Development of a PC Software Package Using Windows 95 and Visual C++ to Evaluate Traffic Safety Improvements Based Upon Accidents Per Unit Time

A Thesis Presented to

The Faculty of the
Fritz J. and Dolores H. Russ
College of Engineering and Technology
Ohio University

In Partial Fulfillment
of the Requirement for the Degree
Master of Science

by
Kuan Tao Yu
November 1996
ACKNOWLEDGMENTS

I would like to express my sincere appreciation to my advisor, Dr. Helmut T. Zwahlen for his guidance, help, and advice throughout this thesis. I would also like to acknowledge the technical input in statistics and quality control from Dr. Richard J. Gerth. I also wish to thank Dr. Lloyd Herman for serving on the committee.

Finally, I wish to thank my parents and friends for their support.
TABLE OF CONTENTS

ACKNOWLEDGMENT ............................................................................................................... III

TABLE OF CONTENTS ........................................................................................................ IV

LIST OF FIGURES ................................................................................................................. VI

LIST OF TABLES .................................................................................................................. VIII

CHAPTER 1 INTRODUCTION ................................................................................................. 1
  1.1 Statement of the Problem ............................................................................................... 1
  1.2 Objectives .................................................................................................................... 5

CHAPTER 2 LITERATURE REVIEW ....................................................................................... 7
  2.1 Literature Review of Statistical Methods ..................................................................... 7
    2.1.1 Conventional Cumulative Sum Chart .................................................................. 7
    2.1.2 Time Series Decomposition .............................................................................. 11
    2.1.3 Winter's Exponential Smoothing ....................................................................... 15
    2.1.4 Linear Exponential Smoothing .......................................................................... 17
    2.1.5 Runs Test Technique ......................................................................................... 18
  2.2 Literature Review of Software Design ......................................................................... 20
    2.2.1 Traditional Methodologies ................................................................................. 20
    2.2.2 Object-Oriented Approach ................................................................................. 23

CHAPTER 3 SYSTEM DESIGN ............................................................................................... 26
  3.1 Assumptions ................................................................................................................. 26
  3.2 Determining Trend and Periodicity Existence .......................................................... 29
    3.2.1 Tests for Trend .................................................................................................. 29
    3.2.2 Tests for Seasonality ......................................................................................... 30
  3.3 Selecting Appropriate Time Series Method .................................................................. 32
  3.4 Standardization of Cumulative Sum Chart .............................................................. 33
  3.5 Graphical Interpretation of the CUSUM Charts ....................................................... 38
  3.6 More Explanation of Std. Shewhart-CUSUM Chart ............................................... 42
  3.7 Tabular CUSUM Chart ............................................................................................. 48
  3.8 System Requirements ............................................................................................... 50
    3.8.1 Hardware Requirements ................................................................................. 50
    3.8.2 Software Requirements ................................................................................... 50
LIST OF FIGURES

Figure 1.1 Standardized Shewhart-CUSUM Chart with a Hypothetical Shift ............. 2
Figure 2.1 Common Components of a Time Series .............................................. 12
Figure 2.2 Waterfall Model of the Software Life Cycle ...................................... 22
Figure 3.1 System Model Work Flow .............................................................. 27
Figure 3.2 Construction of the Standardized CUSUM Chart .............................. 37
Figure 3.3 Shift Identification Flow Chart .................................................... 41
Figure 3.4 Standardized Shewhart-CUSUM Chart of Accidents by Months ........ 43
Figure 3.5 Revised Standardized Shewhart-CUSUM Chart ................................. 45
Figure 3.6 Standardized Shewhart-CUSUM Chart with a Hypothetical Shift ....... 45
Figure 4.1 Hiding Data Members and Function Members in the Classes .......... 55
Figure 4.2 Example of Usage of Polymorphism Mechanism ............................. 58
Figure 4.3 Example of Usage of Inheritance Mechanism ................................ 58
Figure 4.4 Communication Between Classes Through Messages or Events ....... 60
Figure 4.5 Finding Relations Among Classes .................................................. 66
Figure 4.6 Partial Class Hierarchy in a Tree-View Structure .............................. 67
Figure 4.7 Class Hierarchy of TSIE in a Flow Chart ....................................... 68
Figure 5.1 K-S Test for Daytime at Passive RRx’s (Non-Conrail) ...................... 77
Figure 5.2 K-S Test for Nighttime at Passive RRx’s (Non-Conrail) ...................... 77
Figure 5.3  K-S Test for Daytime at Passive RRx's (Conrail Only).......................... 78
Figure 5.4  K-S Test for Nighttime at Passive RRx’s (Conrail Only) ......................... 78
Figure 5.5  Time Series Plot for Daytime at Passive RRx's (Non-Conrail) ................. 79
Figure 5.6  Time Series Plot for Nighttime at Passive RRx’s (Non-Conrail) ............... 79
Figure 5.7  Time Series Plot for Daytime at Passive RRx’s (Conrail Only) ............... 79
Figure 5.8  Time Series Plot for Nighttime at Passive RRx’s (Conrail Only) .............. 80
Figure 5.9  Tabular CUSUM Chart for Daytime at Passive RRx’s (Non-Conrail) ......... 86
Figure 5.10 Tabular CUSUM Chart for Nighttime at Passive RRx’s (Non-Conrail) .... 86
Figure 5.11 Tabular CUSUM Chart for Daytime at Passive RRx’s (Conrail Only) ....... 87
Figure 5.12 The Performance of Std. Shewhart Control Chart (Alpha = 5%) .......... 90
Figure 5.13 The Performance of Tabular CUSUM Chart (H=5, K=0.2, Z=±1.65) ....... 91
Figure 5.14 The Performance of Std. Shewhart-CUSUM Chart (α = .05, Z=±1.65) ... 91
Figure 5.15 Performance Comparison for a 5% Down Shift in Mean ....................... 92
Figure 5.16 Performance Comparison for a 10% Down Shift in Mean ....................... 92
Figure 5.17 Performance Comparison for a 20% Down Shift in Mean ....................... 93
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1.1</td>
<td>Standardized Cusums of Accidents</td>
<td>3</td>
</tr>
<tr>
<td>Table 3.1</td>
<td>Analysis of Variance for Testing Significance of Regression</td>
<td>30</td>
</tr>
<tr>
<td>Table 3.2</td>
<td>An Example of Rank Test</td>
<td>31</td>
</tr>
<tr>
<td>Table 3.3</td>
<td>A Quick Reference to Select Time Series Methods</td>
<td>32</td>
</tr>
<tr>
<td>Table 3.4</td>
<td>Trend Line Identification Rules</td>
<td>39</td>
</tr>
<tr>
<td>Table 3.5</td>
<td>Standardized Cusums of Accidents by Months</td>
<td>44</td>
</tr>
<tr>
<td>Table 3.6</td>
<td>Revised Standardized Cusums of Accidents</td>
<td>46</td>
</tr>
<tr>
<td>Table 3.7</td>
<td>12 Points Data Added to the Standardized CUSUM Chart</td>
<td>47</td>
</tr>
<tr>
<td>Table 3.8</td>
<td>Necessary Files to Run TSIE Program</td>
<td>51</td>
</tr>
<tr>
<td>Table 4.1</td>
<td>Properties List Provided by a SpreadView Class</td>
<td>65</td>
</tr>
<tr>
<td>Table 5.1</td>
<td>Accident Data for Daytime at Passive RRx’s (Non-Conrail)</td>
<td>76</td>
</tr>
<tr>
<td>Table 5.2</td>
<td>Accident Data for Nighttime at Passive RRx’s (Non-Conrail)</td>
<td>76</td>
</tr>
<tr>
<td>Table 5.3</td>
<td>Accident Data for Daytime at Passive RRx’s (Conrail Only)</td>
<td>76</td>
</tr>
<tr>
<td>Table 5.4</td>
<td>Accident Data for Nighttime at Passive RRx’s (Conrail Only)</td>
<td>76</td>
</tr>
<tr>
<td>Table 5.5</td>
<td>Analysis of Variance for Daytime (Non-Conrail)</td>
<td>80</td>
</tr>
<tr>
<td>Table 5.6</td>
<td>Analysis of Variance for Nighttime (Non-Conrail)</td>
<td>80</td>
</tr>
<tr>
<td>Table 5.7</td>
<td>Analysis of Variance for Daytime (Conrail Only)</td>
<td>80</td>
</tr>
<tr>
<td>Table 5.8</td>
<td>Analysis of Variance for Nighttime (Conrail Only)</td>
<td>80</td>
</tr>
</tbody>
</table>
Table 5.9  Tabular Cusums for Daytime at Passive RRx’s (Non-Conrail) .................. 81

Table 5.10  Tabular Cusums for Nighttime (Non-Conrail) ................................. 83

Table 5.11  Tabular Cusums for Daytime (Conrail Only) ................................. 85

Table 5.12  Monthly Accidents with the First Set of 30 Simulated Data ................. 90
Chapter 1

Introduction

1.1 Statement of the Problem

Car accidents have been a major cause of death in the last several decades (Accident Facts 1992). However, research has contributed to a decrease in these accidents by identifying, developing, and evaluating traffic safety improvements which have guided safety programs. In the effort to reduce traffic accidents from motor vehicles, more empirical and extensive statistical techniques need to be employed in analyzing the data. Appropriate use of statistical techniques can greatly increase the efficiency of these processes and often strengthen the conclusions so that results are more objective and reliable. Unfortunately, once raw data has been collected, sequential procedures such as data entry, data calculation and chart drawing can be elaborate and difficult to accomplish. Hence, a more effective approach with new software technology that applies object-oriented design to embellish statistical methods is in great demand.

In this thesis the analyst investigates the reduction of the number of car accidents per month after the implementation of a traffic safety improvement. Suppose that one wishes to arrive at a conclusion concerning the number of accidents and determine whether or not a reduction is caused by the safety improvement. The analyst can consider the
generation of accidents as a process and apply statistical methods to control this process. Thus, two modified CUSUM schemes were constructed in this thesis for the application in a traffic safety program. Monthly accident data was collected and calculated to a cumulative sum for the last three years, as shown in Table 1.1. A special cause, traffic safety improvement, was added to the system starting in January, 1991. The analyst’s concern is to know as soon as possible whether or not this improvement is causing a significant shift from the mean level. Figure 1.1 illustrates a supposed circumstance for applying the standardized Shewhart-CUSUM Chart. The accident data was plotted graphically against the time period with a hypothetical shift.

![Standardized Shewhart-CUSUM Chart with a Hypothetical Shift](image)

**Figure 1.1** Standardized Shewhart-CUSUM Chart with a Hypothetical Shift

Unfortunately, many accidents often result from a complex chain of events that cannot be adequately described by the simple model above. Furthermore, these data bases represent time series data that are assumed to contain some possible components such as seasonal, cyclical, and trend patterns. These patterns are considered to be a problem due to the lack of independence of observations in a time series. It is highly
recommended that one should deduct these causes and smooth the data series before applying any CUSUM chart. By deseasonalizing or detrending the time series data, for instance, one can do data analysis more accurately in the accident reduction efforts.

