$ is the early growth and $\theta_3$ is the growth (growth spurt).

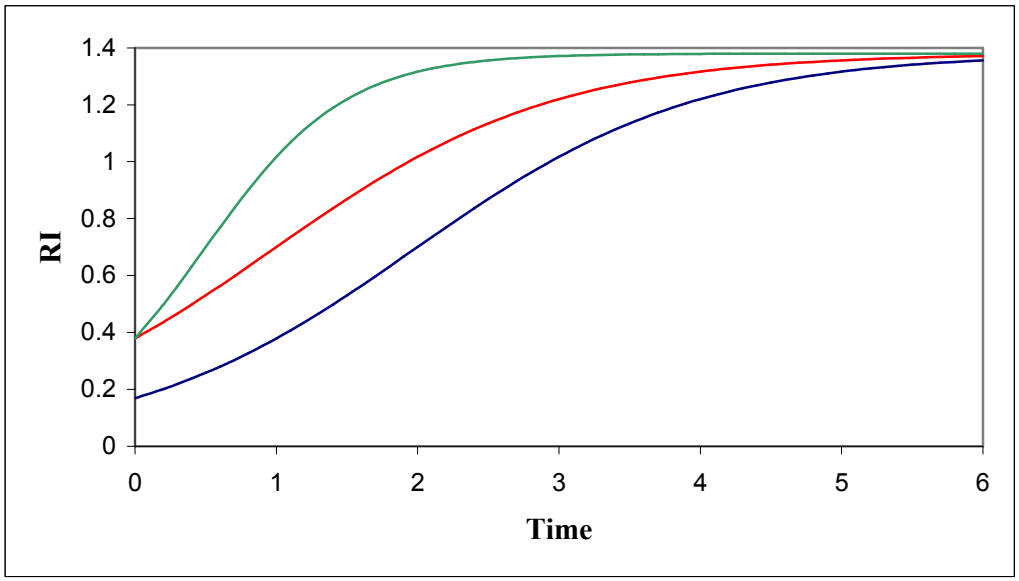



**Figure 3.3: Logistic function:** $\theta_1 = 1.38,\ \theta_2 = 1.97,\ \theta_3 = 1$ (Blue), or $\theta_2 = 0.97,\ \theta_3 = 1$ (Green) or $\theta_2 = 0.97,\ \theta_3 = 1$ (Red)

### 3.1.4 Double Logistic Function

The double logistic function is similar to the logistic function. It is graphically similar to two identical logistic functions bonded together at the point $x = g$. The functional form of the double logistic function is

$$y(x) = \frac{\theta_1}{1+e^{-\theta_2(x-\theta_3)}} + \frac{g-\theta_1}{1+e^{-\theta_4(x-\theta_5)}}, \tag{3.4}$$

where g is known, $\theta_1$ $\theta_2$, $\theta_3$, $\theta_4$ and $\theta_5$ are the parameters( Katsunori Fujii,2006 ). This function is similar to the logistic function and is useful in describing biological growth in which there are two phases. For instance, the human growth which has a childhood phase (e.g., 0 to 2

years of age) and an adult phase (e.g, 2 to 18 years of age). An example of the double logistic function is shown in Figure 3.4.

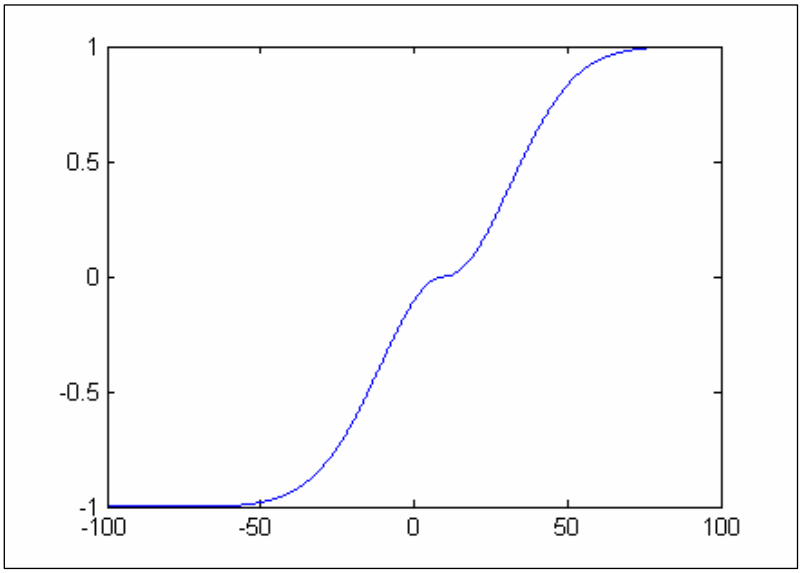



**Figure 3.4: Double logistic function**

### 3. 2. The method for identifying the plateau

The process of identifying the plateau begins by determining an appropriate non-linear functional form. Then the next step is to estimate the plateau of the specified non-linear function. These two steps are described below.

#### 3.2.1 Determining an appropriate functional form for the data.

In this step a functional form to describe the process is identified. To begin, first obtain sample data that represents the process. Then by plotting the data and comparing with known functions as those described above, identify an appropriate function. Identifying an appropriate non-linear function requires the use of empirical evidence, knowledge of the process, and trial and error

experimentation. Sometimes there is historical knowledge about functions that appropriately fit a given process well. In the absence of a historical model or experience with prior data sets, selecting an appropriate function will often require a certain amount of trial and error.

### 3.2.2 The estimation of the non-linear model.

Since iterative methods are applied to fit a non-linear model, appropriate initial values for the estimation of the model parameters need to be determined. Some models are extremely sensitive to the choice of initial values whereas others are less sensitive. There are a number of ways to determine these initial values. Information from previous fits of similar data might be an option. Sometimes appropriate guesses from the functional form could also determine initial values. Finally, most statistical procedures such as the NLIN procedure in SAS are quite robust to the initial values and therefore their specification is generally not an issue.

In this thesis, NLIN procedure in SAS was used to estimate the model parameters. As stated earlier, estimation of a non-linear model is an iterative process and so to begin this process the NLIN procedure first examines the starting value specifications of the parameters. The procedure uses derivatives or approximations to derivatives of the Sum of Squares Error (SSE) with respect to the parameters to guide the search for the parameters that minimize SSE. If a grid of values is specified, NLIN procedure evaluates the residual sum of squares at each combination of the parameter values to determine the set of parameter values producing the lowest residual sum of squares. These parameter values are used only for the initial step of the iteration. (See SAS help and documentation for further details).

Another characteristic of the RI data is that the observations over time are likely to be serially correlated. That is, observations closer together have higher correlations than those farther apart.

A process that best describes such correlations is the auto-regressive process (e.g the first-order autoregressive process (AR (1))). Estimating this auto-correlation via non-linear modeling is tedious and common software does not incorporate this option. Therefore, the algorithm proposed in this thesis does not include the simultaneous estimation of the AR(1) parameter. This will be considered in subsequent research. (See chapter 4 under future research.) (However, in the simulation study presented later in this chapter the data were generated under the assumption of AR(1) errors to examine if the proposed method is insensitive to this assumption.)

### 3.3 Steps for identifying when the plateau is reached.

Once an appropriate non-linear function has been identified and the estimation of the parameters of that function has been established, identification of when the plateau is reached over time is achieved through another iterative or sequential process. The steps in this process are described below.

1. Initialize: Define the number of initial points, $k_0$, for the beginning of the algorithm for the process of interest. Based on prior information, choosing $k_0$ close to the plateau will speed up the convergence of the algorithm. Let the data from these $k_0$ points be denoted by $y_1, y_2, ..., y_{k_0}$ . Select an appropriate non-linear function for the process. Specify the tolerance, $\tau$.
2. Fit model: In the $i^{th}$ iteration fit the non-linear model to the $k_i$ data points.
3. Predict future observations: In the $i^{th}$ iteration, use the model fitted in step 2 and predict the next $l$ observations, $\hat{y}_{k_i+j}$ $j$= 1, 2… to l. (The number of points predicted ($l$) is

arbitrary and could be specific to the process. Compute the average of the squared differences between $\hat{y}_{k_i+j}$ and the last observation in the data $y_{k_i}$. That is,

$$\hat{\mu}_i = \frac{\sum_{j=1}^{l} (\hat{y}_{k_i+j} - y_{k_i})^2}{l}$$

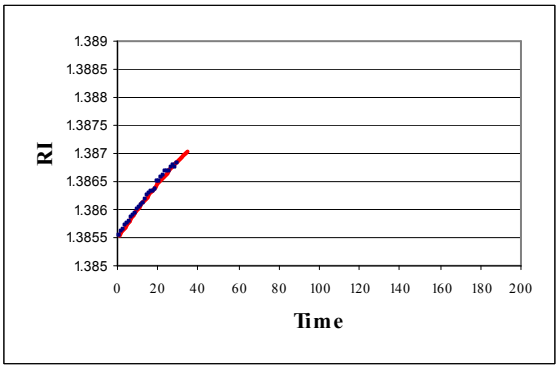
4. Check for convergence: If the average of the squared difference computed in Step 3, $\hat{\mu}_i$ is larger than a predefined tolerance $\tau$ , increase $k_i$ to $k_{i+1} = k_i + l$ and go back to step 2. If $\hat{\mu}_i < \tau$ , the model has reached the plateau. When the iterations stop the time when the plateau is reached is obtained.

The rationale for Step 4 is that if the process has reached the asymptote, the predicted future observations should not vary much from the last recorded observation. Therefore, the difference between the last observation and the predicted observations would be close to zero. For instance, if the process is at the increasing phase of the process, the predicted observations would be significantly larger than the last observation and therefore the proposed algorithm will continue. In general, predicting beyond the range of values of the observed data (i.e., extrapolating) leads to a large variability. Therefore, the convergence of the algorithm is more likely to lead to the true time at which the plateau is reached.

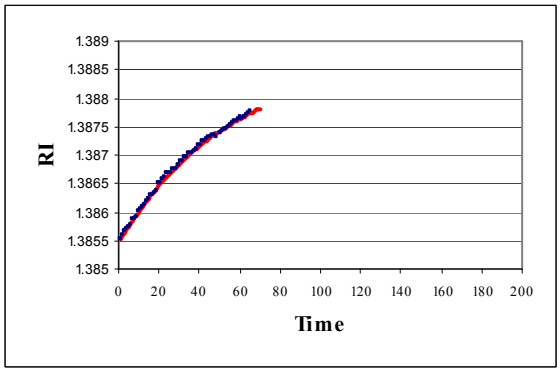
In Figure 3.5 a) - e), a simulated example of the progression of the algorithm through some of the iterations in a typical case is shown. The blue points show the actual data and the red lines show the fitted line with predicted values for five additional points. In Figure 3.5a) the initial step with 30 points is shown. Notice that the process is in the linear phase. The Figures 3.5 b), c), d) and e) show the eighth, 16$^{th}$, 24$^{th}$ and 29$^{th}$ iterations respectively. In this case the algorithm

converged at the $29^{th}$ iteration. The Figure 3.5 f) shows the complete data for comparison. (Larger versions of these figures could be found in the Figure C.1a)-f), Appendix C.)

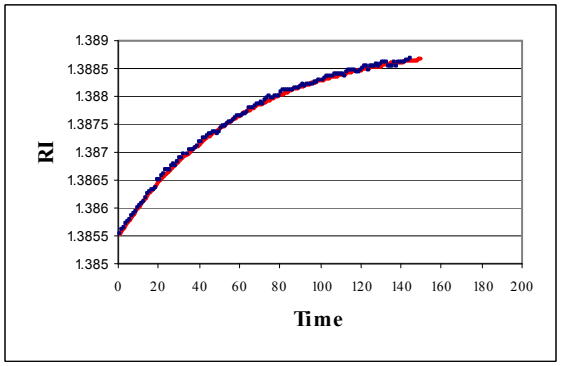
In figure 3.5 the algorithm is presented in the form of a flow chart.