Table 1.1 Standardized Cusums of Accidents

<table>
<thead>
<tr>
<th>Month</th>
<th>Data $X_i$</th>
<th>Normalized Data $Z_i = \frac{X_i - \bar{X}}{S_z}$</th>
<th>Sum $\sum_{i=1}^{i} Z_i$</th>
<th>CUSUM $C_i = \frac{\sum_{i=1}^{i} Z_i}{\sqrt{i}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>35</td>
<td>0.29</td>
<td>0.29</td>
<td>0.29</td>
</tr>
<tr>
<td>Feb</td>
<td>31</td>
<td>-0.09</td>
<td>0.20</td>
<td>0.14</td>
</tr>
<tr>
<td>Mar</td>
<td>37</td>
<td>0.48</td>
<td>0.68</td>
<td>0.39</td>
</tr>
<tr>
<td>Apr</td>
<td>37</td>
<td>0.48</td>
<td>1.16</td>
<td>0.58</td>
</tr>
<tr>
<td>May</td>
<td>39</td>
<td>0.68</td>
<td>1.84</td>
<td>0.82</td>
</tr>
<tr>
<td>Jun</td>
<td>40</td>
<td>0.77</td>
<td>2.61</td>
<td>1.07</td>
</tr>
<tr>
<td>Jul</td>
<td>45</td>
<td>1.25</td>
<td>3.86</td>
<td>1.46</td>
</tr>
<tr>
<td>Aug</td>
<td>43</td>
<td>1.06</td>
<td>4.92</td>
<td>1.74</td>
</tr>
<tr>
<td>Sep</td>
<td>42</td>
<td>0.96</td>
<td>5.88</td>
<td>1.96</td>
</tr>
<tr>
<td>Oct</td>
<td>43</td>
<td>1.06</td>
<td>6.94</td>
<td>2.20</td>
</tr>
<tr>
<td>Nov</td>
<td>38</td>
<td>0.58</td>
<td>7.52</td>
<td>2.27</td>
</tr>
<tr>
<td>Dec</td>
<td>40</td>
<td>0.77</td>
<td>8.29</td>
<td>2.39</td>
</tr>
<tr>
<td>Jan-90</td>
<td>34</td>
<td>0.19</td>
<td>8.49</td>
<td>2.35</td>
</tr>
<tr>
<td>Feb</td>
<td>30</td>
<td>-0.19</td>
<td>8.30</td>
<td>2.22</td>
</tr>
<tr>
<td>Mar</td>
<td>35</td>
<td>0.29</td>
<td>8.59</td>
<td>2.22</td>
</tr>
<tr>
<td>Apr</td>
<td>37</td>
<td>0.48</td>
<td>9.07</td>
<td>2.27</td>
</tr>
<tr>
<td>May</td>
<td>40</td>
<td>0.77</td>
<td>9.84</td>
<td>2.39</td>
</tr>
<tr>
<td>Jun</td>
<td>42</td>
<td>0.96</td>
<td>10.81</td>
<td>2.55</td>
</tr>
<tr>
<td>Jul</td>
<td>44</td>
<td>1.16</td>
<td>11.96</td>
<td>2.74</td>
</tr>
<tr>
<td>Aug</td>
<td>45</td>
<td>1.25</td>
<td>13.21</td>
<td>2.95</td>
</tr>
<tr>
<td>Sep</td>
<td>41</td>
<td>0.87</td>
<td>14.08</td>
<td>3.07</td>
</tr>
<tr>
<td>Oct</td>
<td>40</td>
<td>0.77</td>
<td>14.85</td>
<td>3.17</td>
</tr>
<tr>
<td>Nov</td>
<td>37</td>
<td>0.48</td>
<td>15.33</td>
<td>3.20</td>
</tr>
<tr>
<td>Dec</td>
<td>36</td>
<td>0.39</td>
<td>15.72</td>
<td>3.21</td>
</tr>
<tr>
<td>Jan-91</td>
<td>22</td>
<td>-0.96</td>
<td>14.76</td>
<td>2.95</td>
</tr>
<tr>
<td>Feb</td>
<td>20</td>
<td>-1.15</td>
<td>13.61</td>
<td>2.67</td>
</tr>
<tr>
<td>Mar</td>
<td>19</td>
<td>-1.25</td>
<td>12.37</td>
<td>2.38</td>
</tr>
<tr>
<td>Apr</td>
<td>21</td>
<td>-1.05</td>
<td>11.31</td>
<td>2.14</td>
</tr>
<tr>
<td>May</td>
<td>19</td>
<td>-1.25</td>
<td>10.07</td>
<td>1.87</td>
</tr>
<tr>
<td>Jun</td>
<td>21</td>
<td>-1.05</td>
<td>9.01</td>
<td>1.65</td>
</tr>
<tr>
<td>Jul</td>
<td>18</td>
<td>-1.34</td>
<td>7.67</td>
<td>1.38</td>
</tr>
<tr>
<td>Aug</td>
<td>17</td>
<td>-1.44</td>
<td>6.23</td>
<td>1.10</td>
</tr>
<tr>
<td>Sep</td>
<td>15</td>
<td>-1.63</td>
<td>4.60</td>
<td>0.80</td>
</tr>
<tr>
<td>Oct</td>
<td>17</td>
<td>-1.44</td>
<td>3.16</td>
<td>0.54</td>
</tr>
<tr>
<td>Nov</td>
<td>16</td>
<td>-1.53</td>
<td>1.63</td>
<td>0.28</td>
</tr>
<tr>
<td>Dec</td>
<td>15</td>
<td>-1.63</td>
<td>0</td>
<td>0.00</td>
</tr>
</tbody>
</table>
A cumulative sum chart, which has its origins in statistical quality control, can be applied for the detection of shifts in a traffic device improvement case. It is essential in one’s observations to know the dynamics of the changes in the cumulative sum of the deviations of traffic accidents. Generally speaking, a trend line with a sharp increase or decrease in the slope that persists for several samples is an indication of a possible shift in the process. The standardized Shewhart-CUSUM chart is designed for the purpose of monitoring the movement of the dynamic deviation. An upward or downward trend in the data series indicates that the cumulative sum has shifted and was caused by the treatment (traffic safety improvement). Furthermore, by setting additional control limits, standardized Shewhart-CUSUM charts can be designed to detect the right shift faster under a particular average run length (expected to have a ARL = 10 or less).

As an alternative, another CUSUM scheme, tabular CUSUM chart, was developed from a different approach. This tabular CUSUM can be designed as competitive as a standardized Shewhart-CUSUM chart. A comparison of speed in detecting small mean shifts was carried out for these CUSUM schemes in this thesis. However, since the dynamic nature of a changing cumulative deviation is an integral part of the process characteristics, it is difficult to keep track of these cumulative deviations in time without applying computer software technology.

Traditionally, software development is based on a procedure-oriented design. After identifying the problems to be solved, the process is to decompose the problems into smaller, more specific problems. As this process continues and the project becomes too
large, the program becomes very complex and difficult to maintain. For example, a software project usually needs hundreds of functions for data entry, data calculations, presentation charting, and output printing. It seems impossible to keep the program size small without considering the program structure. In this project, a larger program with thousands of functions is required.

Creation of such applications usually requires careful attention in the software construction. Making programs smarter with high automation is one of the final goals. Recent advances in computer software techniques offer opportunities for the creation of large-scale application design systems that were once thought to be impossible or impractical. Furthermore, these systems require frequent modifications and extensions because of continuing developments in the real world. However, in this thesis a new, object-oriented methodology will be introduced for the improvement of software packages to evaluate traffic safety improvements.

1.2 Objectives

The primary objective of this thesis is to apply an object-oriented design technique to develop a software package for statistical data analysis. This software will provide the following features:
• The ability to run programs under the Microsoft Windows 95 operating system which support menus, icons, dialog boxes, scroll bars, and other components of user-friendly interface.

• The ability to communicate and share data with other window applications such as Excel and Lotus 1-2-3, based on the MS Windows 95 operating system environment.

• The ability to plot accidents per unit time (monthly, yearly) against time in a time series based chart so that one can visually determine if any seasonal or trend patterns exist.

• The ability to construct both standardized Shewhart-CUSUM chart and tabular CUSUM chart in order to detect the correct shift under a certain ARL value.

• The ability to provide a clean hard copy of all raw data and satisfying results.

• The ability to carry out some related statistical data analysis such as Time-Series Decomposition, Holt’s Exponential Smoothing, or Winter’s Exponential Smoothing in order to deseasonalize or detrend the data series.
Chapter 2

Literature Review

2.1 Literature Review of Statistical Methods

2.1.1 Conventional Cumulative Sum Chart

The cumulative sum chart was first practiced by E. S. Page (1954) for quality control purposes, but there were not many practical applications in the manufacturing workplace until five years ago. A cusum is a series of numbers which are the cumulative sum of the differences between an observed series and the corresponding expected series (Woodward and Goldsmith 1964). Another basis of the CUSUM scheme is that successive values of a variable are compared with a predetermined target or reference value as described by A. F. Bissell (1968). The cumulative sum of deviations from this target value is then plotted on a chart against the time. If the accumulation reaches or exceeds a predetermined decision interval (or V-mask control), this indicates that a shift has occurred in the mean level of the variable.

A summary for this chart technique was illustrated by Juran (1974). The following lists show the details of the cumulative sum control chart technique with a typical V-mask control limit for controlling the average:
1. **Typical problems for application.** Tightly controlling a process by detecting moderate shifts in the average with a minimum number of tests due to high costs of a shift or high test costs.

2. **Characteristic plotted.** Cumulative sum of deviation of subgroup average from a reference value or a target value.

3. **Formulas for control limits.**

   A) Obtain an estimate of $\sigma_i$.

   B) Determine the least amount of change $D$ in the average that it is desired to detect, and calculate $\delta = \frac{D}{\sigma_i}$.

   C) Determine the probability level at which decisions are to be made. For limits equivalent to standard control limits, $\alpha = 0.00135$.

   D) Define the scale factor $k$ as the vertical scale distance per unit of the horizontal scale.

   E) Obtain an estimate of lead distance $d$.

   F) Calculate $\frac{D}{k}$ and read the angle $\theta$ of the mask. Enter at row with $\delta = \frac{D}{k}$.

   G) Using $d$ and $\theta$ construct the mask to define the control limits.

4. **Subgroup number.** The guidelines for $\bar{X}$ and $R$ charts apply, but for best results Ewan suggests $n = \frac{2.25s^2}{D}$ and an interval between subgroups of $\frac{T}{6}$ where $T$ is the permissible average time before the shift of $D$ is detected.

5. **Some common patterns of plots.** The mask is placed over the last point plotted. If any of the previous points are covered by the mask, a process shift has occurred.
Points covered by the top of the mask mean a decrease in the average; the bottom of
the mask detects an increase. The first point covered by the mask indicates the time
at which the shift in average started. If all the points are exposed by the mask, the
process is considered in control.

6. **Statistical distribution assumption.** The population of individual measurements is
assumed to be normally distributed.

The cumulative sum chart is very useful in this project and its major advantages are
as follows:

- The cumulative sum chart is much more sensitive and effective in detecting
  small shifts in the mean than Shewhart-type control charts (Duncan 1974,
  Montgomery 1985).

- It provides additional information about the possible time of the actual shift in
  the process (Devor 1992). With such a CUSUM plot in the chart, it is possible
to know precisely when the shift started and when it ended.

- It provides a much clearer pictorial representation of the behavior of the
  process. A time series plot of cumulative sums can provide an easily read visual
  impression of the historical record.

- It is possible to construct a cumulative sum chart without a target value in
  the mean to detect the shifts.
In the past, the cumulative sum is merely the accumulation of differences from the target value. Then, the cumulative sum control chart is plotted against time:

\[ C_t = \sum_{i=1}^{t} (\bar{x}_i - T) \]  \hspace{1cm} (2.1)

thus

\[ C_1 = (\bar{x}_1 - T) \]
\[ C_2 = (\bar{x}_1 - T) + (\bar{x}_2 - T) = C_1 + (\bar{x}_2 - T) \]
\[ C_3 = (\bar{x}_1 - T) + (\bar{x}_2 - T) + (\bar{x}_3 - T) = C_2 + (\bar{x}_3 - T) \]
\[ \vdots \]
\[ C_t = C_{t-1} + (\bar{x}_t - T) \]

where \( C_t \) is called the cumulative sum, \( t \) is the sample size and \( T \) is the target value. \( \bar{x}_i \) represents the average of the \( i \)th subgroup. However, this control chart has been accepted for process control purposes in the manufacturing and chemical industries. The statistics are the sample means or sample variances in the last two decades.

Recently, the cumulative sum chart has been utilized effectively for plotting individual observations such as traffic accidents. Let \( X_1, X_2, X_3, \ldots, X_t \) be individual observations observed up to the sample \( t \) under the assumption that the data sets follow a normal distribution. The cumulative sum can be expressed by
\[ C_i = \sum_{i=1}^{t} (X_i - T) = C_{i-1} + (X_i - T) \tag{2.2} \]

where \( T \) is also the target value, and \( C_i \) is a cumulative sum of the deviations of individual observations from the target. If a process is in control and the true process individual measurement is equal to \( T \), then the cumulative sums, \( C_i \), will be approximately zero and fluctuate about \( T \) without exhibiting any linear trends. However, if the individual observations of the deviations are off target value by a slight amount, \( \delta \), then the plot of the CUSUM will add \( \delta \) with each observation. The plot of the CUSUM will increase or decrease, depending on the sign of \( \delta \). If there is only a moderate change in the observations, a relatively large change in the slope of the CUSUM chart can result, since each new observation has a chance of contributing a shift and the measure being plotted is accumulating these shifts (Walpole & Myers 1989). The analyst watches for a change in the slope of the CUSUM plot as an indication of a shift. For a sustained shift in process, the trend is linear. Therefore, in examining a CUSUM chart, it is important to look for trend lines in a series of cumulative sums.

### 2.1.2 Time Series Decomposition

Time series data are usually collected on a monthly, quarterly or annual basis. Such time series data provide important information when plotted graphically. One way of thinking about the behavior of an actual observed series is to regard it as being made up
of various components. Traditionally, four possible components have been considered in a time series data: seasonal, cyclical, trend and irregular patterns (Brown 1963). Some of the combinations which can occur are presented graphically in Figure 2.1.

![Figure 2.1 Common Components of a Time Series](image)

Decomposition methods were developed in the 1920's in an attempt to identify and control the business cycle by economists. The approach of decomposition methods is to isolate the three components: seasonal, trend, and cycle patterns. There is no way that a statistical method can automatically determine the best pattern to describe a given set of data series and the decision must be based on judgment (Makridakis & Wheelwright 1978). Normally, it is preferable that one should first identify the seasonal patterns and then trend patterns in time series data. Second, one can determine whether or not to deseasonalize or to detrend them. These components can be a nuisance if the analyst is
only interested in the influence of other factors. Consequently, the succeeding step is to assess how much of the increase is due to purely seasonal factors and how much represents real underlying growth. Thus, both seasonal factors and trend can be removed from the data series and other effects can be concentrated.

There are a number of different methods to deseasonalize and detrend in a time series analysis. The major one is the classical time-series decomposition model that uses the concepts of moving averages (MA). The general mathematical representation of the decomposition approach is

\[ X = T \times C \times S \times I \]  \hspace{1cm} (2.3)

where

\[ X = \text{Actual Value} \]

\[ T = \text{Trend Component}: \text{Exhibits a tendency to grow or to decrease fairly steadily over a long period of time.} \]

\[ C = \text{Cyclical Component}: \text{Exhibits a regular up-and-down pattern of the variable within a recurring time period like a week or month.} \]

\[ S = \text{Seasonal Component}: \text{Exhibits cycles that tend to have a longer period than a seasonal component.} \]
The first step in working with this model is to remove the short-term fluctuations from the data series so that the long-term trend and cycle components can be more clearly identified. These short-term fluctuations include both seasonal patterns and irregular variations. They can be removed by calculating an appropriate moving average for the series. The moving average should contain the same number of periods as there are in the seasonal patterns that need to be identified. For instance, if data series are gathered quarterly and the analyst suspects seasonal patterns on a quarterly basis, a four-period moving average is appropriate. If data series are collected monthly and the analyst identifies a monthly pattern, a twelve-period moving average should be applied. The moving average is the basis of most of the methods for estimating the seasonal factors. In recent years, an even more improved method, the center moving average, has been applied to replace the conventional moving average.

After deseasonalizing the data series, the analyst can visually determine if a long-term trend exists on the chart. Such long-term trends can be linear or curvilinear. The simplest form of the trend is a straight line. This linear trend can be described by a linear equation in the form of \( y = a + bx \), where the parameters \( a \) and \( b \) can be determined by the equations using the method of least squares. Consequently, after eliminating the trend factor, a new straight line without any trend can be plotted by calculating appropriate \( y \).
values and x values. Thus, the trend component can be assessed and the detrend procedure can be developed on demand.

2.1.3 Winter’s Exponential Smoothing

The second method of identifying seasonal and trend patterns is Winter’s Exponential Smoothing, which was developed by P. R. Winters in the early 1960’s. It is a three parameter model that can be described in the form of equations:

\[ F_t = \alpha \frac{X_t}{I_{t-p}} + (1 - \alpha)(F_{t-1} + T_{t-1}) \]  
(2.4)

\[ S_t = \beta \frac{X_t}{F_t} + (1 - \beta)S_{t-p} \]  
(2.5)

\[ T_t = \gamma (F_t - F_{t-1}) + (1 - \gamma)T_{t-1} \]  
(2.6)

where

- \( F_t \) = Smoothed Value
- \( X_t \) = Actual Value
- \( T_{t+1} \) = Trend Estimate
- \( S_t \) = Seasonal Estimate
- \( \alpha \) = Smoothing Constant for the Data (0 < \( \alpha < 1 \))
- \( \beta \) = Smoothing Constant for Seasonal Estimate (0 < \( \beta < 1 \))
- \( \gamma \) = Smoothing Constant for Trend Estimate (0 < \( \gamma < 1 \))
\[ p = \text{Number of Periods in the Seasonal Cycle}. \]

One of the difficulties in the use of Winter's Exponential Smoothing is determining the optimal values for \( \alpha \), \( \beta \), and \( \gamma \). A recommended method for choosing the optimal smoothing constants is to minimize the Root Mean Squared Error (RMSE). But it is inefficient to calculate the minimum RMSE \( \left( \sqrt{\frac{\sum_{t=1}^{n}(R_t - S_t)^2}{n}} \right) \) by using different combinations of \( \alpha \), \( \beta \), and \( \gamma \) values manually. Fortunately, the analyst can implement this mechanism as a function in a computer program. First, one can compute the smoothed series for the values of \( \alpha \), \( \beta \), and \( \gamma \) of 0.4, 0.3, and 0.1, respectively. In the same way, one can secondly compute the smoothed series for the values of \( \alpha \), \( \beta \), and \( \gamma \) of 0.4, 0.3, and 0.2, respectively. Every time the value of RMSE can be obtained from a combination of \( \alpha \), \( \beta \), and \( \gamma \). Accordingly, the minimum value of RMSE can be found and the optimal value of \( \alpha \), \( \beta \), and \( \gamma \) can thus be adopted confidently.

Through Winter's Exponential Smoothing technique, seasonal estimation and trend estimation can be determined from the calculation of the mathematical model given above. The seasonal estimation is smoothed in Equation 2.5 and the trend estimation is smoothed in Equation 2.6. This method accurately accounts for any seasonal fluctuation or linear trend at the same time from the data series.
2.1.4 Linear Exponential Smoothing

The linear (Holt's) exponential smoothing technique can be employed when the data series contains the trend pattern in a time series plot. This two-parameter exponential smoothing was developed by C. C. Holt and is an extension of simple exponential smoothing. It adds a trend factor to the smoothing equation as a means of adjusting the trend. Two equations and two smoothing constants are adopted in the model:

\[ F_{t+1} = \alpha X_t + (1 - \alpha)(F_t + T_t) \]  \hspace{1cm} (2.7)

\[ T_{t+1} = \beta(F_{t+1} - F_t) + (1 - \beta)T_t \]  \hspace{1cm} (2.8)

where:

- \( F_{t+1} \) = Smoothed Value for Period \( t + 1 \)
- \( X_t \) = Actual Value
- \( T_{t+1} \) = Trend Estimate
- \( \alpha \) = Smoothing Constant for the Data (0 < \( \alpha \) < 1)
- \( \beta \) = Trend Constant.

In Equation 2.7 the first parameter \( \alpha \) is used for the smoothing constant which is the weight of the most recent observation to the older observation. The second parameter, \( \beta \) is used the same way for the trend factor and the trend estimate is calculated in Equation 2.8. This method accurately accounts for any linear trend in the data series but little or no seasonal pattern.
2.1.5 Runs Test Technique

All the techniques of data analysis for time series data are typically based on an assumption of *sample randomness*, that is, on the assumption that the sample data was collected by some randomization procedure (Walpole & Myers 1989). Such an assumption, however, may be tested by the employment of a nonparametric procedure called the Wald-Wolfowitz Runs Test. The runs test, based on the order in which the sample observations are obtained, is a particularly easy technique for testing the null hypothesis $H_0$ that the observations have indeed been drawn at random.

To illustrate the runs test, suppose that the analyst is dealing with a time series of $n$ observations, where $n$ is even and greater then 30. Define the number of runs, $R$, as a subsequence of one or more identical symbols representing a common property of the data. For example, consider a series of data: 5.3, 4.1, 5.3, 5.8, 5.8, 2.9, 4.2, 2.4, -0.3, 2.8, 5.0, 5.2, -0.5, -1.3, 4.9, 4.7, 5.3, 2.5, -0.2, 1.9, -2.5, 3.5, 6.5, 2.3. For the given sample the median value is 3.2. By comparing each data point with the median, the analyst replaces each data point by the symbol “1” if it is bigger than the median (3.2), and by the symbol “0” if it is smaller than the median. The sequence for the data series is obtained:

```
1 1 1 1 1 0 1 0 0 0 1 1 0 1 0 0 0 0 0 1 0
```

where the analyst has grouped subsequences of similar symbols. Such groupings contain 10 runs in this case. The reason for doing this in testing randomness is to find the ordering or positioning of the items in the sequence, not the frequency of items of each
type. Consequently, one should test the null hypothesis when the significance level $\alpha$ is 0.05 for a two-tailed test:

$$H_0: \text{the data series is random}$$

$$H_1: \text{the data series is not random}$$

Then, the following equations can be used to test the null hypothesis for a sample size that is greater than 30.

$$\mu_R = \frac{2n_1 \cdot n_2}{n_1 + n_2} + 1 \quad (2.9)$$

$$\sigma_R = \sqrt{\frac{2n_1 \cdot n_2 (2n_1 \cdot n_2 - n_1 - n_2)}{(n_1 + n_2)^2 (n_1 + n_2 - 1)}} \quad (2.10)$$

$$Z = \frac{R - \mu_R}{\sigma_R} \quad (2.11)$$

where

$R$ is the total number of runs

$\mu_R$ is the mean value of $R$

$\sigma_R$ is the standard deviation of $R$

$n_1$ is the number of observations greater than median

$n_2$ is the number of observations less than median
\( n \) is the sample size (\( n = n_1 + n_2 \))

After the transformation from three formulas mentioned above, the decision rule is:

\[
\text{Reject } H_0 \text{ if } \quad \frac{\frac{R - \frac{n}{2} - 1}{\sqrt{n^2 - 2n}}}{\sqrt{\frac{4(n-1)}{2}}} < -Z_{\alpha} \quad (2.12)
\]

The random variable \( Z \) follows a standard normal distribution. From the symbol series in the preceding paragraph it is clear that the number of runs is 10. Since the sample size is 24 observations, the value of the test statistic is,

\[
\frac{R - \frac{n}{2} - 1}{\sqrt{n^2 - 2n}} = \frac{10 - 12 - 1}{\sqrt{24^2 - 2 \cdot 24}} = \frac{-1.25}{92}
\]

After checking the table of the standard normal distribution, one can see that the \( \alpha \) value corresponding to \( Z_{\alpha} = -1.25 \) is 0.1056. Hence, the null hypothesis of randomness can not be rejected, and thus this series data is random.

2.2 Literature Review of Software Design

2.2.1 Traditional Methodologies

A great many methods have been proposed for resolving traditional programming design, such as the waterfall method, structured analysis and structured design. These
methods helped with many compact software applications in past years, but have become more unworkable today.

Most of the design methods in software applications are based on the traditional waterfall model of the software life cycle from the late 1960's. The waterfall method, as a linear process, often follows a logical progression: requirement specification, analysis, design, testing, integration and maintenance (see Figure 2.2). Initially, the idea is that one phase should be completed before the next one is started. Very often, serious problems can be caused by insufficient feedback and an inability to respond to a change in time (Stroustrup 1991). In addition, this model does not accurately describe how software is actually developed and constantly fails to fit the real world situation today. For example, a user may request a new feature after the design stage and then require a change backward. If the requirements change, a system based on these approaches may require massive restructuring. At this point, the iteration design or spiral model is more practical to programming strategy. Each different phase of a software project should not be strictly separated.