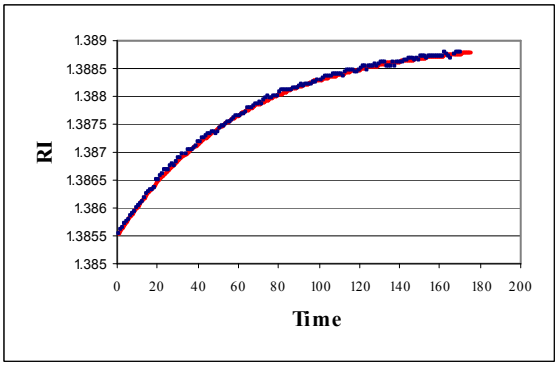

a) Iteration 1



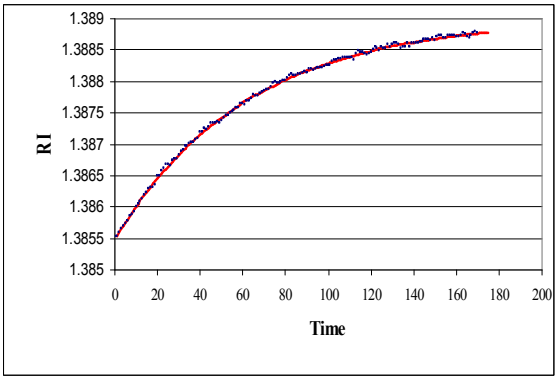

b) Iteration 8



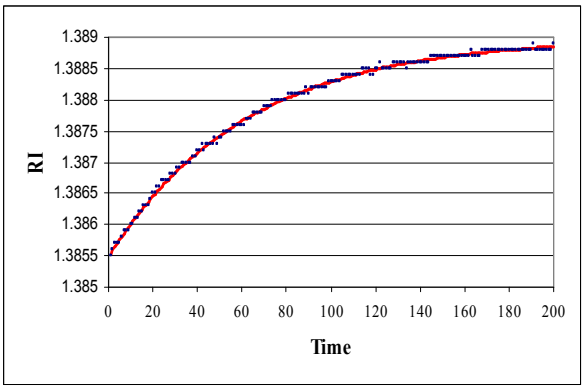

c) Iteration 16




d) Iteration 24




e) Iteration 29




f) All data

**Figure 3.5: Progression of the Algorithm**

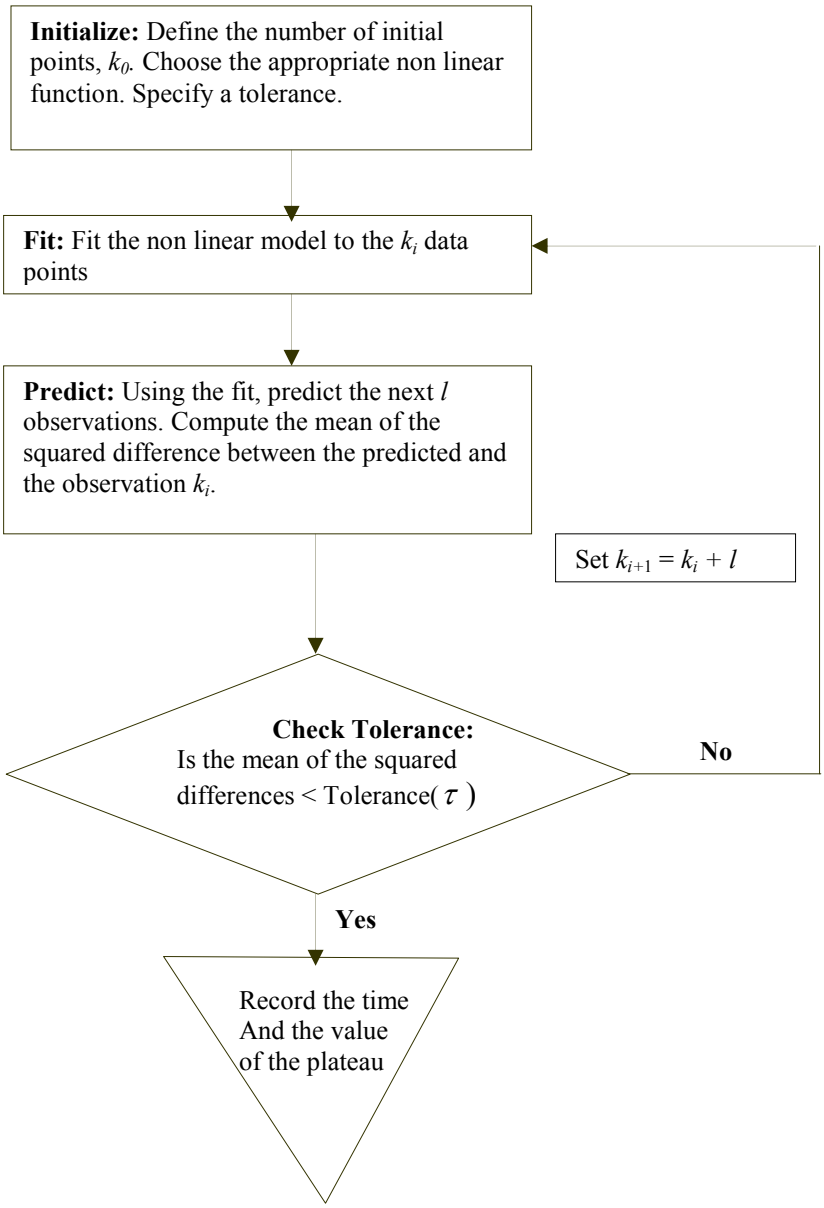



**Figure 3.6 Flowchart describing the Algorithm for identifying the plateau**

### 3.4 Application of the proposed algorithm

Although numerous industries generate numerous data in which the application of the proposed algorithm would be appropriate, due to unavailability of company's specific confidential data, it was not possible to demonstrate the method using actual raw data. However, estimates of parameters from actual processes were obtained and data were simulated from these estimates so that the properties of the algorithm could be studied, while preserving the confidentiality of the actual data. In what follows, the simulation study and the results are presented.

#### 3.4.1 Simulation Process: Obtaining the estimates of the model parameters for simulation.

In pharmaceutical applications the RI-time curve exhibit multiple degrees of slopes leading to varying areas of plateaus representing many processes. Data collected from multiple scenarios discussed earlier were obtained from liquid manufacturing process that involves addition of raw material at varying times. Before fitting the model in each scenario, first the region for which the steady state plateau is to be identified was extracted from the complete mixing profile of that process. In these processes plateaus may be reached in an increasing direction, a decreasing direction or both. The process is increasing at the beginning phases and continues to increase after reaching several plateaus. When the process starts decreasing, in most of the products it indicates the final mixing stage. In this thesis only the plateaus for increasing profiles were used for the simulations.

For the simulation study, representative data sets were obtained from a local pharmaceutical company. They were first plotted two-dimensionally, with the X axis representing the time and the Y axis representing the RI. Based on experience and examining the

shape of the graph as well as the residual plots, the Gompertz function seemed appropriate to fit the process. In addition, since the Gompertz function is versatile in fitting variety of shapes it was chosen in the simulation study for testing the algorithm. The form of the function fitted was

$$y = \alpha(1 - \exp(-\beta - \gamma t)) \tag{3.5}$$

Here, $\alpha$ is the asymptote, $\beta$ and $\gamma$ are the growth rate, $t$ is time and exp is the exponential. (See Section 3.1.2 for more details). The above non-linear model was fitted to all the representative data sets provided to us and corresponding estimates of the three model parameters were obtained using NLIN procedure in SAS.

The initial parameters that are required for fitting were supplied in the model by trial and error until the convergence criterion for the maximum likelihood estimation was met. The parameters α, β, γ and the Mean Square Error (MSE) were estimated. The residual plots for each model fit were examined. While the normality seemed to be appropriate for all the processes, there was an indication of lack of independence. The observations over time seemed to be autocorrelated. Therefore, to represent the actual data in the simulations it was decided that an autoregressive (AR(1)) error would be generated. To decide upon what would be the magnitude of the autocorrelation the AR (1) estimates were obtained from the residuals of these models using the formula,

$$\hat{\rho} = \frac{\sum_{t=2}^{n} \varepsilon_t . \varepsilon_{t-1}}{\sum_{t=1}^{n} \varepsilon_t^2}. \tag{3.6}$$

One may also estimate $\rho$ using the option DW in the model statement in PROC REG using SAS. (**MODEL** e=*/dw;),* where DW is the Durban-Watson statistic to test if the serial correlation is zero.

Once these estimates were obtained from all the representative data sets provided, the minimum, median and maximum of the three model parameters were recorded. For the simulation study these three levels of the parameter estimates were used to form 27 combinations. The MSE and the $\hat{\rho}$ were set at the median value. These combinations are presented in tables A1 and A2, of Appendix A. (Although this entailed major part of the thesis work, the details of the model fit and the graphs of the original data presented to us are not provided to preserve confidentiality.)

### 3.4.2 Generation of simulated data

The following steps were used to simulate the data using SAS.

Step 1. Autocorrelated normal distributed residuals.

First random numbers having a normal distribution with mean 0 and variance 1 were generated using the following statement in SAS,

$$z = rannor(\mathbf{seed}).$$

Next $z$ was scaled by the MSE using the statement

$$u = sqrt(mse) \cdot z$$

Using the estimate of autocorrelation ( $\hat{\rho}$ ) the autocorrelated normal errors were generated using the following statement

$$e_i = \hat{\rho} \cdot e_{i-1} + u_i,$$

where the initial value of $e_0$ is set as 0.

Step 2. Gompertz data

For each combination of the parameter values, observations following a Gompetz model were generated using the formula,

$$y = \alpha(1 - \exp(-\beta - \gamma x))$$

where the time $t = 1, 2, \ldots, 200$.

Step 3. The RI data

The RI data were generated by adding the autocorrelated errors from Step 1 to the Gompertz data, *y,* obtained in Step 2 as

$$RI = y + e.$$

The process was repeated 100 times. Thus, there were 27 combinations of data each of which had 100 samples following a Gompertz model over 200 time points.

**3.5. Application of the algorithm to the simulated data**

The proposed method to estimate the plateau as described in section 3.3 was applied to the simulated data and the asymptotes (α) were estimated for each combination of the parameters.

Step 1: For the initial step $k_0 = 30$ points were selected. The Gompertz function was chosen. The tolerance was set to be 0.000000009, based on the variance observed in the representative processes.

Step 2: The Gompertz model was fitted using the NLIN procedure in SAS.

Step 3: Using the OUTPUT statement the predicted values for the next five observations ($l$ = 5) were obtained. In the $i^{\text{th}}$ iteration, the squared differences between the predicted values $\hat{y}_{k_i+j}$, ($j$=1, 2…..$l$) and $y_{k_i}$ were computed.

Step 4: The mean of the differences computed in step 3 was obtained using the PROC MEANS procedure. This mean was compared to the tolerance of 0.000000009.

Step 5: When the process converged, for each simulation, the plateau, the number of iterations, the parameter estimates and their standard errors were recorded.

Step 6: Recall, $\alpha$ represents the true plateau. Therefore, for each combination of the parameters, the Bias and MSE were calculated as

$$\text{Bias} = \frac{1}{N}\sum_{s=1}^{N}(\hat{\alpha}_s - \alpha), \tag{3.7}$$

and

$$\text{MSE} = \frac{1}{N}\sum_{s=1}^{N}(\hat{\alpha}_s - \alpha)^2, \tag{3.8}$$

where $\alpha_s$ represents the estimated plateau from the $s^{th}$ simulation and $N$ represents the number of simulations (100 in this case)

**3.6 Simulation Results**

The purpose of the simulation study performed here is mainly to examine the characteristics of the sequential algorithm for different types of processes. Specifically, in terms of bias and MSE, the purpose was to examine how the size of alpha, beta and gamma affect the algorithm. The simulation results from the 27 combinations of model parameters, sorted from lowest to highest

bias of the estimated plateaus are presented in Table B1, Appendix B. The same results are presented again in Table B2 Appendix B but sorted from lowest to highest MSE of the estimated plateau. Table B3 presents the results sorted for highest number of mean iterations the algorithm takes to converge. In these tables, column Alpha represents the asymptote, columns Beta and Gamma are the growth rates for the curves. The other columns of the table are the mean and variance of plateaus across 100 simulations for each combination of model parameters. The tables also show the mean and variance of number of iterations required for the statistical model to meet the condition of the tolerance.

For the discussion, the three values of each of the model parameters which are minimum, median and maximum will be referred to as Low, Medium and High respectively (See Table A1, Appendix A).

**3.7 Conclusions**

The figure 3.9 presents the plots of plateau MSE and bias for all the 27 combinations.

From the results several characteristics emerge.

- In general the bias and the MSE for all the combinations seem to be low. (See Tables B1-B3, Appendix B and figure 3.9). Recall, the low, medium and high values for the various parameters were determined from the actual data. Therefore, the algorithm seems to identify the plateau accurately for most practical situations.
- The main difference in the results seems to arise for the low beta and low gamma combinations. Irrespective of the level for Alpha when Beta and Gamma are small the bias and the MSE is the largest for these combinations. In figure 3.9 the points represented by the combinations, 111, 211 and 311 (low Beta and low Gamma) have the

highest bias and MSE. The best performance of the proposed algorithm (i.e. the least MSE and bias) is in the models with the combinations of medium or high Beta with medium Gamma. (Table B1 and B2, Appendix B).

- When Gamma is low the number of iterations required by the model to meet the tolerance level is much higher and could go up to 20. (See Table B3, Appendix B). The number of iterations essentially represents the number of time points needed to identify a plateau. In the algorithm used, 20 iterations would imply 130 time points (30 + 20 x 5). The algorithm seems to converge with a mean of 30 to 35 time points when Beta and Gamma are high or medium. These are the cases that represent a steep increase in the early stages of the process. There were combinations (group 1 and 10) with a low Gamma that took 150 time points to identify the plateau. These are the cases where the processes gradually increase to the plateau.