Structured analysis was also introduced as a more manageable and definitive approach to software design. The programmer must understand the problem, build a semantic model of the problem, and finally translate the model into code. Functional decomposition allows this process to operate and always follows a top-down strategy. From the top-down strategy, the first design step is to specify the program's intended function. One must answer the question, "What does the program do?" The programmer
describes the major steps that the program must perform and then defines the description by breaking each major step into smaller ones. It treats a program as a description of a process and breaks it down into sub-processes. This method can help reduce the complexity of a system into a set of less complex subsystems which perform better than procedural programming.

![Waterfall Model of the Software Life Cycle](image)

*Figure 2.2 Waterfall Model of the Software Life Cycle*

Because of the nature of structured analysis, this approach usually separates the program into functions and data types. Functions perform a specific operation and data types are manipulated by functions. Such a module mechanism is often called data abstraction. One potential problem can be found in this data abstraction from past experience; However, once the function or data type has been defined, it is isolated with
the rest of the program. There is no way to adapt it to new uses except by modifying its original definition, which can lead to severe inflexibility and difficult maintenance. The relations among hundreds of distinct functions can be easily confused in a large program. The second potential problem is that data may be lost or revised accidentally if the relations among functions and data types are so ambiguous.

2.2.2 Object-Oriented Approach

Object orientation represents a major shift from traditional methods of software construction. In the 1970s, the Smalltalk project at Xerox Palo Alto Research Center was launched by Alan Kay. It grasped the new concept of classes, as used in Simula-67, and supported a graphical interface. The artificial intelligence research community embraced this new idea into the LISP language. By that time, all these languages were available only within research laboratories. Around 1984, Apple Macintosh released its first commercial object-oriented development environment for its operating system. As a result, the Mac is easier to use and has attracted thousands of users. In 1988, the NeXT machine was introduced by providing further functionality, tools, and object libraries to facilitate the integration of the object-oriented environment. However, while more modern multitask operating systems have prevailed in the 1990s, object-oriented design will continue this trend in software development and design. By taking advantage of this technique, object orientation should enable a designer to model applications closer to the
real world environment. A designer can then simulate and realize solutions to the problems.

Object-oriented system design was not developed to replace structured analysis, but rather to enhance it. Since the 1960s, structured analysis that is based primarily on decomposing functional components of a system has been developed and received great acclaim. This approach is straightforward (top-down or bottom-up), but it forces programmers to focus on functions with little regard for the structure of the data type. The designs often result in more code and less data, because the organization of the data type is a derivative of the processes and their need to interact. Furthermore, when the designing system becomes large and complex, errors are inevitable and maintenance becomes difficult. As a different approach, programmers do not analyze a problem in terms of functions or procedures; nor do programmers describe it in terms of the data type. Instead, they analyze the problem as a system of interactive objects. The questions programmers often ask are, “What are the objects?” “What characteristics does this object possess?” and “What functions does this object accomplish?” By advancing the data abstraction to a higher level, object-oriented design is constructed around objects that are more intelligently formed.

In order to improve the inflexibility problem of the traditional waterfall model, object-oriented design develops an iterative and incremental process which regularly contains the following activities (Meyer 1988):
1. Identifies the classes.

2. Assigns attributes and functions for each class.

3. Finds relationships between the classes.

4. Arranges the classes into hierarchies.

The object-oriented approach strengthens the design process by adding new mechanisms such as object, class, polymorphism, and inheritance concepts so that the system is capable of managing complexity and improving productivity. It modifies the waterfall model to a simplified form which is blurred with each phase. However, the object-oriented approach does not pretend to replace conventional software design. Programmers still need structured analysis to develop software by means of the object-oriented design, which always focuses on how programs should be organized, on how codes can be reused, and on how complexity can be managed.
Chapter 3

System Design

3.1 Assumptions

1. System model design: One of the objectives of this thesis is to develop and present a design model which can be adopted to determine whether a reduction of traffic accidents is caused by the effect of a safety improvement. To create this general working system, the analyst performs a number of principal tasks. The following list identifies these tasks and shows the reader how each works. Figure 3.1 presents the general work flow in the development of a system model.

- Gather the raw data in at least 24 sets over a period of time.
- Preprocess raw data to check some assumptions such as normality, sample randomness, and sample independence.
- Process data, for example by entry into a spreadsheet and draw the graphic chart in a time series form.
- Statistically determine if any trend pattern or seasonal pattern exists in data series.
- Determine if it is necessary to remove seasonality or trend from the data.
Start

Gather the Raw Data Over a Period of Time

Check Randomness and Normality Assumptions

Enter Data into a Spreadsheet and Draw the Time Series Chart

Statistically Determine If any Trend or Seasonality Exist

Apply Cumulative Sum Technique

Interpret Chart to Identify Shift

End

Figure 3.1 System Model Work Flow
• Apply the cumulative sum chart techniques.

• Interpret the chart to find any shifts.

2. The system model process mentioned above is usually iterative. It is impossible to define the final system model the first time. In most situations, it is very unwise to design a system model too comprehensive in the beginning.

3. The dependent variables in this system model are time related factors such as days, months, quarters, and years.

4. The independent variables in this system model are traffic accidents or traffic accident rates. Usually accident data with small counts can be treated on a natural distribution like Poisson or negative binomial distributions. In this thesis, accident data was assumed to be normally distributed. Therefore, the analyst should regularly preprocess normality tests such as the normality plot or the Kolmogorov-Smirnov test before applying any CUSUM control charts.

5. In this thesis, the CUSUM plot is applied for individual measurements (sample size \( n = 1 \)) which is different from the CUSUM chart for subgroup measurements in the quality control field.

6. A long data series (at least 36 data points, more if data series are seasonal) is necessary to make use of Time Series Decomposition or Winter's exponential smoothing techniques.
7. Time series data are typically based on an assumption of sample randomness, i.e.,
the sample data has been collected independently and at random. The runs test is
recommended for the randomization procedure in this project.

8. When an out-of-control signal was detected after applying the CUSUM chart, a
search for the assignable cause should be initiated immediately. It is possible to
detect a false shift, thus caution is throughout the process.

3.2 Determining Trend and Periodicity Existence

After drawing the time series chart, the analyst can visually search for any trend or
seasonal pattern in the chart initially. But this approach is somewhat subjective when
human judgment is involved. What the analyst needs is a reliable inference to
statistically test the hypothesis of a trend or seasonal effect.

3.2.1 Tests for Trend

If time series data are characterized in the form of \( y = a + bx \), a hypothesis testing
for simple linear regression should be applied here to check the significance of
regression. In other words, the analyst needs to know whether or not a linear trend exists
in a data series. The appropriate hypotheses are

\[
H_0: b = 0 \\
H_1: b \neq 0
\]
where $b$ is the slope of the regression line. If $b = 0$, the regression line is horizontal which means no trend in the line. So one can even express the hypothesis in this manner:

\[ H_0: \text{there is no linear relationship between } x \text{ and } y \]
\[ H_1: \text{there is a linear relationship between } x \text{ and } y \]

If $H_0: b = 0$ is true, the statistic

\[ F_0 = \frac{SS_R / 1}{SS_E / (n - 2)} = \frac{MS_R}{MS_E} \]  

(3.1)

follows the $F$ distribution, and one can reject $H_0$ if $F_0 > F_{a,1,n-2}$. The test procedure is arranged in an analysis of variance table such as Table 3.1 (Montgomery 1991) and can be easily carried out from statistical software or MS-EXCEL. Thus, the significance test of linear regression can be applied to statistically determine whether a linear trend exists in a time series plot or not.

**Table 3.1 Analysis of Variance for Testing Significance of Regression**

<table>
<thead>
<tr>
<th>Source of Variation</th>
<th>Sum of Squares $SS_R$</th>
<th>Degrees of Freedom</th>
<th>Mean Square $MS_R$</th>
<th>$F_0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression</td>
<td></td>
<td>1</td>
<td>$MS_R$</td>
<td>$MS_R/MS_E$</td>
</tr>
<tr>
<td>Error</td>
<td>$SS_E$</td>
<td>$n - 2$</td>
<td>$MS_E$</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>$S_{yy}$</td>
<td>$n - 1$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 3.2.2 Tests for Seasonality

There is a rank test proposed by Friedman (1937) and reviewed by Kendall *et al.* (1990) which can be applied to detect seasonal patterns in a time series data. The analyst
can rank the values within each year from 1 to 12 for monthly data and let the years represent $c$ columns and the months $r$ rows. Then each column represent a permutation of the integers 1,2,..., 12. Summing across each row gives the monthly score $M_j$, $j = 1,2,..., 12$. Consequently, under the null hypothesis of no seasonal pattern, the test statistic

$$T = \frac{12 \sum_{j=1}^{r} \left[ M_j - \frac{c(r + 1)}{2} \right]^2}{cr(r + 1)}$$

is approximately distributed as $\chi^2$ with $(r-1)$ degrees of freedom. An example is shown in Table 3.2 when $r$ is 12 and $c$ is 3. The rank test gives

$$T = \frac{12 \cdot 1205}{3 \cdot 12 \cdot 13} = 30.89$$

which is significant, based on 11 degrees of freedom.

**Table 3.2 An Example of Rank Test**

<table>
<thead>
<tr>
<th>Month</th>
<th>1990</th>
<th>90</th>
<th>91</th>
<th>90 Ranks</th>
<th>90 Ranks</th>
<th>91 Ranks</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>3540</td>
<td>3450</td>
<td>2900</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>Feb</td>
<td>3090</td>
<td>3030</td>
<td>2920</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Mar</td>
<td>3660</td>
<td>3560</td>
<td>3410</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>11</td>
</tr>
<tr>
<td>Apr</td>
<td>3720</td>
<td>3700</td>
<td>3220</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>May</td>
<td>3900</td>
<td>3970</td>
<td>3780</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>20</td>
</tr>
<tr>
<td>Jun</td>
<td>3980</td>
<td>4190</td>
<td>4090</td>
<td>7</td>
<td>10</td>
<td>10</td>
<td>27</td>
</tr>
<tr>
<td>Jul</td>
<td>4460</td>
<td>4430</td>
<td>4130</td>
<td>12</td>
<td>11</td>
<td>11</td>
<td>34</td>
</tr>
<tr>
<td>Aug</td>
<td>4250</td>
<td>4540</td>
<td>4230</td>
<td>10</td>
<td>12</td>
<td>12</td>
<td>34</td>
</tr>
<tr>
<td>Sep</td>
<td>4190</td>
<td>4140</td>
<td>3900</td>
<td>9</td>
<td>9</td>
<td>8</td>
<td>26</td>
</tr>
<tr>
<td>Oct</td>
<td>4310</td>
<td>3970</td>
<td>3930</td>
<td>11</td>
<td>8</td>
<td>9</td>
<td>28</td>
</tr>
<tr>
<td>Nov</td>
<td>3770</td>
<td>3700</td>
<td>3600</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>17</td>
</tr>
<tr>
<td>Dec</td>
<td>4030</td>
<td>3620</td>
<td>3330</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>16</td>
</tr>
</tbody>
</table>
3.3 Selecting Appropriate Time Series Method

How can one decide which methods are most appropriate for a particular situation in selecting a time series method? Is it necessary to deseasonalize or detrend the time series data before processing the CUSUM techniques? Before answering these questions, the analyst needs a general knowledge of time series analysis that will help he or she to decide which method to use. In the literature review chapter, many techniques have already been discussed which are considerably helpful. Although there are no easy and fast rules in this regard, there are guidelines to assist in making the determination. For example, if one is preparing monthly data that exhibits considerable seasonal patterns, one would want to use one of the methods that is designed to handle such seasonal fluctuations. If one has a data series that exhibits considerable trend patterns, one would want to use one of the methods that is designed to handle trend fluctuations. Table 3.3 provides a quick reference summary to select an appropriate time series method.

Table 3.3 A Quick Reference to Select Time Series Methods

<table>
<thead>
<tr>
<th>De-seasonalize/ Detrend</th>
<th>Comments</th>
<th>Methods Suggested</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Irregular only</td>
<td>No need to detrend or deseasonalize</td>
<td>None</td>
</tr>
<tr>
<td>2. Irregular and Seasonal</td>
<td>Need to deseasonalize</td>
<td>Time Series Decomposition or Winter's Exponential Smoothing</td>
</tr>
<tr>
<td>3. Irregular and Trend</td>
<td>Need to detrend</td>
<td>Holt's Exponential Smoothing or Simple Regression Method</td>
</tr>
<tr>
<td>4. Irregular, Trend, and Seasonal</td>
<td>Need to detrend or deseasonalize</td>
<td>Winter's Exponential Smoothing</td>
</tr>
</tbody>
</table>
Capturing seasonal or trend patterns in the data series is a safe procedure for the purpose of accuracy, but it is not necessary to process it every time. The analyst would strongly suspect that these patterns may cause the false signal in a CUSUM chart. If one suspects that the seasonal pattern is caused by nature, one can deseasonalize the data series with an applicable time series technique. If one suspects that the trend pattern is caused by some special treatment in the case, one can detrend the data series in order to eliminate this effect. The analysts can be free to determine if they want to deseasonalize or detrend the data series under the certain circumstance. The primary objective here is to exclude all the effects one can catch so that the data series can be more precise for future use. However, the procedure of seasonality removal has two advantages:

1. After stripping away the effects of the seasonal pattern in a time series, one can better see the underlying pattern in the data and process them without any irrelevant effects.

2. The accident data may be dependent or serially correlated because of seasonal pattern or trend pattern. If the dependence of sample data is neglected, inferences about the estimated model may be misleading.

### 3.4 Standardization of Cumulative Sum Chart

The formal procedure in the CUSUM chart technique has been widely adopted for statistical control purposes to determine whether the process is out of control. The
conventional CUSUM chart scheme usually takes the form of either the V-mask or decision interval. These two different approaches usually produce exactly the same results. Because both schemes are based on several parameters such as target value, V-mask angle ($\theta$) or lead distance ($d$), it is tedious to compute all of these parameters. Furthermore, the performance of a traditional CUSUM chart depends heavily on the accurate estimation of these parameters. Often such accuracy is difficult to obtain. Due to these difficulties, it is reasonable to find a better approach to construct the CUSUM chart.

The use of conventional control charts has been questioned by analysts in the last few years. The Shewhart control chart uses past history only in an indirect manner. It ignores any information given by the entire sequence of points. This characteristic makes the Shewhart control chart relatively insensitive to small shifts (1.0 standard deviations or less in the mean level) in the process (Montgomery 1985). On the other hand, a V-mask CUSUM control chart uses past history in its decision making procedure. CUSUM looks for changes in the slope of the plotted data points, rather than the distance of a point from a central value. It detects small changes quicker than the Shewhart control chart. As a rule of thumb, in the area of 0.5 to 1.5 standard deviations from the mean level, the V-mask CUSUM control chart will detect changes twice as quickly. However, there are some drawbacks to the V-mask CUSUM control chart. First, it is slow in detecting large shifts in a data series. Second, the success of a V-mask CUSUM control chart depends heavily on the accurate estimation of its parameters. Consequently, a standardized
Shewhart-CUSUM chart or a tabular CUSUM chart can be applied to improve the performance of both the Shewhart control chart and the V-mask CUSUM control chart.

Furthermore, it is important to understand the fundamental difference between the Shewhart control chart and V-mask CUSUM chart in their basic objectives (DeVor 1992). The Shewhart control chart is usually designed only for the purpose of detecting the occurrence of special causes in the sample mean. On the other hand, V-mask CUSUM control chart is designed for the purpose of monitoring the dynamics of the changes in the individual observations or sample means. This means that the V-mask CUSUM chart is practiced to detect shifts for both special causes and common causes which might produce false shifts. Since the focus of this project is special causes, one can employ a new approach that combines these two charts to take advantage of both the sensitivity of a cumulative statistic and the detection of the occurrence of special causes.

The result is called a combined Shewhart-CUSUM control scheme as first introduced by Lucas (1982). This combined Shewhart-CUSUM control chart will detect not only small shifts (less than 1 standard deviation in the mean), but also large shifts (2 or 3 standard deviations). However, without using a target value or subgroup measurement in a designated routine sampling plan, the analyst only deals with individual observations of the CUSUM plot in this project. This process will significantly simplify the statistic from the combined Shewhart-CUSUM control scheme.

First, one can transform all the individual observations of any random variable, \( X_i \), to a new set of observations of a normal random variable \( Z_i \) with mean zero and variance
one. The following consequence should create a new approach of the CUSUM charts for individual observations, which are also assumed to have a common normal distribution with sample mean $\mu_x$ and sample standard deviation $\sigma_x$. Given sample individual observations, $X_1, X_2, X_3, \ldots, X_i$, one can standardize them by

$$Z_i = \frac{X_i - \mu_x}{\sigma_x} \quad (3.3)$$

Thus, instead of summing the $X_i's$ in the past, the analyst can sum the $Z_i$ values. An instant consequence of the above process is that now the analyst can write the cumulative sums as follows:

$$C_i = \frac{\sum_{i=1}^{i} Z_i}{\sqrt{t}} \quad (3.4)$$

which follows a standard normal distribution. This will allow the analysts to plot the $C_i$ on a control chart with constant control limits of $\pm 3$ (standard deviations) and a centerline with mean zero ($\mu_0$). Instead of controlling original data ($X_i$), one can control the transformed values ($C_i$ or $Z_i$) for a control chart. Hence, a simplified control chart with the same appearance as Shewhart charts can be constructed for the cusums. Figure 3.2 represents a top-down procedure of the computation for this standardized CUSUM chart.
Collect Individual Observations (At Least 24 Sets in Time Order)

Computation for the Sample Mean and Variance

\[
\overline{X} = \frac{\sum_{i=1}^{t} X_i}{t}, \quad S_x = \sqrt{\frac{\sum_{i=1}^{t} X_i^2 - \left(\sum_{i=1}^{t} X_i\right)^2}{t(t-1)}}
\]

Transform \( X_i \) to \( Z_i \)

\[
Z_i = \frac{X_i - \overline{X}}{S_x}
\]

Sum the \( Z_i \)'s

\[
\text{sum}_i = \sum_{i=1}^{t} Z_i
\]

Obtain the Standardized CUSUM

\[
C_i = \frac{\sum_{i=1}^{t} Z_i}{\sqrt{t}}
\]

Plot the \( C_i \) on the Standardized Control Chart

Interpret the Chart, Find the Possible Shifts in the Sums

Figure 3.2 Construction of the Standardized CUSUM Chart
However, this standardized Shewhart-CUSUM control chart is unlike the V-mask CUSUM chart or Shewhart control chart in many ways. The Shewhart control chart is for controlling the mean only, and usually comes alone with the use of R chart for controlling dispersion. Neither charts should be used alone, due to the dynamic changes of the mean and variance of a process variable. It is unreasonable to control the mean alone and ignore the variance. In this new CUSUM scheme, Z values were introduced for controlling both the mean and variation of the observations simultaneously. However, the analyst has the same problem when the V-mask CUSUM chart is used for controlling the mean or variance alone. Therefore, instead of using raw data for the CUSUM chart, the analyst can apply Z values for controlling both of them at once.

3.5 Graphical Interpretation of the CUSUM Charts

How can one determine that a shift has formed after constructing the CUSUM plot? It is rather difficult to detect the presence of a shift in the CUSUM plot without any standard principles. The question is, “What standard or decision rules can be appropriately applied for the control limits to detect the shift?” As a discipline mentioned earlier in chapter two, a trend line is formed by a sequence of cusum data according to the size of the shift in the mean and the length that such a shift has sustained. Consequently, whether a trend line on a CUSUM plot is a signal of a shift depends on the slope of the line and the number of successive points on the line (DeVor 1992). As described, the larger the slope of the line on the line, the more likely it is that
the apparent signal is a true signal. Table 3.4 is a recommended list of decision rules for trend line identification as “threshold” values. For example, if a four-point line shows up in a CUSUM plot with a slope greater than 1.25 standard deviation, one would identify it as a signal with a possible shift. If an eight-point line or greater shows up with a slope greater than 0.25 standard deviation, one would also identify it as a signal with a possible shift. This is the first rule of setting control limits in the new CUSUM chart.

Table 3.4 Trend Line Identification Rules

<table>
<thead>
<tr>
<th>Number of Points on the Line</th>
<th>Shift Size (Slope)</th>
<th>Type I Error, $\alpha = 0.01$</th>
<th>Type I Error, $\alpha = 0.05$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>3</td>
<td>2.49</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1.75</td>
<td>1.41</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>1.25</td>
<td>0.9</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.9</td>
<td>0.58</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.6</td>
<td>0.35</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.4</td>
<td>0.15</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.25</td>
<td>0.04</td>
<td></td>
</tr>
</tbody>
</table>

Type I Error ($\alpha=0.01$) was adapted from DeVor (1992). See Appendix B for more detail of advanced calculation.

Just as in interpreting a standardized Shewhart-CUSUM chart for shift detection, the analyst can set a three sigma distance ($\pm 3\sigma$) from the centerline to locate the control limits which generally apply the probability of type I error ($\alpha = 0.0027$) under the assumption of data normally distributed. If $\alpha = 0.05$ is preferred, the control limits on the chart accordingly can be set in 1.65 sigma distance. This is the second rule of setting control limits for the new CUSUM scheme when the Z values of individual observations are compared against these control limits.

After using two kinds of control limits for the shift detection procedure, one would expect an improved performance to detect the shift faster. The average run length (ARL)
is the average number of sample points that must be plotted before a point indicates an out-of-control condition (Montgomery 1991) and can be applied for the comparison in performance of different CUSUM schemes. As discussed above, a standardized Shewhart-CUSUM chart with extra control limits should result in a smaller ARL. Ten or less runs are desired in this project because of the economical consideration. A simulation was performed later to generate the expected ARL.

Another problem one may encounter is that the size of the slope is difficult to judge visually on a computer display because it is relative to the scales of the plot. Since the CUSUM plot is drawn on the graphic display, it is burdensome to adjust both horizontal scale and vertical scale to appropriately correspond with each other. Furthermore, there is no “correct” choice of scale for any standardized CUSUM plots. For simplification, one should choose the ratio 1:1 for the relative scales between the cusum axis (x-axis) and the sample number axis (y-axis). Hence, the slope of a suspected trend line can be calculated when the interval between two sample numbers is equal to 1 standard deviation, which is also the unit for the cusums. Even though the scale of the CUSUM plots can be handled as mentioned above, it is still not safe to visually determine the shift size in the charts. In order to facilitate the trend identification of standardized CUSUM plots, a better approach through the programming algorithmic technique is needed to solve this dilemma. Figure 3.3 shows a computer program flow chart that can be used for the shift detection procedure.
Figure 3.3 Shift Identification Flow Chart
3.6 More Explanation of Std. Shewhart-CUSUM Chart

Suppose that the analyst has an interest in the traffic safety improvement case. Figure 3.4 shows the 24 standardized cusums of traffic accidents from January 1989 to December 1990. With the standardization of the raw data, the appearance of the std. Shewhart-CUSUM chart is always the same; Shewhart control limits and centerline are always set to 3, -3, and 0, respectively. It can be seen from the plot of Figure 3.4 that the cusums are upward first and then downward and upward again along the centerline. The last cusum constantly ends up at zero. The calculations of the standardized cusums of traffic accidents, following this case, are shown in Table 3.5.

For the purpose of explaining the use of standardized Shewhart-CUSUM chart, suppose that one alters the first four data points by subtracting 2 from each set and then replots the chart. The sample mean and the sample standard deviation, calculated from the revised 24 data points, are different (shown in Table 3.6). A revised standardized Shewhart-CUSUM chart based on the new estimates is shown in Figure 3.5. The standard deviations of the first four data points vary from positive numbers to negative numbers. This variation give us an insight into the std. Shewhart-CUSUM chart.

Now suppose the analyst changes the traffic safety device from the period of January 1991 to December 1991. Consequently, the analyst added twelve months of data points to the data series and revised the chart again. The sample mean and standard deviation, recalculated from the 36 data points, are also altered. The standardized Shewhart-CUSUM chart and the corresponding computation based on the new estimates are shown
in Figure 3.6 and Table 3.7, respectively. It is interesting to observe the obvious nonrandom trend formed by the last 12 points that reflect the hypothetical shift in the chart. This illustration demonstrates the general capability of CUSUM charts to quickly identify shifts in the mean. Another advantage is the ability to help determine the time of the shift more precisely by locating the starting point of a trend on the chart. Therefore, one can infer that the traffic safety improvement truly caused a shift in the chart.