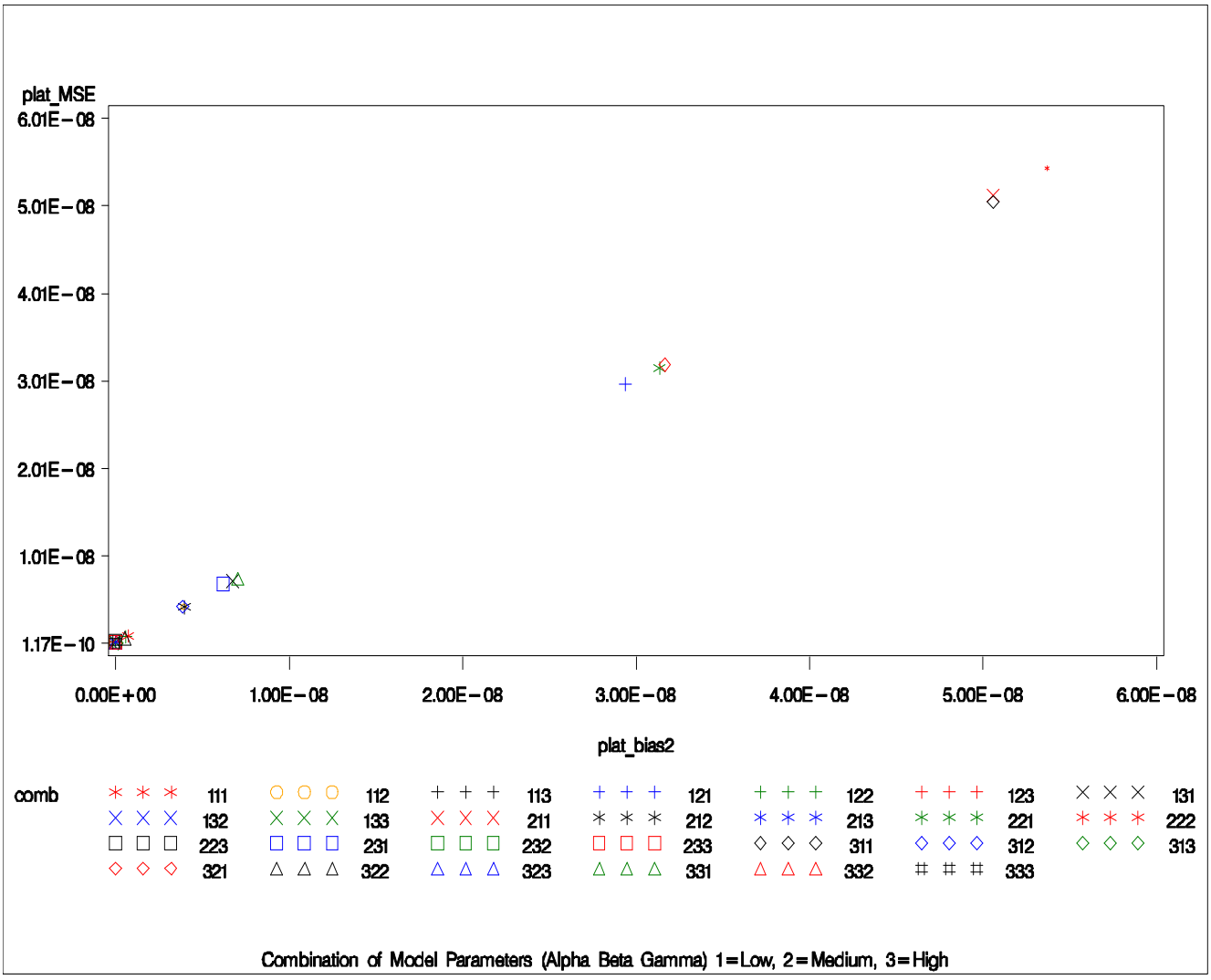



**Figure 3.7: Plot of bias and MSE of the estimated plateaus for the 27 combinations**

# Chapter 4

# Summary and Future Work

## 4.1 Summary

The purpose of this thesis was to use a non-linear statistical model to determine the mixing end point at which the simulated RI data reach a steady state or plateau in multistage liquids manufacture processing as quickly, accurately and reliably as possible. In this thesis, an algorithm that sequentially fits non-linear models and predicts the future observations was used to automatically estimate the steady state plateau. The proposed algorithm is significantly different from the other existing methods primarily because they do not fully utilize the mechanistic (functional) form of the process.

To evaluate the usefulness of the algorithm a simulation study was performed and the results were presented. The algorithm proves to be successful in accurately estimating the plateaus for most of the situations presented by the simulation study. The method is insensitive to changes in the asymptote but irrespective of level of Alpha; low Beta and low Gamma produces relatively large bias and MSE.

## 4.2 Future Work

The proposed method of non linear statistical algorithm is significantly different from the other existing methods. During the development of the proposed method, a variety of other research opportunities arose. Some of these are summarized here.

Two of the main limitations of the simulation study presented are the following. The proposed algorithm was not compared to the existing methods. Second the sensitivity of the algorithm to misspecifications of the non-linear function was not evaluated. The SAS software provides macros for implementing the EWMA and EWMV methods with the Western Electric Rules. In future, it would be useful to apply these methods to the data simulated and estimate the time at which these methods identify the plateau and compare them with the proposed algorithm. Similarly, one could consider other non-linear models such as exponential, logistic, and so on to estimate the time at which the plateau is identified for the data generated using the Gompertz model. Then evaluate the sensitivity of the algorithm to this misspecification of the non-linear function.

. Another limitation of the simulation study is, although the data were generated using autocorrelated errors, the fit of the non-linear model did not incorporate this serial correlation. This limitation exists primarily because the software used (SAS NLIN) does not include straightforward option for using the autocorrelation. Although the simulation study results seem to indicate robustness against this lack of independence, it would be useful to study whether the procedure converges faster (i.e. the plateau is identified earlier) by incorporating the autocorrelation.

Finally, the results presented here were all for monotonically increasing functions. The case of monotonically decreasing function also arises often in the pharmaceutical applications (Figure 4.1). Although the general methodology that works for the increasing function would easily adapt to the decreasing case there are certain peculiarities about the decreasing cases that might require further considerations. For instance, it is known that, during the last stage of mixing liquids, where it is highly variable across products, the algorithm might have to be customized.

Moreover, some of the processes have both increasing and decreasing stages and several of each. For instance, a process may have several stages, each of which may be increasing or decreasing and at each stage the process may reach a plateau. Extensions of the proposed method to incorporate these differences and perhaps simultaneously identify all the plateaus might be of interest.

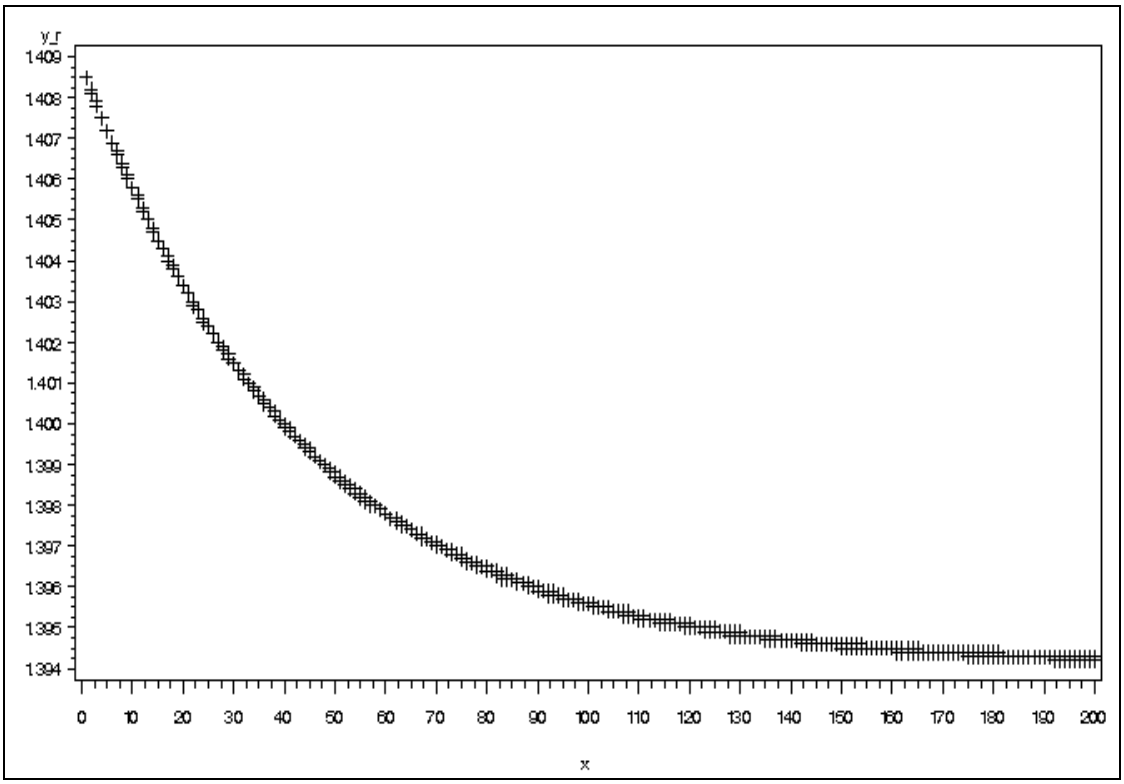



**Figure 4.1: Simulated monotonically descending function**

In summary, in this thesis a sequential algorithm based on consecutive fits of non-linear models was proposed for estimating the time at which a steady state in pharmaceutical processes is reached. The algorithm proposed has the potential to incorporate all aspects of the process, namely, the mechanistic model, the variability in the process and the serial correlations. One of the main advantages of this algorithm is, by identifying the plateau quickly, it could save a considerable amount of time in applications.

Often the purpose of identifying the plateau is to start a new process (perhaps by adding new materials to the substances being observed). The following Figure 4.2 shows how one could save time if the plateau is identified early. By implementing our algorithm in real time the savings could be materialized.

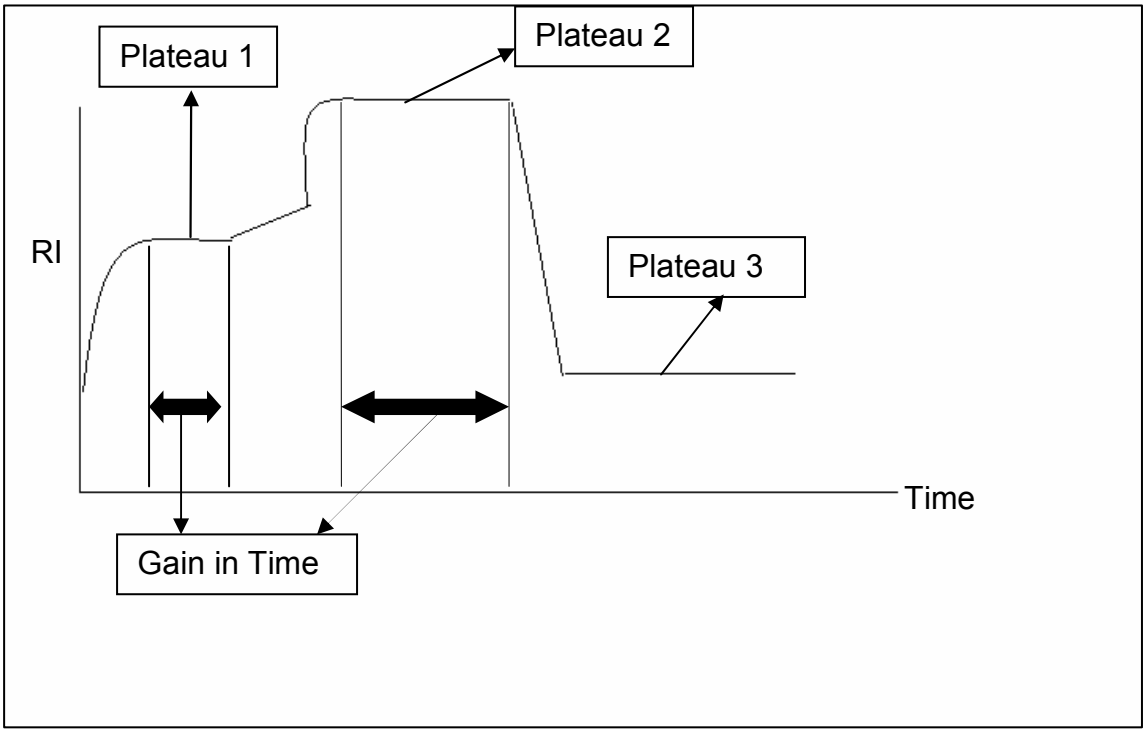



**Figure 4.2: An example of process showing gain in time if plateau is identified early**

References

Ahlfors, Lars V. (1953). *Complex Analysis*. McGraw-Hill Book Company, Inc.

Box, George, Jenkins Gwilym M., and Gregory C. Reinsel (1994). *Time Series Analysis: Forecasting and Control*, third edition. Prentice-Hall,

Mood, A., F. Graybill and D. Boes (1974). *Introduction to the Theory of Statistics,* Third edition, McGrawHill.

DaCosta, C.J. and Baenziger, J.E. et al (2003). A rapid method for assessing lipid: protein and detergent: protein ratios in membrane–protein crystallization*,* 59: 77–83

FDA (2002). Pharmaceutical cGMPs for the 21st Century: A Risk-Based Approach, U.S. Department of Health and Human Services, Rockville, MD.

FDA (2004). Guidance for Industry PAT - A Framework for Innovative Pharmaceutical Development, Manufacturing, and Quality Assurance, Office of Pharmaceutical Science in the Center for Drug Evaluation and Research (CDER), FDA, U.S. Department of Health and Human Services, Rockville, MD.

Fujii, Katsunori (2006). Connection between Growth / Development and Mathematical Function. International Journal of Sport and Health Science Vol.4, 216-232.

Gershenfeld, Neil (1999). *The Nature of Mathematical Modeling*, Cambridge, UK.: Cambridge University Press.

Hinz, Dirk C. (2005). Process Analytical Technologies in the pharmaceutical Industry: the FDA PAT initiative.384:1036-1042.

Keats, J.B. and Montgomery, D.C. (2001). *Statistical Quality Control,* John Wiley and Sons, New York.

Kenkel, John (2002). *Analytical Chemistry for Technicians*, Third Edition, Table 15.1, page 427.

Kotz, S. and Johnson, N.L. (1993). *Process Capability Indices,* Chapman and Hall.

Laird A. K. (1964). Dynamics of tumor growth. Br J of Cancer 18: 490–502.

Macgregor, J.F and Harris, T.J. (1993). *Exponentially Moving Variance, Journal of Quality Technology,* Volume 25, No 2.

Morgan, Joseph (1953).*Introduction to Geometrical and Physical Optics*, McGraw-Hill Book Company, Inc

Pearl R, Reed LJ. (1920) On the Rate of Growth of the Population of the United States since 1790 and Its Mathematical Representation. Proc Natl Acad Sci USA.;6(6):275–288

Roberts, (1959). *Control chart tests based on geometric moving averages*,Technometrics. v1. 239-250.

Refractive index**.** (2009). In Encyclopædia Britannica*.* Encyclopædia Britannica Online: http://www.britannica.com/EBchecked/topic/495677/refractive-index

*SAS® 9.3 Help and Documentation*, Cary, NC: SAS Institute Inc.