Since the interpretation of a CUSUM chart is primarily a search for possible shifts in the data series and is not so much concerned with the actual level of the data on the chart, one may even use a plot of the CUSUM chart without control limits. A CUSUM plot of individual observations can reveal possible changes in the chart by identifying any trend lines. As such, the charting of CUSUM can be applied to the example of traffic safety improvement.

*Figure 3.4* Standardized Shewhart-CUSUM Chart of Accidents by Months
Table 3.5 Standardized Cusums of Accidents by Months

<table>
<thead>
<tr>
<th>Month</th>
<th>Data $X_i$</th>
<th>Normalized Data $Z_i = \frac{X_i - \bar{X}}{S_x}$</th>
<th>Sum $\sum_{i=1}^{t} Z_i$</th>
<th>CUSUM $C_i = \frac{\sum_{i=1}^{t} Z_i}{\sqrt{i}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan, 89</td>
<td>33</td>
<td>-0.974</td>
<td>-0.974</td>
<td>-0.974</td>
</tr>
<tr>
<td>Feb</td>
<td>34</td>
<td>-0.497</td>
<td>-1.472</td>
<td>-1.041</td>
</tr>
<tr>
<td>Mar</td>
<td>35</td>
<td>-0.020</td>
<td>-1.492</td>
<td>-0.861</td>
</tr>
<tr>
<td>Apr</td>
<td>33</td>
<td>-0.974</td>
<td>-2.466</td>
<td>-1.233</td>
</tr>
<tr>
<td>May</td>
<td>34</td>
<td>-0.497</td>
<td>-2.963</td>
<td>-1.325</td>
</tr>
<tr>
<td>Jun</td>
<td>38</td>
<td>1.412</td>
<td>-1.551</td>
<td>-0.633</td>
</tr>
<tr>
<td>Jul</td>
<td>39</td>
<td>1.889</td>
<td>0.338</td>
<td>0.128</td>
</tr>
<tr>
<td>Aug</td>
<td>34</td>
<td>-0.497</td>
<td>-0.159</td>
<td>-0.056</td>
</tr>
<tr>
<td>Sep</td>
<td>36</td>
<td>0.457</td>
<td>0.298</td>
<td>0.099</td>
</tr>
<tr>
<td>Oct</td>
<td>33</td>
<td>-0.974</td>
<td>-0.676</td>
<td>-0.214</td>
</tr>
<tr>
<td>Nov</td>
<td>32</td>
<td>-1.452</td>
<td>-2.128</td>
<td>-0.642</td>
</tr>
<tr>
<td>Dec</td>
<td>36</td>
<td>0.457</td>
<td>-1.671</td>
<td>-0.482</td>
</tr>
<tr>
<td>Jan, 90</td>
<td>34</td>
<td>-0.497</td>
<td>-2.168</td>
<td>-0.601</td>
</tr>
<tr>
<td>Feb</td>
<td>30</td>
<td>-2.406</td>
<td>-4.574</td>
<td>-1.222</td>
</tr>
<tr>
<td>Mar</td>
<td>35</td>
<td>-0.020</td>
<td>-4.594</td>
<td>-1.186</td>
</tr>
<tr>
<td>Apr</td>
<td>37</td>
<td>0.935</td>
<td>-3.659</td>
<td>-0.915</td>
</tr>
<tr>
<td>May</td>
<td>35</td>
<td>-0.020</td>
<td>-3.679</td>
<td>-0.892</td>
</tr>
<tr>
<td>Jun</td>
<td>38</td>
<td>1.412</td>
<td>-2.267</td>
<td>-0.534</td>
</tr>
<tr>
<td>Jul</td>
<td>37</td>
<td>0.935</td>
<td>-1.332</td>
<td>-0.306</td>
</tr>
<tr>
<td>Aug</td>
<td>34</td>
<td>-0.497</td>
<td>-1.830</td>
<td>-0.409</td>
</tr>
<tr>
<td>Sep</td>
<td>36</td>
<td>0.457</td>
<td>-1.372</td>
<td>-0.299</td>
</tr>
<tr>
<td>Oct</td>
<td>35</td>
<td>-0.020</td>
<td>-1.392</td>
<td>-0.297</td>
</tr>
<tr>
<td>Nov</td>
<td>37</td>
<td>0.935</td>
<td>-0.457</td>
<td>-0.095</td>
</tr>
<tr>
<td>Dec</td>
<td>36</td>
<td>0.457</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>
Figure 3.5 Revised Standardized Shewhart-CUSUM Chart

Figure 3.6 Standardized Shewhart-CUSUM Chart with a Hypothetical Shift
Table 3.6 Revised Standardized Cusums of Accidents

<table>
<thead>
<tr>
<th>Month</th>
<th>Data</th>
<th>Normalized Data $Z_i = \frac{X_i - \bar{X}}{S_x}$</th>
<th>Sum $\sum_{i=1}^{t} Z_i$</th>
<th>CUSUM $C_i = \frac{\sum_{i=1}^{t} Z_i}{\sqrt{t}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan, 89</td>
<td>34</td>
<td>-0.974</td>
<td>-0.974</td>
<td>-0.974</td>
</tr>
<tr>
<td>Feb</td>
<td>30</td>
<td>-2.406</td>
<td>-4.574</td>
<td>-1.222</td>
</tr>
<tr>
<td>Mar</td>
<td>35</td>
<td>-0.020</td>
<td>-4.594</td>
<td>-1.186</td>
</tr>
<tr>
<td>Apr</td>
<td>37</td>
<td>0.935</td>
<td>-3.659</td>
<td>-0.915</td>
</tr>
<tr>
<td>May</td>
<td>35</td>
<td>-0.020</td>
<td>-3.679</td>
<td>-0.892</td>
</tr>
<tr>
<td>Jun</td>
<td>38</td>
<td>1.412</td>
<td>-2.267</td>
<td>-0.534</td>
</tr>
<tr>
<td>Jul</td>
<td>37</td>
<td>0.935</td>
<td>-1.332</td>
<td>-0.306</td>
</tr>
<tr>
<td>Aug</td>
<td>34</td>
<td>-0.497</td>
<td>-1.830</td>
<td>-0.409</td>
</tr>
<tr>
<td>Sep</td>
<td>36</td>
<td>0.457</td>
<td>-1.372</td>
<td>-0.299</td>
</tr>
<tr>
<td>Oct</td>
<td>35</td>
<td>-0.020</td>
<td>-1.392</td>
<td>-0.297</td>
</tr>
<tr>
<td>Nov</td>
<td>37</td>
<td>0.935</td>
<td>-0.457</td>
<td>-0.095</td>
</tr>
<tr>
<td>Dec</td>
<td>36</td>
<td>0.457</td>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>
### Table 3.7 12 Points Data Added to the Standardized CUSUM Chart

<table>
<thead>
<tr>
<th>Month</th>
<th>Data $X_i$</th>
<th>Normalized Data $Z_i = \frac{X_i - \bar{X}}{S}$</th>
<th>Sum $\sum_{i=1}^{t} Z_i$</th>
<th>CUSUM $C_i = \frac{\sum_{i=1}^{t} Z_i}{\sqrt{t}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan, 89</td>
<td>35</td>
<td>0.51</td>
<td>0.51</td>
<td>0.51</td>
</tr>
<tr>
<td>Feb</td>
<td>36</td>
<td>0.74</td>
<td>1.25</td>
<td>0.88</td>
</tr>
<tr>
<td>Mar</td>
<td>37</td>
<td>0.97</td>
<td>2.22</td>
<td>1.28</td>
</tr>
<tr>
<td>Apr</td>
<td>35</td>
<td>0.51</td>
<td>2.73</td>
<td>1.36</td>
</tr>
<tr>
<td>May</td>
<td>34</td>
<td>0.28</td>
<td>3.01</td>
<td>1.35</td>
</tr>
<tr>
<td>Jun</td>
<td>38</td>
<td>1.20</td>
<td>4.21</td>
<td>1.72</td>
</tr>
<tr>
<td>Jul</td>
<td>39</td>
<td>1.43</td>
<td>5.64</td>
<td>2.13</td>
</tr>
<tr>
<td>Aug</td>
<td>34</td>
<td>0.28</td>
<td>5.92</td>
<td>2.09</td>
</tr>
<tr>
<td>Sep</td>
<td>36</td>
<td>0.74</td>
<td>6.66</td>
<td>2.22</td>
</tr>
<tr>
<td>Oct</td>
<td>33</td>
<td>0.05</td>
<td>6.71</td>
<td>2.12</td>
</tr>
<tr>
<td>Nov</td>
<td>32</td>
<td>-0.18</td>
<td>6.53</td>
<td>1.97</td>
</tr>
<tr>
<td>Dec</td>
<td>36</td>
<td>0.74</td>
<td>7.27</td>
<td>2.10</td>
</tr>
<tr>
<td>Jan, 90</td>
<td>34</td>
<td>0.28</td>
<td>7.55</td>
<td>2.09</td>
</tr>
<tr>
<td>Feb</td>
<td>30</td>
<td>-0.64</td>
<td>6.91</td>
<td>1.85</td>
</tr>
<tr>
<td>Mar</td>
<td>35</td>
<td>0.51</td>
<td>7.42</td>
<td>1.92</td>
</tr>
<tr>
<td>Apr</td>
<td>37</td>
<td>0.97</td>
<td>8.39</td>
<td>2.10</td>
</tr>
<tr>
<td>May</td>
<td>35</td>
<td>0.51</td>
<td>8.90</td>
<td>2.16</td>
</tr>
<tr>
<td>Jun</td>
<td>38</td>
<td>1.20</td>
<td>10.10</td>
<td>2.38</td>
</tr>
<tr>
<td>Jul</td>
<td>37</td>
<td>0.97</td>
<td>11.07</td>
<td>2.54</td>
</tr>
<tr>
<td>Aug</td>
<td>34</td>
<td>0.28</td>
<td>11.35</td>
<td>2.54</td>
</tr>
<tr>
<td>Sep</td>
<td>36</td>
<td>0.74</td>
<td>12.09</td>
<td>2.64</td>
</tr>
<tr>
<td>Oct</td>
<td>35</td>
<td>0.51</td>
<td>12.60</td>
<td>2.69</td>
</tr>
<tr>
<td>Nov</td>
<td>37</td>
<td>0.97</td>
<td>13.57</td>
<td>2.83</td>
</tr>
<tr>
<td>Dec</td>
<td>36</td>
<td>0.74</td>
<td>14.31</td>
<td>2.92</td>
</tr>
<tr>
<td>Jan, 91</td>
<td>30</td>
<td>-0.64</td>
<td>13.67</td>
<td>2.73</td>
</tr>
<tr>
<td>Feb</td>
<td>32</td>
<td>-0.18</td>
<td>13.49</td>
<td>2.65</td>
</tr>
<tr>
<td>Mar</td>
<td>31</td>
<td>-0.41</td>
<td>13.08</td>
<td>2.52</td>
</tr>
<tr>
<td>Apr</td>
<td>30</td>
<td>-0.64</td>
<td>12.45</td>
<td>2.35</td>
</tr>
<tr>
<td>May</td>
<td>28</td>
<td>-1.10</td>
<td>11.35</td>
<td>2.11</td>
</tr>
<tr>
<td>Jun</td>
<td>27</td>
<td>-1.33</td>
<td>10.02</td>
<td>1.83</td>
</tr>
<tr>
<td>Jul</td>
<td>28</td>
<td>-1.10</td>
<td>8.93</td>
<td>1.60</td>
</tr>
<tr>
<td>Aug</td>
<td>27</td>
<td>-1.33</td>
<td>7.60</td>
<td>1.34</td>
</tr>
<tr>
<td>Sep</td>
<td>24</td>
<td>-2.01</td>
<td>5.59</td>
<td>0.97</td>
</tr>
<tr>
<td>Oct</td>
<td>26</td>
<td>-1.56</td>
<td>4.03</td>
<td>0.69</td>
</tr>
<tr>
<td>Nov</td>
<td>24</td>
<td>-2.01</td>
<td>2.01</td>
<td>0.34</td>
</tr>
<tr>
<td>Dec</td>
<td>24</td>
<td>-2.01</td>
<td>0</td>
<td>0.00</td>
</tr>
</tbody>
</table>
3.7 Tabular CUSUM Chart