Strop, Pavel and Brunger, Axel T. (2005). *Refractive index-based determination of detergent concentration and its application to the study of membrane proteins*, Howard Hughes Medical Institute and Departments of Molecular and Cellular Physiology, Neurology and Neurological Sciences, Stanford Synchrotron Radiation Laboratory, Stanford University, James H. Clark Center, Stanford, California

Sweet A.L, (1986) Control Charts using coupled Exponentially Weighted Moving Averages, IEEE Transactions, 18: 26-33

Tague, Nancy R. (2004), *The Quality Toolbox*, Second Edition, ASQ Quality Press

US Pharmacopeia National Formulary, (2006), Rockville, Maryland

United States Department of Agriculture (www.usda.gov)

Urbani, A. and Warne, T. (2005). *A colorimetric determination for glycosidic and bile salt-based detergents: Applications in membrane protein research*. Anal. Biochem: 336, 117–124.

Wheeler, D J & Chambers, D S (1992) .*Understanding Statistical Process Control*, Addison-Wesley.

Winsor ,Charles P. (1932). The Gompertz Curve as a Growth Curve, Department of Biology, School of Hygiene and Public Health, Johns Hopkins University, Proc Natl Acad Sci USA; 18(1): 1–8

Wheldon, T.E. (1988). *Mathematical Models in Cancer Research*, Bristol: Adam hilger

# Appendix A

## Table A1: Levels for Model Parameter

| Model Parameter | Low(min) | Medium(median) | High(Max) |
|---|---|---|---|
| Alpha | 1.3890 | 1.3906 | 1.3990 |
| Beta | 5.9787 | 6.9432 | 9.0448 |
| Gamma | .0160 | .1199 | .5426 |

## Table A2: Combination Code for Model Parameters

| Group | Alpha | Beta | Gamma | Combination |
|---|---|---|---|---|
| 1 | Low | Low | Low | 111 |
| 2 | Low | Low | High | 113 |
| 3 | Low | Low | Medium | 112 |
| 4 | Low | High | Low | 131 |
| 5 | Low | High | High | 133 |
| 6 | Low | High | Medium | 132 |
| 7 | Low | Medium | Low | 121 |
| 8 | Low | Medium | High | 123 |
| 9 | Low | Medium | Medium | 122 |
| 10 | High | Low | Low | 311 |
| 11 | High | Low | High | 313 |
| 12 | High | Low | Medium | 312 |
| 13 | High | High | Low | 331 |
| 14 | High | High | High | 333 |
| 15 | High | High | Medium | 332 |
| 16 | High | Medium | Low | 321 |
| 17 | High | Medium | High | 323 |
| 18 | High | Medium | Medium | 222 |
| 19 | Medium | Low | Low | 211 |
| 20 | Medium | Low | High | 213 |
| 21 | Medium | Low | Medium | 212 |
| 22 | Medium | High | Low | 231 |
| 23 | Medium | High | High | 233 |
| 24 | Medium | High | Medium | 232 |
| 25 | Medium | Medium | Low | 221 |
| 26 | Medium | Medium | High | 223 |
| 27 | Medium | Medium | Medium | 222 |

## Appendix B

**Table B1: Simulation results sorted by lowest MSE**

| Obs | Alpha | Beta | Gamma | Plateau mean | Plateau variance | Plateau Bias$^2$ | Plateau MSE | mean # iteration | Variance # iteration | comb |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 1.3990 | 5.9787 | 0.5426 | 1.39901 | 0 | 1.1671E-10 | 1.1671E-10 | 1.00 | 0.0000 | 313 |
| **2** | 1.3890 | 9.0448 | 0.1199 | 1.38900 | 2.3963E-10 | 4.0203E-12 | 2.4365E-10 | 1.00 | 0.0000 | 132 |
| **3** | 1.3890 | 6.9432 | 0.5426 | 1.38900 | 2.6862E-10 | 5.0072E-15 | 2.6862E-10 | 1.00 | 0.0000 | 123 |
| **4** | 1.3906 | 9.0448 | 0.5426 | 1.39060 | 2.8946E-10 | 2.282E-14 | 2.8948E-10 | 1.00 | 0.0000 | 131 |
| **5** | 1.3890 | 9.0448 | 0.5426 | 1.38900 | 2.9236E-10 | 2.9581E-12 | 2.9532E-10 | 1.00 | 0.0000 | 133 |
| **6** | 1.3990 | 9.0448 | 0.5426 | 1.39900 | 2.9436E-10 | 2.6232E-12 | 2.9698E-10 | 1.00 | 0.0000 | 333 |
| **7** | 1.3990 | 6.9432 | 0.5426 | 1.39900 | 2.8883E-10 | 8.3252E-12 | 2.9716E-10 | 1.00 | 0.0000 | 323 |
| **8** | 1.3990 | 9.0448 | 0.1199 | 1.39900 | 2.8791E-10 | 1.4029E-11 | 3.0194E-10 | 1.00 | 0.0000 | 332 |
| **9** | 1.3906 | 9.0448 | 0.1199 | 1.39060 | 3.1106E-10 | 4.3813E-14 | 3.1111E-10 | 1.00 | 0.0000 | 232 |
| **10** | 1.3890 | 5.9787 | 0.5426 | 1.38900 | 3.1833E-10 | 7.9862E-13 | 3.1913E-10 | 1.00 | 0.0000 | 113 |
| **11** | 1.3906 | 5.9787 | 0.5426 | 1.39060 | 3.3355E-10 | 7.3605E-15 | 3.3356E-10 | 1.00 | 0.0000 | 213 |
| **12** | 1.3906 | 6.9432 | 0.5426 | 1.39060 | 3.5275E-10 | 1.6743E-12 | 3.5443E-10 | 1.00 | 0.0000 | 223 |
| **13** | 1.3990 | 6.9432 | 0.1199 | 1.39898 | 2.2735E-10 | 5.2742E-10 | 7.5477E-10 | 1.01 | 0.0100 | 222 |
| **14** | 1.3890 | 6.9432 | 0.1199 | 1.38898 | 3.0784E-10 | 5.9558E-10 | 9.0342E-10 | 1.01 | 0.0100 | 122 |
| **15** | 1.3906 | 6.9432 | 0.1199 | 1.39057 | 2.7848E-10 | 7.0819E-10 | 9.8667E-10 | 1.00 | 0.0000 | 222 |
| **16** | 1.3906 | 5.9787 | 0.1199 | 1.39054 | 3.2435E-10 | 3.97376E-9 | 4.29811E-9 | 1.19 | 0.2161 | 212 |
| **17** | 1.3890 | 5.9787 | 0.1199 | 1.38894 | 3.8949E-10 | 3.91803E-9 | 4.30751E-9 | 1.16 | 0.1560 | 112 |
| **18** | 1.3990 | 5.9787 | 0.1199 | 1.39894 | 4.855E-10 | 3.85203E-9 | 4.33754E-9 | 1.25 | 0.2904 | 312 |
| **19** | 1.3906 | 9.0448 | 0.0160 | 1.39052 | 7.4867E-10 | 6.20088E-9 | 6.94955E-9 | 3.83 | 12.0617 | 231 |
| **20** | 1.3890 | 9.0448 | 0.0160 | 1.38892 | 5.0806E-10 | 6.75481E-9 | 7.26287E-9 | 3.24 | 8.8105 | 131 |
| **21** | 1.3990 | 9.0448 | 0.0160 | 1.39892 | 5.0257E-10 | 7.03024E-9 | 7.53281E-9 | 3.27 | 8.5425 | 331 |
| **22** | 1.3890 | 6.9432 | 0.0160 | 1.38883 | 3.8436E-10 | 2.93761E-8 | 2.97604E-8 | 19.83 | 0.2839 | 121 |
| **23** | 1.3906 | 6.9432 | 0.0160 | 1.39042 | 2.5923E-10 | 3.1348E-8 | 3.16072E-8 | 19.89 | 0.1999 | 221 |
| **24** | 1.3990 | 6.9432 | 0.0160 | 1.39882 | 3.2737E-10 | 3.16492E-8 | 3.19765E-8 | 19.85 | 0.2096 | 321 |
| **25** | 1.3990 | 5.9787 | 0.0160 | 1.39878 | 0 | 5.05724E-8 | 5.05724E-8 | 28.00 | 0.0000 | 311 |
| **26** | 1.3906 | 5.9787 | 0.0160 | 1.39038 | 7.5772E-10 | 5.05977E-8 | 5.13554E-8 | 17.78 | 1.8905 | 211 |
| **27** | 1.3890 | 5.9787 | 0.0160 | 1.38877 | 7.6302E-10 | 5.36799E-8 | 5.44429E-8 | 28.44 | 1.4812 | 111 |

**Table B2: Simulation results sorted by lowest Bias**

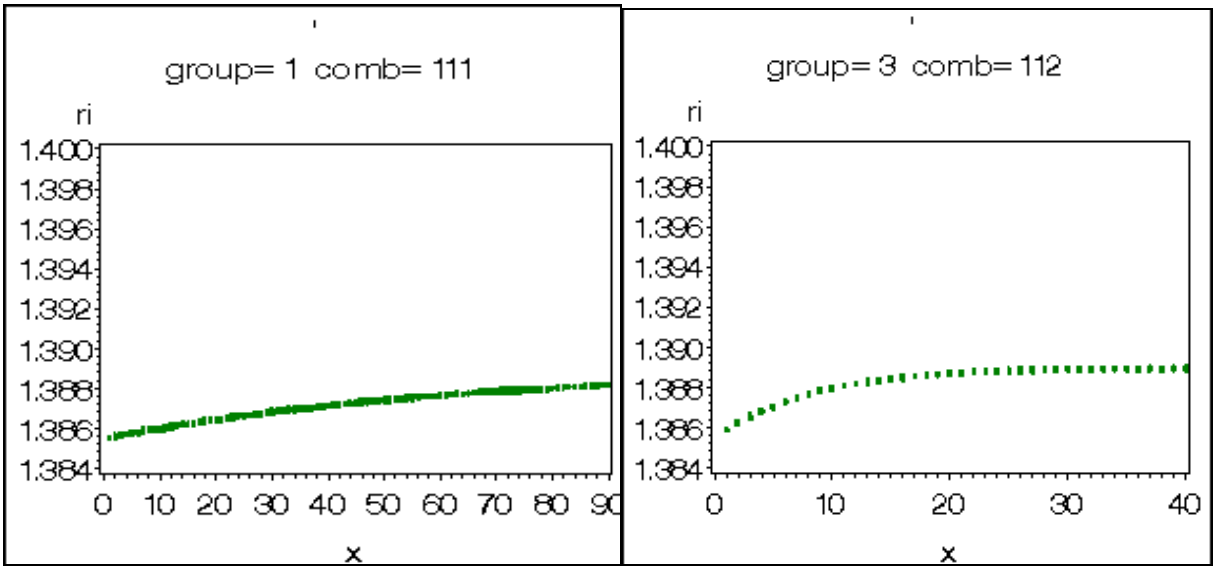
| Obs | Alpha | Beta | Gamma | Plateau mean | Plateau Variance | Plateau Bias$^2$ | Plateau MSE | Mean # iteration | Mean # Variance | Group |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 1.3890 | 6.9432 | 0.5426 | 1.38900 | 2.6862E-10 | 5.0072E-15 | 2.6862E-10 | 1.00 | 0.0000 | 123 |
| **2** | 1.3906 | 5.9787 | 0.5426 | 1.39060 | 3.3355E-10 | 7.3605E-15 | 3.3356E-10 | 1.00 | 0.0000 | 213 |
| **3** | 1.3906 | 9.0448 | 0.5426 | 1.39060 | 2.8946E-10 | 2.282E-14 | 2.8948E-10 | 1.00 | 0.0000 | 233 |
| **4** | 1.3906 | 9.0448 | 0.1199 | 1.39060 | 3.1106E-10 | 4.3813E-14 | 3.1111E-10 | 1.00 | 0.0000 | 232 |
| **5** | 1.3890 | 5.9787 | 0.5426 | 1.38900 | 3.1833E-10 | 7.9862E-13 | 3.1913E-10 | 1.00 | 0.0000 | 113 |
| **6** | 1.3906 | 6.9432 | 0.5426 | 1.39060 | 3.5275E-10 | 1.6743E-12 | 3.5443E-10 | 1.00 | 0.0000 | 223 |
| **7** | 1.3990 | 9.0448 | 0.5426 | 1.39900 | 2.9436E-10 | 2.6232E-12 | 2.9698E-10 | 1.00 | 0.0000 | 333 |
| **8** | 1.3890 | 9.0448 | 0.5426 | 1.38900 | 2.9236E-10 | 2.9581E-12 | 2.9532E-10 | 1.00 | 0.0000 | 133 |
| **9** | 1.3890 | 9.0448 | 0.1199 | 1.38900 | 2.3963E-10 | 4.0203E-12 | 2.4365E-10 | 1.00 | 0.0000 | 132 |
| **10** | 1.3990 | 6.9432 | 0.5426 | 1.39900 | 2.8883E-10 | 8.3252E-12 | 2.9716E-10 | 1.00 | 0.0000 | 323 |
| **11** | 1.3990 | 9.0448 | 0.1199 | 1.39900 | 2.8791E-10 | 1.4029E-11 | 3.0194E-10 | 1.00 | 0.0000 | 332 |
| **12** | 1.3990 | 5.9787 | 0.5426 | 1.39901 | 0 | 1.1671E-10 | 1.1671E-10 | 1.00 | 0.0000 | 313 |
| **13** | 1.3990 | 6.9432 | 0.1199 | 1.39898 | 2.2735E-10 | 5.2742E-10 | 7.5477E-10 | 1.01 | 0.0100 | 222 |
| **14** | 1.3890 | 6.9432 | 0.1199 | 1.38898 | 3.0784E-10 | 5.9558E-10 | 9.0342E-10 | 1.01 | 0.0100 | 122 |
| **15** | 1.3906 | 6.9432 | 0.1199 | 1.39057 | 2.7848E-10 | 7.0819E-10 | 9.8667E-10 | 1.00 | 0.0000 | 222 |
| **16** | 1.3990 | 5.9787 | 0.1199 | 1.39894 | 4.855E-10 | 3.85203E-9 | 4.33754E-9 | 1.25 | 0.2904 | 312 |
| **17** | 1.3890 | 5.9787 | 0.1199 | 1.38894 | 3.8949E-10 | 3.91803E-9 | 4.30751E-9 | 1.16 | 0.1560 | 112 |
| **18** | 1.3906 | 5.9787 | 0.1199 | 1.39054 | 3.2435E-10 | 3.97376E-9 | 4.29811E-9 | 1.19 | 0.2161 | 312 |
| **19** | 1.3906 | 9.0448 | 0.0160 | 1.39052 | 7.4867E-10 | 6.20088E-9 | 6.94955E-9 | 3.83 | 12.0617 | 231 |
| **20** | 1.3890 | 9.0448 | 0.0160 | 1.38892 | 5.0806E-10 | 6.75481E-9 | 7.26287E-9 | 3.24 | 8.8105 | 131 |
| **21** | 1.3990 | 9.0448 | 0.0160 | 1.39892 | 5.0257E-10 | 7.03024E-9 | 7.53281E-9 | 3.27 | 8.5425 | 331 |
| **22** | 1.3890 | 6.9432 | 0.0160 | 1.38883 | 3.8436E-10 | 2.93761E-8 | 2.97604E-8 | 19.83 | 0.2839 | 121 |
| **23** | 1.3906 | 6.9432 | 0.0160 | 1.39042 | 2.5923E-10 | 3.1348E-8 | 3.16072E-8 | 19.89 | 0.1999 | 221 |
| **24** | 1.3990 | 6.9432 | 0.0160 | 1.39882 | 3.2737E-10 | 3.16492E-8 | 3.19765E-8 | 19.85 | 0.2096 | 321 |
| **25** | 1.3990 | 5.9787 | 0.0160 | 1.39878 | 0 | 5.05724E-8 | 5.05724E-8 | 28.00 | 0.0000 | 311 |
| **26** | 1.3906 | 5.9787 | 0.0160 | 1.39038 | 7.5772E-10 | 5.05977E-8 | 5.13554E-8 | 17.78 | 1.8905 | 211 |
| **27** | 1.3890 | 5.9787 | 0.0160 | 1.38877 | 7.6302E-10 | 5.36799E-8 | 5.44429E-8 | 28.44 | 1.4812 | 111 |

**Table B3: Simulation results sorted by Highest Mean of number of Iterations**

| Obs | Alpha | Beta | Gamma | Plateau Mean | Plateau Variance | Plateau Bias | Plateau Mean | Mean # Iteration | Variance # Iteration | Group |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 1.389 | 5.9787 | 0.016 | 1.3887683108 | 7.630245E-10 | 5.3679888E-8 | 5.4442913E-8 | 28.44 | 1.4812121212 | 111 |
| **2** | 1.399 | 5.9787 | 0.016 | 1.3987751169 | 0 | 5.0572415E-8 | 5.0572415E-8 | 28 | 0 | 311 |
| **3** | 1.3906 | 6.9432 | 0.016 | 1.3904229465 | 2.592334E-10 | 3.1347951E-8 | 3.1607184E-8 | 19.89 | 0.1998989899 | 221 |
| **4** | 1.399 | 6.9432 | 0.016 | 1.3988220979 | 3.273729E-10 | 3.164916E-8 | 3.1976533E-8 | 19.85 | 0.2095959596 | 321 |
| **5** | 1.389 | 6.9432 | 0.016 | 1.3888286055 | 3.843646E-10 | 2.9376085E-8 | 2.976045E-8 | 19.83 | 0.2839393939 | 121 |
| **6** | 1.3906 | 5.9787 | 0.016 | 1.3903750607 | 7.577229E-10 | 5.0597683E-8 | 5.1355406E-8 | 17.78 | 1.8905050505 | 211 |
| **7** | 1.3906 | 9.0448 | 0.016 | 1.3905212543 | 7.486706E-10 | 6.2008787E-9 | 6.9495493E-9 | 3.83 | 12.061717172 | 231 |
| **8** | 1.399 | 9.0448 | 0.016 | 1.3989161535 | 5.025698E-10 | 7.0302402E-9 | 7.53281E-9 | 3.27 | 8.5425252525 | 331 |
| **9** | 1.389 | 9.0448 | 0.016 | 1.3889178123 | 5.080596E-10 | 6.754812E-9 | 7.2628716E-9 | 3.24 | 8.8105050505 | 131 |
| **10** | 1.399 | 5.9787 | 0.1199 | 1.3989379353 | 4.855048E-10 | 3.8520315E-9 | 4.3375363E-9 | 1.25 | 0.2904040404 | 312 |
| **11** | 1.3906 | 5.9787 | 0.1199 | 1.3905369623 | 3.243535E-10 | 3.9737551E-9 | 4.2981087E-9 | 1.19 | 0.2160606061 | 212 |
| **12** | 1.389 | 5.9787 | 0.1199 | 1.3889374058 | 3.894858E-10 | 3.9180277E-9 | 4.3075136E-9 | 1.16 | 0.155959596 | 112 |
| **13** | 1.389 | 6.9432 | 0.1199 | 1.3889755956 | 3.078403E-10 | 5.955755E-10 | 9.034158E-10 | 1.01 | 0.01 | 122 |
| **14** | 1.399 | 6.9432 | 0.1199 | 1.3989770345 | 2.273517E-10 | 5.274151E-10 | 7.547668E-10 | 1.01 | 0.01 | 222 |
| **15** | 1.389 | 5.9787 | 0.5426 | 1.3889991063 | 3.183279E-10 | 7.986212E-13 | 3.191265E-10 | 1 | 0 | 113 |
| **16** | 1.389 | 9.0448 | 0.5426 | 1.3889982801 | 2.923635E-10 | 2.95806E-12 | 2.953216E-10 | 1 | 0 | 133 |
| **17** | 1.389 | 9.0448 | 0.1199 | 1.3889979949 | 2.396321E-10 | 4.020342E-12 | 2.436524E-10 | 1 | 0 | 323 |
| **18** | 1.389 | 6.9432 | 0.5426 | 1.3890000708 | 2.686166E-10 | 5.007195E-15 | 2.686216E-10 | 1 | 0 | 123 |
| **19** | 1.399 | 5.9787 | 0.5426 | 1.3990108032 | 0 | 1.167085E-10 | 1.167085E-10 | 1 | 0 | 313 |
| **20** | 1.399 | 9.0448 | 0.5426 | 1.3990016196 | 2.943568E-10 | 2.623199E-12 | 2.9698E-10 | 1 | 0 | 333 |
| **21** | 1.399 | 9.0448 | 0.1199 | 1.3989962544 | 2.87913E-10 | 1.402949E-11 | 3.019425E-10 | 1 | 0 | 332 |
| **22** | 1.399 | 6.9432 | 0.5426 | 1.3989971147 | 2.888309E-10 | 8.325155E-12 | 2.97156E-10 | 1 | 0 | 323 |
| **23** | 1.3906 | 5.9787 | 0.5426 | 1.3905999142 | 3.335486E-10 | 7.360471E-15 | 3.33556E-10 | 1 | 0 | 213 |
| **24** | 1.3906 | 9.0448 | 0.5426 | 1.3906001511 | 2.894571E-10 | 2.281967E-14 | 2.894799E-10 | 1 | 0 | 233 |
| **25** | 1.3906 | 9.0448 | 0.1199 | 1.3905997907 | 3.110639E-10 | 4.381342E-14 | 3.111077E-10 | 1 | 0 | 232 |
| **26** | 1.3906 | 6.9432 | 0.5426 | 1.390601294 | 3.527536E-10 | 1.674348E-12 | 3.544279E-10 | 1 | 0 | 223 |
| **27** | 1.3906 | 6.9432 | 0.1199 | 1.3905733881 | 2.784779E-10 | 7.081924E-10 | 9.866703E-10 | 1 | 0 | 222 |

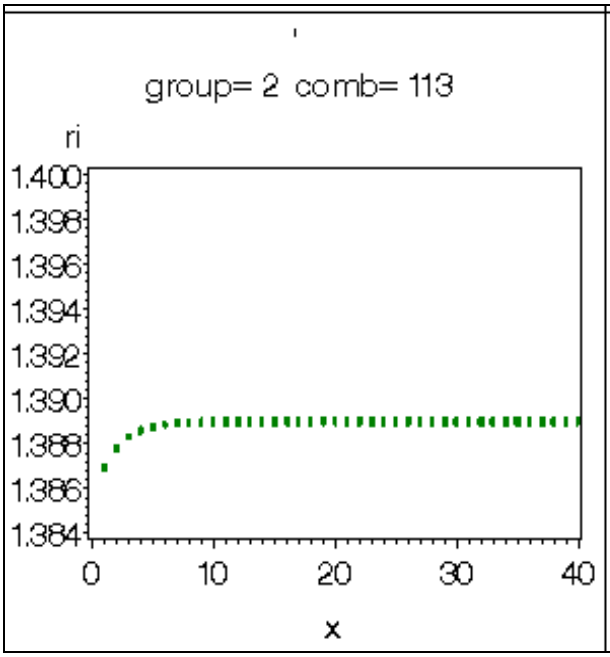
## Appendix C

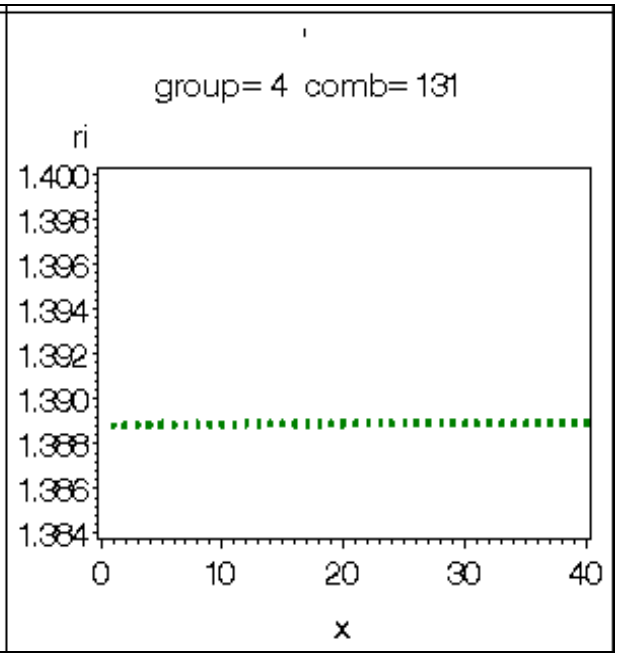
**The plots of estimated plateau for all 27 combinations of model parameters**