A CUSUM procedure for controlling the mean can be implemented in tabular form. The procedure proposed by Lucas (1982) is easier to carry out if a spreadsheet is applied and will be examined below. Due to the introduction of the combined Shewhart-CUSUM concept, the analyst also transforms the magnitude of individual observations to the standardize Z values for use. This tabular CUSUM chart can not only detect the shift by using the decision interval H value, but also by using the individual Z values for an out-of-control signal. Compared to the standardized Shewhart-CUSUM chart, one can expect a competitive performance of fast shift detection. In general, a list of Z values of individual observations would indicate whether the corresponding data points are inside or outside the control limits (Ryan 1989). Therefore, the analyst may wish to design a tabular CUSUM scheme with a very small K value (like 0.1 or 0.2), proper H value, and suitable Z value to obtain a particular ARL (less than 20 expected). The following list identifies these tasks for a typical tabular CUSUM procedure:

1. Initialization for K, H, and \( \alpha \) values.

   - K (Reference Value) is one-half of the magnitude of the mean shift that one wishes to detect (Woodall 1986). Typically, one mean shift (in Z units) is selected for the detection of the desired shift. If a smaller mean shift is preferred, one can even choose 0.1 or 0.2 of the K value. The K value shares the same scale as the Z value.
• H is the decision interval for an out-of-control signal. Normally, H is chosen according to the K value. It is set to either 4 or 5 in most cases. Nomographs were given by Goel and Wu (1971) for CUSUM charts to find the values of H and K that approximately yield the desired ARL. In recent advances, an optimal CUSUM scheme is recommended by Gan (1991) to determine the optimal value of H and K after selecting the smallest acceptable in-control ARL.

• \( \alpha \) is the probability of committing a type I error or false alarm. For example, if \( \alpha = 0.05 \) is applied, the control limits for the Z value are \( \pm 1.65 \). If \( \alpha = 0.01 \) is preferred, the control limits for the Z value can be set to \( \pm 2.33 \).

2. Calculate the sample mean and sample standard deviation to create a normal distribution \( N(\mu, \sigma) \).

3. Calculate Z values \( (Z = \frac{X - \mu}{\sigma}) \) for all individual observations.

4. Calculate \( S_{H_i} \) values \( (S_{H_i} = \max\left[0, (z_i - k) + S_{H_{i-1}}\right]) \) for positive shift detection.

5. Calculate \( S_{L_i} \) values \( (S_{L_i} = \max\left[0, (-z_i - k) + S_{L_{i-1}}\right]) \) for negative shift detection.

6. Compare H value with \( S_{H_i} \) and \( S_{L_i} \) to detect the shift for an out of control signal.
7. By setting the starting values to H/2, the fast initial response (FIR) can be applied to improve the sensitivity of a tabular CUSUM chart devised by Lucas and Crosier (1982).

3.8 System Requirements

3.8.1 Hardware Requirements

- A PC with a 486DX or higher processor running MS-Windows 95.

- Four megabytes of RAM and a hard disk with 10 MB of available disk space.

- A graphics display (15 inches) and Super VGA or higher resolution video adapter capable of display 256 colors or greater.

- A 1.44 megabytes, 3.5' floppy disk drive and a Windows-compatible mouse.

3.8.2 Software Requirements

Because a Traffic Safety Improvement Evaluation (TSIE) software package will be developed in this thesis, the following basic software requirements are needed:

- MS-Windows 95 running in a 32-bit protected-mode is required for the operating system.
• Table 3.8 below shows all the files necessary for the TSIE application to run properly. The user should ensure that all files exist on the user’s computer and they are the proper version.

Table 3.8 Necessary Files to Run TSIE Program

<table>
<thead>
<tr>
<th>No.</th>
<th>File Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TSIE.EXE</td>
<td>Main Program (Executable File)</td>
</tr>
<tr>
<td>2</td>
<td>VCFI32.OCX</td>
<td>Spreadsheet Object</td>
</tr>
<tr>
<td>3</td>
<td>VCFI32.OCX</td>
<td>Chart Object</td>
</tr>
<tr>
<td>4</td>
<td>VCFIDL32.DLL</td>
<td>Chart Function DLL</td>
</tr>
<tr>
<td>5</td>
<td>VCFIWZ32.DLL</td>
<td>Chart Wizard DLL</td>
</tr>
<tr>
<td>6</td>
<td>OLEPRO32.DLL</td>
<td>Windows 95 OLE 2 DLL</td>
</tr>
<tr>
<td>7</td>
<td>MFCANS32.DLL</td>
<td>Windows 95 Application DLL</td>
</tr>
<tr>
<td>8</td>
<td>MSVCRT20.DLL</td>
<td>Windows 95 Application DLL</td>
</tr>
<tr>
<td>9</td>
<td>KERNEL32.DLL</td>
<td>Windows 95 32-bit Kernel</td>
</tr>
<tr>
<td>10</td>
<td>USER32.DLL</td>
<td>Windows 95 32-bit Users</td>
</tr>
<tr>
<td>11</td>
<td>GDI32.DLL</td>
<td>Windows 95 32-bit GDI</td>
</tr>
<tr>
<td>12</td>
<td>COMDLG32.DLL</td>
<td>Windows 95 32-bit Common Dialog Library</td>
</tr>
<tr>
<td>13</td>
<td>OLEAUT32.DLL</td>
<td>Windows 95 OLE 2 32-bit Automation</td>
</tr>
<tr>
<td>14</td>
<td>OC30.DLL</td>
<td>Windows 95 OLE 2 DLL</td>
</tr>
</tbody>
</table>

• Item 1 file is the major part of the Traffic Safety Improvement Evaluation Program. Users should create a new sub-directory in which to put this executable file.

• Items 2, 3, 4, and 5 may be in the Windows System directory (c:\windows\system) on the computer’s path or in the sub-directory which the user specifies. During the installation process, these items are registered in the Windows 95 registration database. If the user moves these items to a different directory or renames the directory that contains it, the information in the registry is no longer valid. This may cause the TSIE program to be unable
to run. To solve this problem, the user must use the REGSVR32.EXE utility to re-register these items in a new location.

- Items 6, 7, 8, 9, 10, 11, 12, and 13 are part of Microsoft Windows 95’s redistributable files necessary for the TSIE application to operate properly. The user’s software environment should have these files installed and registered appropriately.

- If modifications are necessary for the TSIE program, one should have MS-Visual C++ (version 4.0 or later) installed to make the modifications to the software and recompile it.

3.8.3 Monitor Issues

Be aware that different monitor modes can cause problems. VGA monitors (640 * 480 pixels) not only display information at lower resolution, but they also fit less information on the screen than higher resolution devices such as SVGA (800 * 600 pixels). A common mistake among users is to build the TSIE application on an SVGA monitor, only to find that the application does not display information correctly on a VGA monitor. For example, a spreadsheet that displays 40 rows and 8 columns on an SVGA monitor might display only 30 rows and 6 columns on a VGA monitor. Testing on different monitors during installation will help detect these problems before running the application.


Chapter 4

Object-Oriented Design

4.1 Key Concepts

4.1.1 Abstraction

"Abstraction" is the process of ignoring details that are not relevant to the current purpose in order to concentrate on essential characteristics. For example, this complex world is full of things that one can see. In forming such abstractions, one chooses to ignore most of the details and focus on a subject such as a person, dog, or table that concerns him (Shlaer & Mellor 1988). One knows these items possess additional details; but one simply chooses not to use them. As another example, in developing a student registration system, one can think about Paul Steven and abstract him as a student, thinking only about his major and his grades. When working on a college baseball team application, one can abstract Paul as a player and be concerned with only his position and batting average: one person, two completely different abstractions. There is a lot of information about Paul Steven but one just focuses on the interesting items and ignores all others. This technique is an important method in managing complexity in software design.
The abstraction technique has been commonly used in traditional programming. Data abstraction or function abstraction are very typical examples of this kind. Since restraints and errors have occurred often in the past, it is reasonable to switch to a higher level of abstraction. An intelligent structure which can cope with automation is needed. With support for combined data abstraction and function abstraction, construction of an object provides a much more powerful abstraction mechanism to fulfill this goal.

4.1.2 Encapsulation

"Encapsulation" (or information hiding) is the process of hiding the data members and function members that process the data into a single entity, called a class. It is common practice to divide a large program into several classes, each of which has a clearly defined interface of data and functions that the other classes can reuse. Thus, those aspects of classes that need not be visible to other objects are hidden and allow the user to use an object without understanding how the class for that object is implemented. The aim of encapsulation is to make each class as independent of the others as possible. Hiding the data in a class can then be secured without accidental change by other classes. This is well illustrated in Figure 4.1.

The first example from daily life is fairly easy to understand. One does not know how a television works on the inside, but that does not stop him from watching TLC or the Discovery channel. The user knows how to use the channel selector and other front-panel controls. To an object, the user needs to build encapsulation into objects so that he
can use them without knowing how they work internally. In other words, the user can insert many function members and data members into a class that can interact correctly.

For the student registration system example, a Student class can encapsulate both the data members, such as name, major, and address information, and function members such as calculating the correct grades. In the baseball application, the Player class could encapsulate both the player data members, such as name and position information, and the player function members, such as calculating batting statistics. Thus, other parts of the application could request the batting average without needing to know how those values were calculated.

![Figure 4.1 Hiding Data Members and Function Members in the Classes](image-url)
4.1.3 Classes and Objects

Classes are the fundamental organizing units of object-oriented programming. A class is a structure that contains some data members and some functions to operate on that data. A class may include public, private, and protected parts to specify the visibility of the members based on the encapsulation (information hiding) concept. Once the user has defined a class, he can do the following tasks:

- **Derive a new class from it.** When deriving a new child class from the parent class, the user not only can add new data or function members to the child class but also can override inherited function members. This mechanism lets the user add to or change the functionality of the parent class giving the user a chance to expand the program.

- **Add its data and function members to that of another class.** As with inheritance derivation, the user can add new data or function members and override inherited ones to change the behavior of the new class. This mechanism saves the user much time in rewriting the function or data that he already has.

- **Create an instance of the class.** The user can declare one or more instances of that type. These declared instances are so called “objects.” When one talks about designing objects, one is talking about designing classes. Classes help programmer model the way real-life things behave. Software designers are trying to build classes that can be mapped to the problem domain. By regarding classes as objects, a more flexible system is then obtained.
4.1.4 Polymorphism

The word polymorphism originates from the Greek meaning “many forms” or “many types.” Polymorphism is the ability to call appropriate member functions for an object without specifying the object’s exact type. The purpose is to reuse the existing object to develop a new object and keep the size of the program small. For the student registration system example, in Figure 4.2 there is a student class hierarchy that contains a virtual GradeCalculation function. This virtual GradeCalculation function in Student Class is actually a placeholder. On the contrary, GradeCalculation functions in both MechanicalStudent Class and ChemicalStudent Class are implemented in different calculations due to different majors. In C++ language, the statement student->GradeCalculation() will call the appropriate function automatically, without requiring the programmer to examine the type of object to which student object points.

4.1.5 Inheritance

Inheritance is a relationship among classes where a child class can share the data member and function member of a parent class and adapt it for its own use, i.e., inheritance provides a mechanism to create new classes as modifications to existing classes. Through inheritance, classes can be derived from other classes. A derived class can inherit all the functions and data members of its parent. The main advantage of this approach is that existing classes can be reused to a great extent. For the student registration system example, the MechanicalStudent Class and ChemicalStudent Class
needs to track basic information about a student, such as his name and address. Instead of repeating the data and function members in both derived classes, the programmer can put them into a parent class, such as a Student class. The MechanicalStudent Class and ChemicalStudent Class can then be child classes of the Student class and inherit all of the data members and function members from that class (see Figure 4.3).