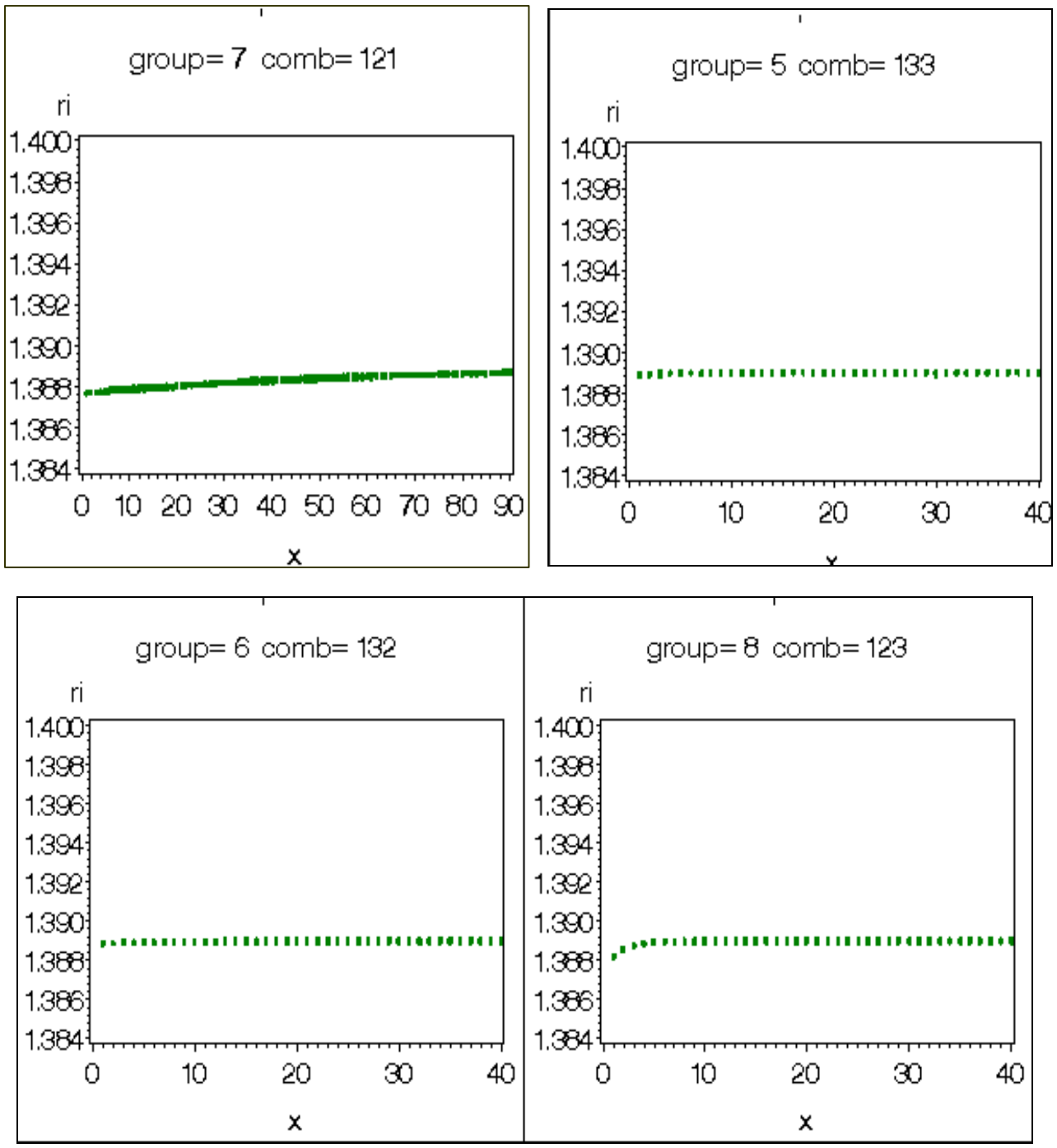


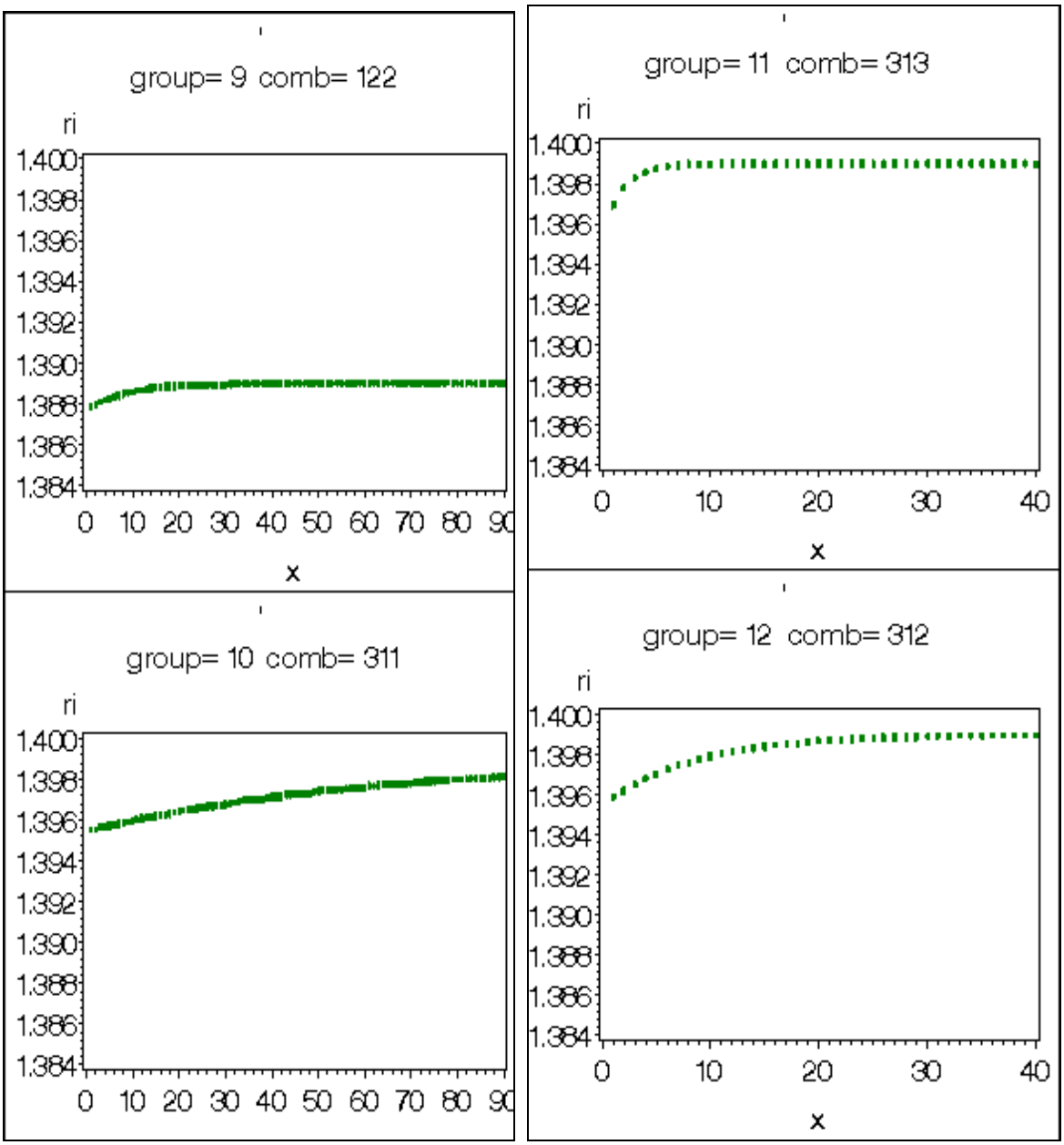


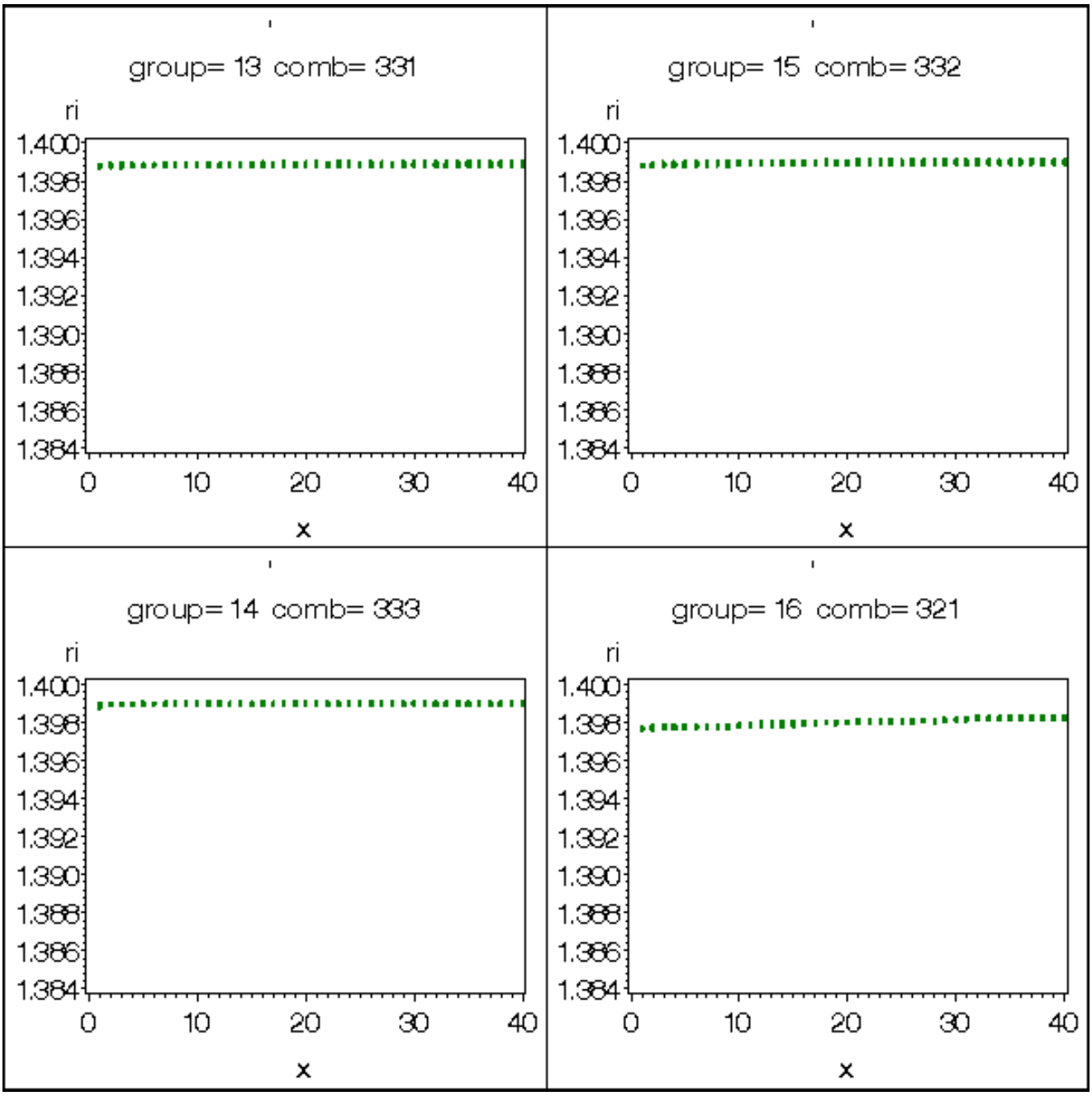


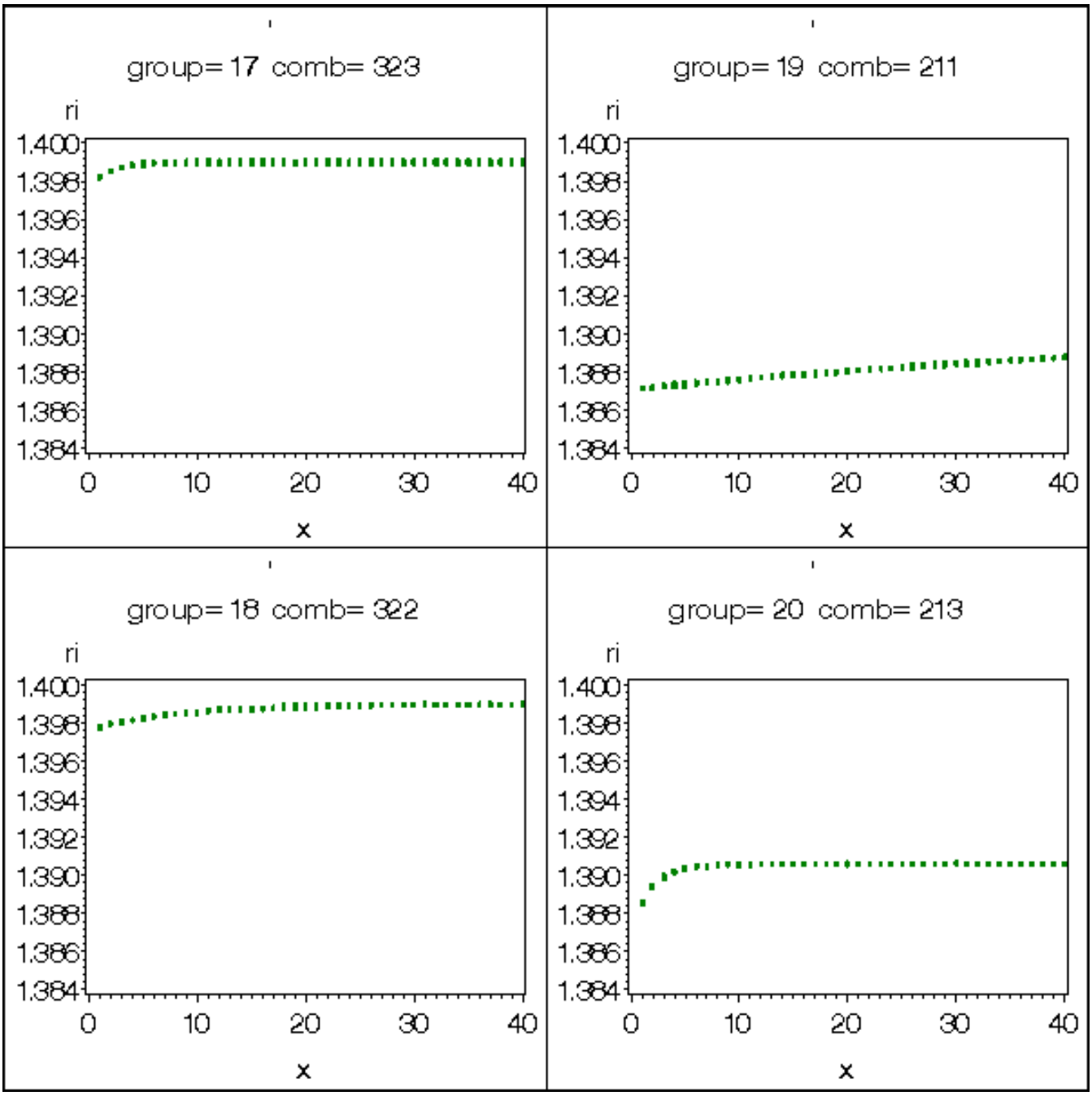


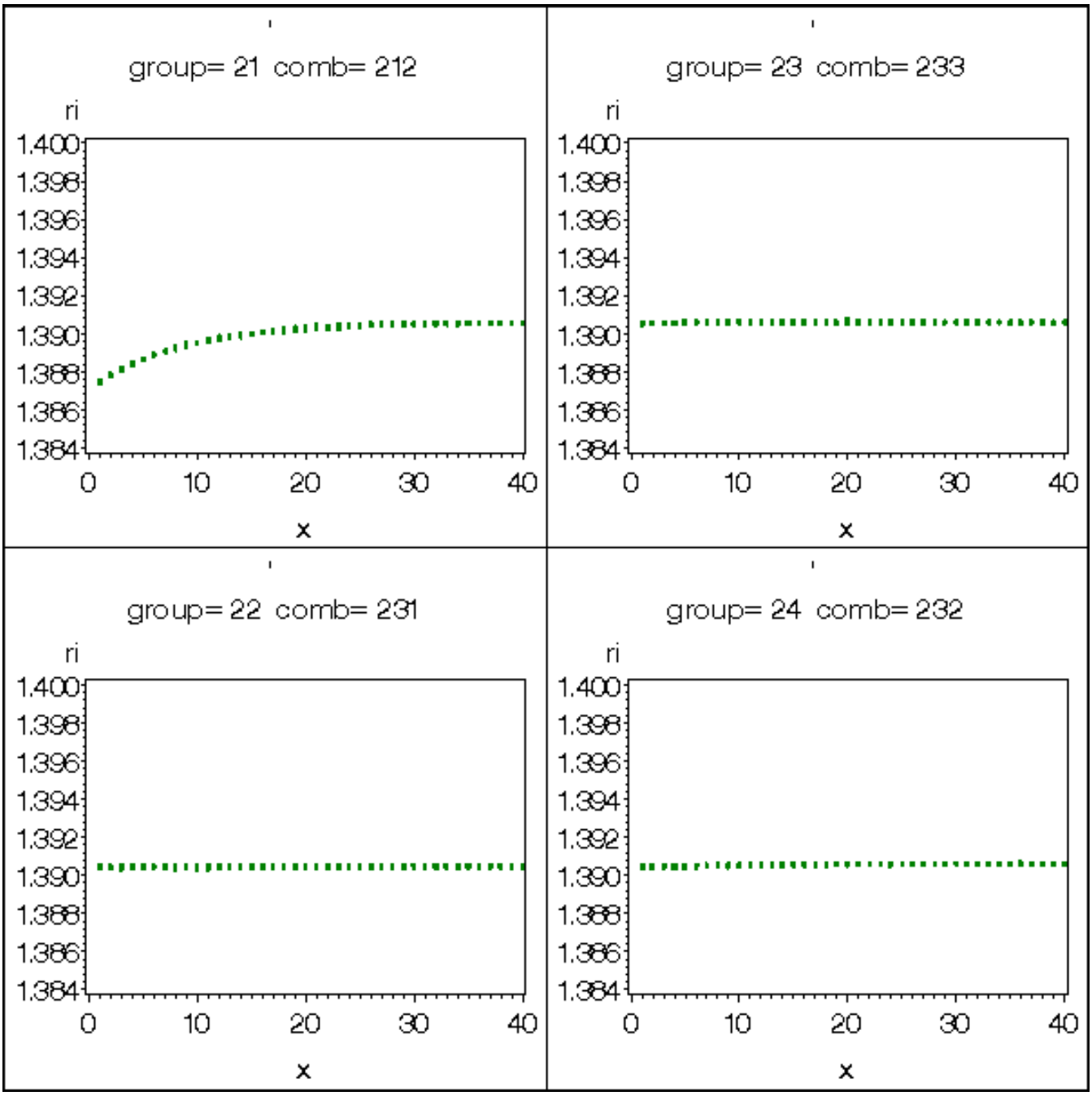



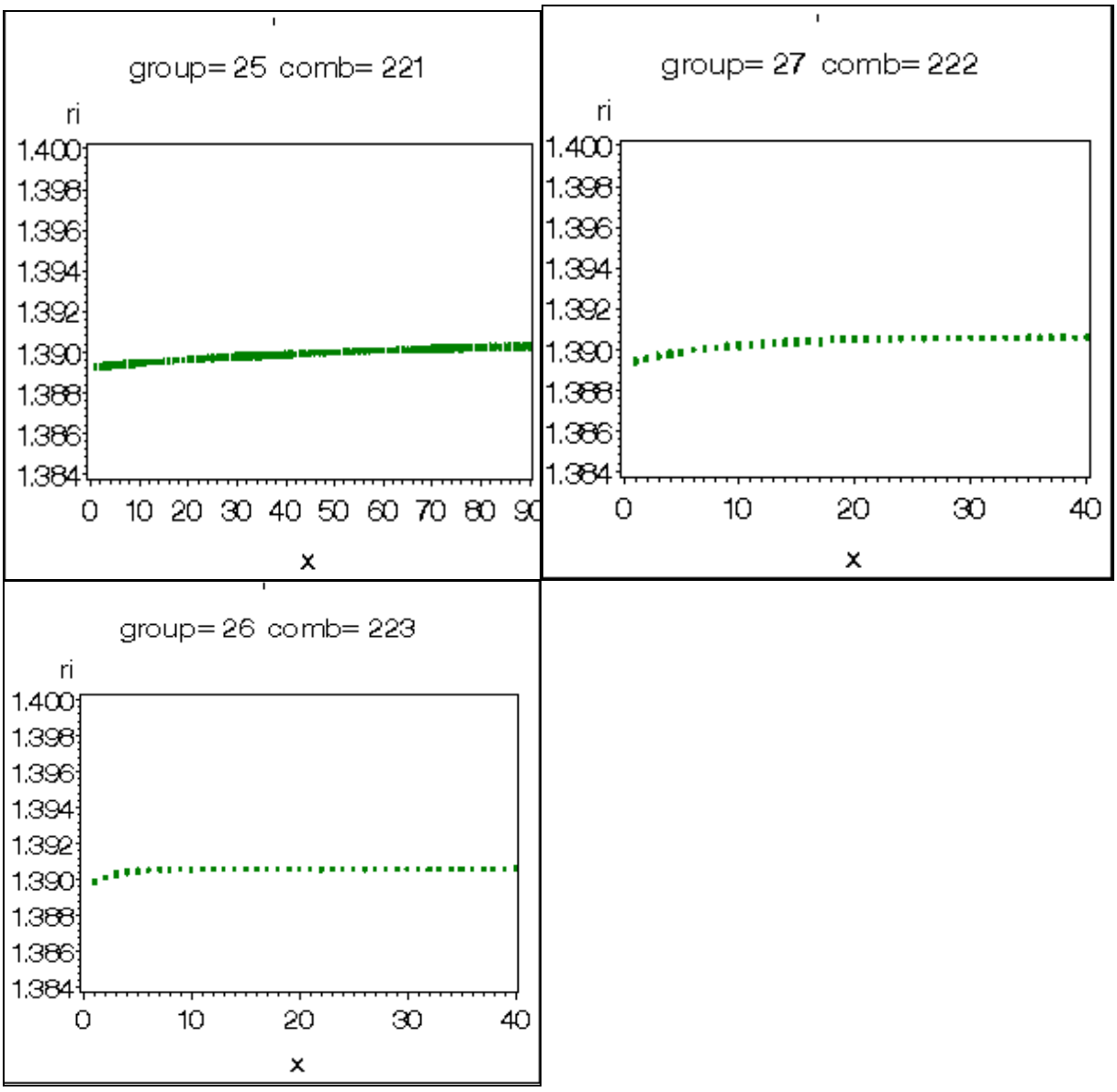

group= 7 comb= 121
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40 50 60 70 80 90
x
group= 5 comb= 133
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40
x
group= 6 comb= 132
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40
x
group= 8 comb= 123
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40
x

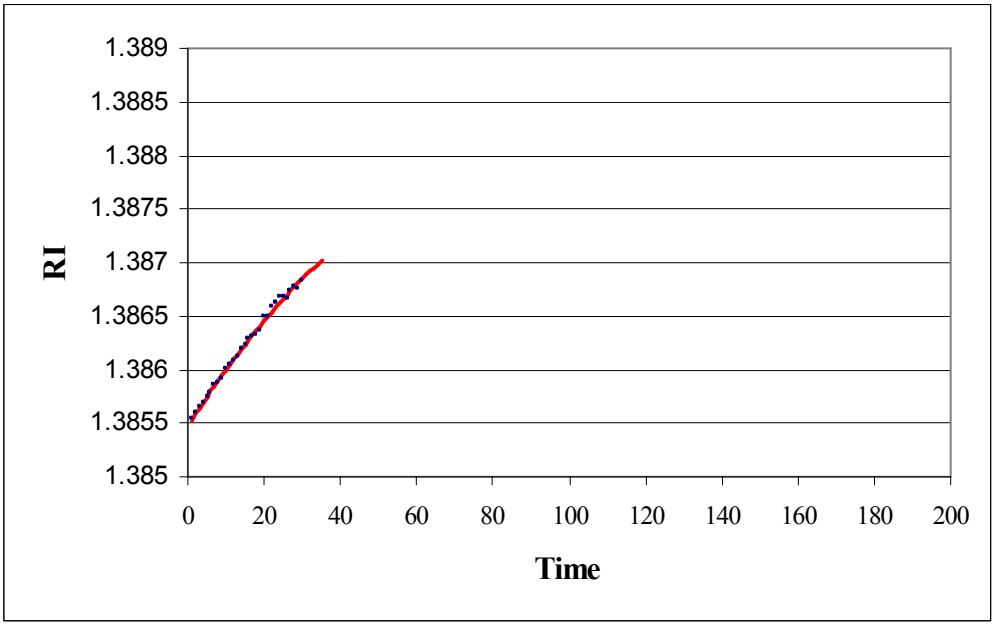

group= 9 comb= 122
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40 50 60 70 80 90
x
group= 11 comb= 313
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40
x
group= 10 comb= 311
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40 50 60 70 80 90
group= 12 comb= 312
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40
x


group= 13 comb= 331
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x
group= 15 comb= 332
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x
group= 14 comb= 333
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x
group= 16 comb= 321
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x

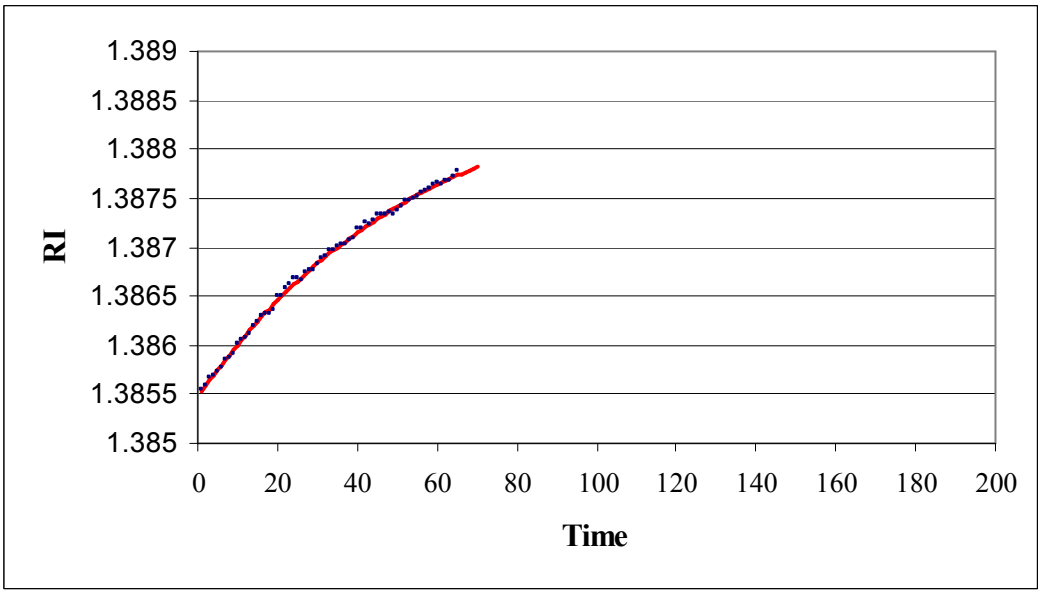

group= 17 comb= 323
group= 19 comb= 211
group= 18 comb= 322
group= 20 comb= 213
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x


group= 21 comb= 212
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x
group= 23 comb= 233
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x
group= 22 comb= 231
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x
group= 24 comb= 232
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0
10
20
30
40
x


group= 25 comb= 221
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40 50 60 70 80 90
x
group= 27 comb= 222
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40
x
group= 26 comb= 223
ri
1.400
1.398
1.396
1.394
1.392
1.390
1.388
1.386
1.384
0 10 20 30 40
x

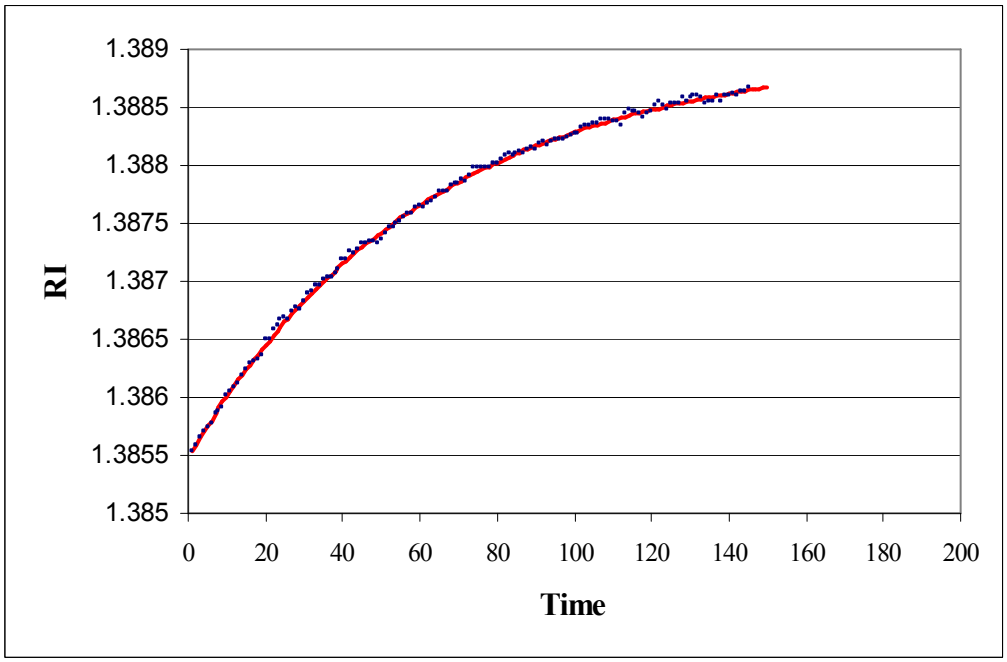



a) Iteration 1

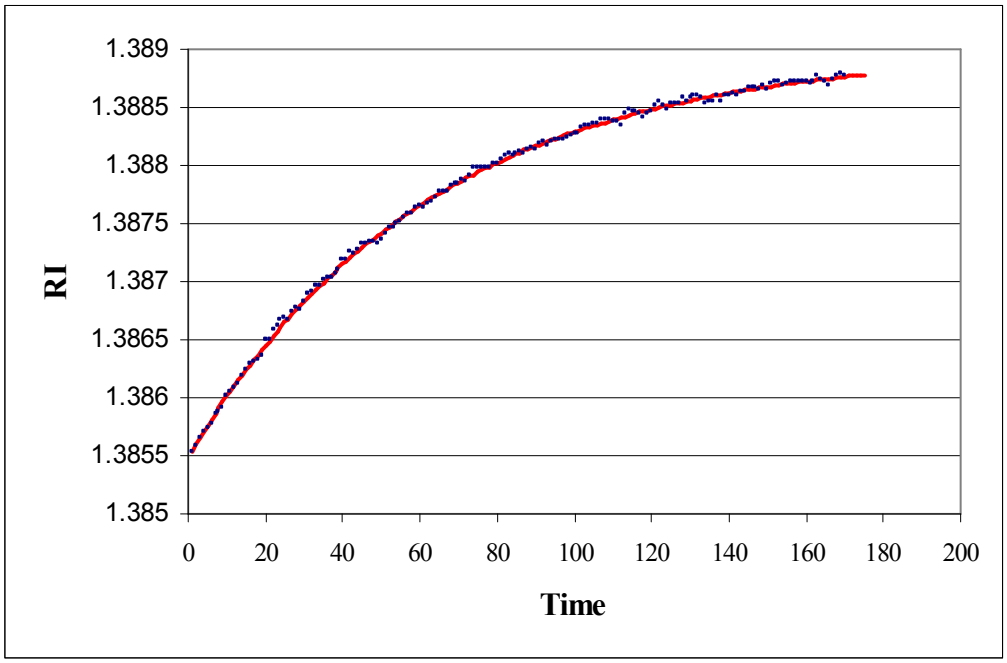



b) Iteration 8

**Figure C.1.a)-b): Progression of the Iterations**

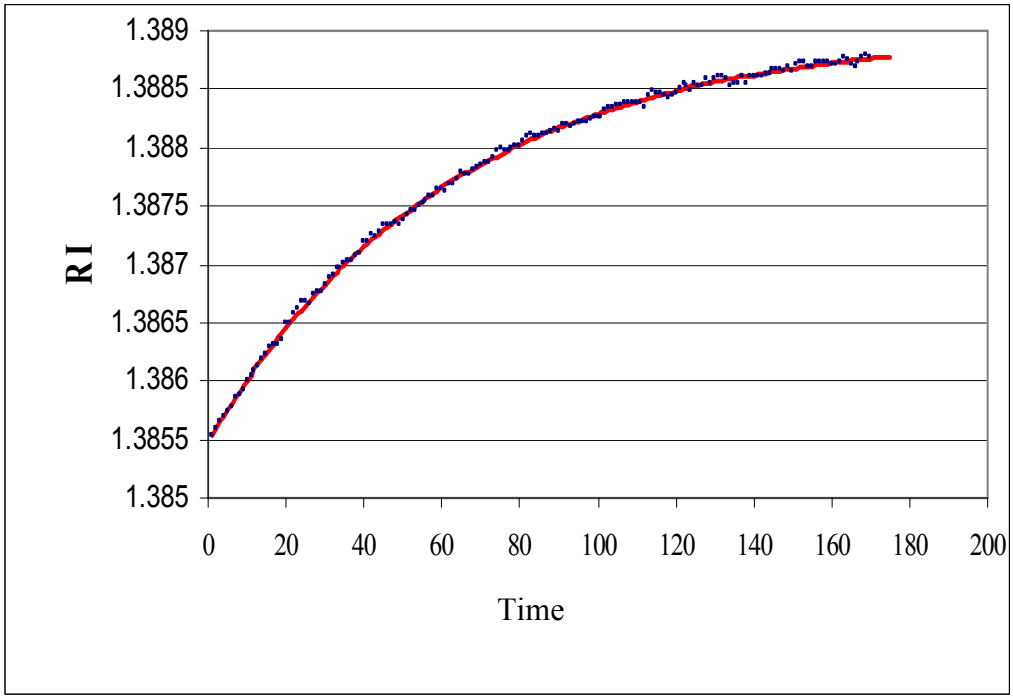



c) Iteration 16

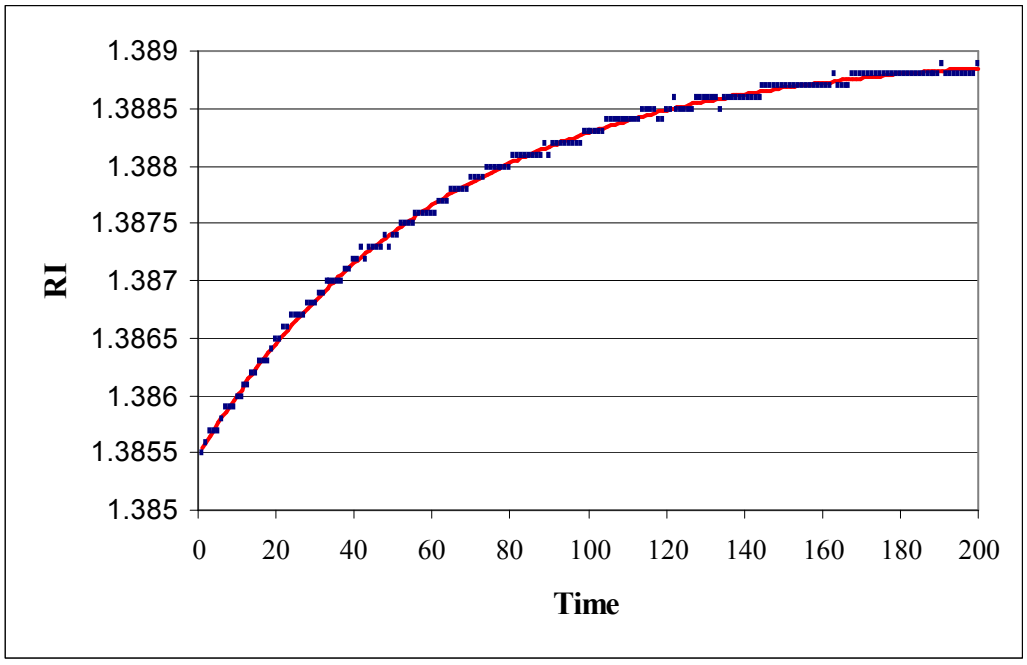



d) Iteration 24

**Figure C.1c)-d): Progression of the Iterations**




e) Iteration 29




f) Final iteration

**Figure C.1e)-f) Progression of the Iterations**

# Appendix D

D1: SAS codes for generating simulated data

```
%let _prognam=Y15_20060182; ** SAS program name;
%let _output1=&_prognam;
%let _byvars = product product_code batch plateau ;
%let _footnote= "&_prognam-Statistical Project &_projnum-&_reqnum  ";

/*Plateau 1-positive function*/

%let _alphavali=%str(1.3890,1.3990,1.3906);*(min,max,median for Alpha);
%let _betavali=%str(5.9787,9.0448,6.9432); *(min,max,median for Beta);
%let _gammavali=%str(.0160,.5426,.1199); *(min,max,median for Gamma);
%let _arvali=%str(.5160); *(median for Ar(1);
%let _msevali=%str(.000000000485); *(median for MSE);

 options merror mprint mlogic symbolgen ;
libname library "C:\20060182";

data simulation_inc;
group=0;
dir=1;
do  alpha=&_alphavali;
    do beta=&_betavali;
       do gamma=&_gammavali;
         do ar=&_arvali;
           do mse=&_msevali;
                group=group+1;
               do i=1 to 100;*# of profiles generated using each comb;
                   e=0;
                  do x=1 to 200; *number of points on each plateau;
              z = rannor(1011 );*generate standard normal random value;
                  u=sqrt(mse)*z;  *scale with mse;
                  e = ar*e + u ;*generate autocorrelated errors;

      /*Generate gompertz function*/

              Y = alpha*(1-exp(-beta-(gamma*x))) ;
ri=y+e;*value of RI(function+error);
y_r = round(ri, 0.0001);
                    output;
                            end;
                        end;
                     end;
               end;
         end ;
   end;
end;
run;
```

```
proc sort data=simulation_inc;
by group;
run;

/*For negative funtions plateau 2 and 5*/

%let _alphavald=%str(1.3732,1.3941,1.3921); *(min,max,median);
%let _betavald=%str(4.5508,9.1550,7.3702);
%let _gammavald=%str(.0230,.3714,.1002);

%le1t _arvald=%str(.522);
%let _msevald=%str(.00000000149);
 options merror mprint mlogic symbolgen ;

data simulation_dec;
group=0;
dir=2;
do  alpha=&_alphavald;
 do beta=&_betavald;
  do gamma=&_gammavald;
   do ar=&_arvald;
    do mse=&_msevald;
    group=group+1;
      do i=1 to 100; *# profiles generated using each combination;
                                        e=0;
             do x=1 to 200; *number of points on each plateau;
             z = rannor(1011 );*generate standard normal random value;
             sqrt(mse)*z;  *scale with mse;
             e = ar*e + u ;*generate autocorrelated errors;
  Y = alpha*(1+exp(-beta-(gamma*x))) ; /*Generate gompertz function*/
  ri=y+e;*value of RI(function+error);
  y_r = round(ri, 0.0001);
                                  output;
                             end;
                        end;
                   end;
              end;
         end ;
   end;
end;
run;
proc sort data=simulation_dec;
by group;
run;
quit;


data simulation;
set simulation_inc simulation_dec;
run;

MAKE PERMANENT DATA SET;
data library.&_prognam;
 set simulation;
  run;
```

## D2: SAS Codes for Fitting Non Linear Model And Estimating Asymptote

```
libname library "C:\20060182\programs";
  %let  _alphavali=%str(1.3890,1.3990,1.3906);*(min,median,max  for  alpha)  ;
  %let _betavali=%str(5.9787,9.0448,6.9432);*( min,median,max for Beta) ;
  %let _gammavali=%str(.0160,.5426,.1199); *(min,median,max for gamma) ;

data work;
  set library.y15_20060182;
  run;

options mlogic mprint symbolgen;
*creating k for alpha,l for beta and m for gamma as indicator variable with
1=low,2=medium and 3=high;

%macro test;

%let groupc=0;
%let groupc=25;
  %do  k=1 %to 3;
      %do  l=1 %to 3;
          %do m=1 %to 3;
                      %do n=1 %to 100;
                            %do j=5 %to 100 %by 5;*selecting initial 25 points
and making increments of 5;
                           %let _count=25+&j;

* Selection of data;
    data work1;
    set work;
    where group=&groupc and  i=&n and dir=1 and x<=200;
    if _n_<= &_count then y_r1 = ri;
    else y_r1=.;
    niter=&j;
    run;
*fitting Gompertz function and predicting asymptote;

    proc nlin data=work1 maxiter=32000 noprint ;
    parms alpha1= 1   beta1= 1 gamma1=1;
    model y_r1 =alpha1*(1-exp(beta1-gamma1*x)) ;
    output out=predict_alpha_&k.&l.&m. p=p_y;
    run;

*pulling last RI value;

     Data fix_ri_&k.&l.&m.;
     Set predict_alpha_&k.&l.&m end=lastobs;
     If lastobs then output;
     Run;

     Proc sql noprint;
     Select ri into:&var2 from fix_ri_&k.&l.&m.;*selectinf last RI into var2;
     Quit;
```

```
     %put &var2;



      data tolerance_&k.&l.&m.;
      set predict_alpha_&k.&l.&m.;
      diffy_r=(&var2-p_y)**2;   * computing prediction error muhat;
      if _n_>&_count and _n_<=&_count+5;
      output;
      run;

*computing means of prediction errors and predicted RI;
     proc means data=tolerance_&k.&l.&m. noprint ;
     var niter y_r diffy_r;
     output out= Mdiffy_r_&groupc. mean=nitr meany_r diff var=vnit vri vdiff;
     run;

     data mdiff_&groupc.;
     set Mdiffy_r_&groupc.;
     diffa=diffy_r;
      run;

     proc sql noprint;
     select diffa into : var1
     from mdiff_&groupc.;
     quit;
    %put &var1;

    data Mdiffy_r_&groupc.;
     set mdiffy_r_&groupc.;
    k =&k;
    l=&l;
    m=&m;
    n=&n;
    groupc=&groupc;
    run;

*recording the last observation for each iteration when condition is met and
appending all those in one dataset;
    %if &j=1 %then %do;
    proc sql;
    create table new&n. like mdiffy_r_&groupc.;
    quit;

    proc append base=new&n. data=mdiffy_r_&groupc.;
    run;
    %end;
    %else %do;
    proc append base=new&n. data=mdiffy_r_&groupc.; run;
    %end;

*tolerance level tau condition;
    %if %sysevalf(&var1<.000000009)%then %do;
    %put Condition met;
```

```
    %goto next;
                    %end;
                              %end;
%next:

**Append data sets;
proc append base=mdiffy_r_main_&groupc. data=mdiffy_r_&groupc.;
 run;
                 %end;
             %end;
        %end;
 %end;

*create permanent library;
data library.inc_&groupc;
set mdiffy_r_main_&groupc.;
run;
%mend ;

 %test;
```

# VITA

Akriti Saxena was born on June 28, 1979 in the city of Lucknow, Uttar Pradesh, India. She had her schooling from Carmel High School, Lucknow and stood at 15th rank in the State High School Merit list, 1995. She completed her Intermediate school from Loreto Convent, Lucknow, 1997. She received her Bachelor of Science (Mathematics Statistics and Economics) from University of Lucknow, 2000. She did a Post Graduate course in Garment Manufacturing Technology from National Institute of Fashion Technology, New Delhi, India, 2002. She worked as a merchandiser in India's largest export house Orient Craft, Gurgaon .She then began pursuing her Masters of Science in Operations Research from Virginia Commonwealth University, Richmond, Virginia, 2005 and after graduation continued with a Masters of Science in Biostatistics at Medical College of Virginia of Virginia Commonwealth University.

During her academic life, she has worked as a Research Assistant in Virginia Commonwealth University and also as a Teaching Assistant for undergraduate as well as graduate level statistics classes. She has received Student award from Glaxo Smithkline in 2005.She was also selected as a student intern at Wyeth Consumer Healthcare, Richmond, VA.