Figure 4.2 Example of Usage of Polymorphism Mechanism

Figure 4.3 Example of Usage of Inheritance Mechanism
4.1.6 Event-Driven with Messages

Traditional programming is usually written as a sequence of steps grouped into functions or procedures. Procedures issue requests (stimuli) for external input and then wait for it. When input arrives, control resumes within the procedure that made the call. The flow of control is determined by the ordering of procedures and control mechanisms, such as if, switch, and while. This implies that the programmer arbitrarily decides how to structure the entire program in an unrealistic order during program creation. Programs are assumed to capture the essence of the real world. Unfortunately, the world is changing and the ideas and models of the world frequently fail to match. What we design is an inflexible system.

Object-oriented design, however, is organized around events or messages. An event occurs when something within a program changes. Messages are used to signal that an event has taken place or is about to occur. For example, MS-Window 95 posts a message called WM_LBUTTONDOWN each time the left mouse button is pressed. A program can be written to watch for these messages and respond to them appropriately. By setting events within a class, one defines relationships between classes which are associated through data members or function members. This relationship is the abstraction of a set of associations that systematically hold messages between different classes. When sending a message, the programmer specifies the communication protocol among the classes of a program. Moreover, messages are not only used among classes, but also
operated among different functions within a class. Figure 4.4 illustrates all the relationships among these classes.

![Diagram of class hierarchy and message loop]

**Figure 4.4** Communication Between Classes Through Messages or Events

### 4.1.7 Class Hierarchies

Class hierarchy is very useful in presenting a finished system to help others understand or maintain programs. After a long period of time in developing the basic concept of objects, classes, and inheritance, the programmer still does not have any rules or guidelines available to help integrate all these ideas into a complete design. What is needed is a better hierarchical structure that can be adopted for higher productivity and easy maintenance. Unfortunately, drawing hierarchy by hand or using a standard drawing
package is tedious. Furthermore, a software program that contains thousands of functions may cause problems if one only builds a function-driven or data-driven hierarchy. A new approach that applies class hierarchy is more useful for the high level object-oriented structuring of an application. A well-designed class hierarchy should have the following abilities:

- The ability to choose the appropriate representation to work at any stage of importance. This is similar to the situation of reading a book. One can read a book by scanning chapters, by checking the index or by reading word for word.

- It is essential that the hierarchy fits a program that is integrated with the program text (source code). For example, a program can be represented in three different views: as source codes in text form, as functions in list form, or as classes in list form.

- The separate views for each representation should be available and the programmer should be free to move or edit them.

Having constructed a hierarchy for the purposes of designing and implementing a program, it can be used dynamically to view and navigate through a developing application, forming the basis of a browsing tool, or an “active” documentation. Primary criteria is the need to allow multiple views of a program with shared information and consistency under change.
4.2 Designing an Object-Oriented System

4.2.1 Identify Candidate Object Classes

The first step in an object oriented-design project is to define the classes that the program needs. The designer attempts to identify the key abstractions from the problem domain. Each real-world entity is mapped onto a software project. One technique for identifying classes is to write a description of the program's purpose, list all the nouns that appear in the description, and choose needed classes from that list (Booch 1991). This is a simplistic approach whose success depends on how well the original description is written. For a banking application, typical objects would be Account, Bank, Loan, Credit and Customer. For a student registration system application, typical objects would be Student, Course, Time, Classroom, Lecturer, and so forth.

Identification of the classes requires a considerable knowledge of the problem domain. The designer must study the problem requirements while learning the terminology and fundamentals of the problem domain. However, in this thesis, a Spreadsheet class is a likely candidate class for easy data entry and calculation. A Chart class is the obvious next candidate for a graphic drawing object. The programmer can now pick up the potential classes to support in this project:

- **SpreadView Class (Spreadsheet Class)** can display information in a series of rows and columns, including ones that display row and column headings. In addition, a spreadsheet is excellent for the calculation environment.
• **ChartView Class (Chart Class)** can be used to design graphs interactively and display better presentations.

• **Multiple Document Interface (MDI) Class** allows users to display multiple documents simultaneously, with each document displayed in its own window.

• **Window Class** provides the base functionality of all window classes such as dialog boxes, controls, control bars, child windows, and so forth.

• **Menu Class** allows a convenient and consistent way to group commands and provides easy access to functions.

• **Dialog Boxes Class** is used to display information to the user or ask the user for some data entry.

• Some other controls such as **Frame, Command Button, Check Box, Combo Box**, and **Scroll Bar** are useful classes for users to more easily interact with a graphic user interface (GUI).

• **Print Class** provides a device-independent drawing capability to send text and graphics to the printer.

• **Document Class** provides basic functionality such as creating a document, loading a file, and saving a file.

• **Tool Bar Class** provides a row of bitmap buttons that can act like push buttons to speed up the usage of the application.
• **CUSUM Class** provides the functions for CUSUM techniques.

• **Runs Test Class, Time Series Decomposition Class, Winter’s Exponential Smoothing Class, Holt’s Exponential Smoothing Class** and so on.

### 4.2.2 Assign Attributes and Functions

Once the programmer has identified a class, the next task is to determine what responsibilities it has. Each class has “attributes,” which are the properties or characteristics that describe it and its functions, which are used to take specific action. For example, in the TSIE program a **SpreadView** class could have *Title, Visible* and *Width* attributes, as shown in Table 4.1. The programmer can change the values of these attributes in design time.

Each class also has “functions,” which are how their states change during those interactions and how an object interacts with other objects. For instance, a **ShowGridLines** function will set the visibility state of the **SpreadView**’s grid. A **ShowHScrollBar** will set the visibility state of the **SpreadView**’s horizontal scrollbar. However, the act of assigning attributes and functions (or behaviors) gives the programmer a much clearer idea of what constitutes a useful class. Each class’s interface is determined through the encapsulation mechanism. The functions provided by each class and the functions used by each class are predetermined. This process is also likely to be iterative in deciding what a class should know and what it can do.
4.2.3 Find Relationships Among the Classes

This step is an extension of the previous one. After a child class is derived from the parent class, the relationships among classes are determined automatically through the concept of inheritance. For the TSIE program example showing part of the hierarchies in Figure 4.5, if \texttt{CView} Class contains five public function members, then \texttt{CSpreadView} Class and \texttt{CChartView} Class both can have these five functions. If \texttt{CView} Class contains two virtual functions then \texttt{CSpreadView} Class and \texttt{CChartView} Class both should have their own version of virtual functions under the same name. That is, often one class depends upon another class because it cannot be used unless the other class exists. This is necessary when one class calls the function members of the other class.
4.2.4 Arrange the Classes into Hierarchies

Arranging the classes into hierarchies is the last step that uses all the information gained from the previous three steps. When adding new classes into the hierarchies, one should keep in mind all the concepts mentioned above, including which parent class the programmer wants to derive from, what functions and attributes the new class can inherit, and what new functions and attributes the new class can create.

This is a complicated but powerful process when the programmer handles an object orientation environment through the class hierarchy mechanism. By using Visual C++ with Microsoft Foundation Classes (MFC) to accomplish the tasks for the TSIE program, the software designer works with more than 150 pre-built classes. Most information is hidden through the encapsulation mechanism and abstraction concept. The programmer controls the whole class structure via a tree diagram. The programmer can also jump to any stage within a program to make any changes, if necessary. Figure 4.6 shows new classes derived directly from the MFC class library in the TSIE project. One can switch easily from class to class, view to view, and function to function in the
class hierarchy. Figure 4.7 shows a complete class hierarchy of the TSIE program in a flow chart view. While the white box is part of the MFC application framework, the shadow box is the main part of the TSIE program.

Figure 4.6 Partial Class Hierarchy in a Tree-View Structure
4.3 Software Design Alternatives

Most of the software design methods used in the past were based on a function-driven or data-driven decomposition of the system. These approaches differ in many ways from the approach taken by object-oriented design where data and function are highly integrated. Generally speaking, most object-oriented methods were developed in academia by different individuals. The following are the typical models:

- Object-Oriented Analysis (OOA) by Coad and Yourdon (1991).

Moreover, since programmers are pragmatic and more interested in getting the job done efficiently and effectively than worrying about object-orientation, rapid application development (RAD) tools have become another favorite approach. This is why Visual Basic and PowerBuilder came into play in this field, and have been added to the list in the software design arena as additional alternatives. Instead of creating basic components (new classes), RAD tools like PowerBuilder and Visual Basic are more concerned with getting applications built quickly with existing components. All the required components are already pre-built from the package or from commercial supplies. These tools can save programmers time while objects can be reused repeatedly.

However, as graphical user interface environments continue to grow in popularity, more software developments are heavily based on object-oriented design techniques.
Selecting an appropriate software language to take full advantage of object-oriented design is not an easy task. It would be prudent to spend time evaluating several languages that are associated with object-oriented design. Some languages like C++ or Object Pascal have already become popular for public use while others like Ada 95, Modula-3 and Eiffel are still in the research state. A summary of some of these languages is given below.

1. **C++ Language:**

   - C++ is a hybrid language that was designed to cope with a complete object-oriented concept from its ancestor, the C language.

   - C++ is compatible with the C language which has a tremendous library, such as standard mathematics library and run time library. It is easier for those C language developers to upgrade to C++.

   - C++ programs not only run faster but also contain a smaller size of executable files. It is also a reliable and maintainable language.

   - However, C++ is difficult to learn and use.

2. **Smalltalk Language:**

   - In its third decade of existence, Smalltalk distinguishes itself from other object-oriented languages with its complete object-oriented organization and
environment. It supports everything in an object-oriented language, including inheritance, polymorphism, and data encapsulation.

- To a novice, **Smalltalk** is the best tool in which to learn the object-oriented technique.

- It does not perform well in execution speed and is not widespread in use.

3. **Object Pascal Language:**

- Performance in **Object Pascal** is excellent although it fails to support the data binding aspect of encapsulation.

- It is easier to learn the **Object Pascal** language than the **C++** language.

- **Object Pascal** is gaining widespread popularity (Borland Delphi).

4. **Visual Basic Language:**

- It is certainly not an object-oriented language like **C++** or **Smalltalk**, but **Visual Basic** employs a wide range of object-oriented concepts.

- **Visual Basic** fully exploits the power of encapsulation which allows users to use an object without knowing how it does its work; all users have to do is to set properties and call methods from each control (**VBX** or **OCX**).

- In **Visual Basic**, there is no class inheritance. Without inheritance, there is no polymorphism.
5.1 Results

5.1.1 Case Study 1: Accident Data Analysis

Here, a traffic safety improvement case is studied to detect the time at which shifts in accident frequencies occurred. The number of car accidents per month were collected for 84 consecutive months from January 1989 to January 1995 and are shown in Table 5.1, Table 5.2, Table 5.3, and Table 5.4, respectively. These tables give the monthly accidents for daytime or nighttime only without a dawn or dusk situation. A modification to traffic warning devices was made at railroad crossings (RRx’s) to decrease the frequency of traffic accidents. In order to identify a shift as soon as possible, it was decided to study the data series by using the Tabular CUSUM chart and some related data analysis techniques.

There are various assumptions that must be satisfied before any control chart can be put to use. Under this consideration, the basic assumptions are that the data are independent and normally distributed. The analyst can run the TSIE program first to check if accident data for daytime at passive railroad crossings (Non-Conrail) were drawn at random. The runs test was applied at the 0.01 level of significance to test the
null hypothesis of randomness. The total number of runs in the sequence is 37. Since $z = -1.317$ is greater than -1.96, the null hypothesis of randomness cannot be rejected at the $\alpha = 0.05$ level of significance. It is safe to infer that these accident data have been drawn at random. By running the same procedure, accident data for daytime or nighttime at passive railroad crossing (including Table 5.2, Table 5.3, and Table 5.4) have also been drawn at random.

To check if the accident data shown in Table 5.1 can be assumed to be normally distributed, a Kolmogorov-Smirnov test was constructed and is shown in Figure 5.1. This set of data series, with the exception of one or two points, appears to be well described under the Normal Curve. Since the mean ($\mu = 4.714$) of this data series is slightly close to the variance ($\sigma^2 = 6.255$), one can suspect that it would follow a Poisson distribution. Under this consideration, the analyst observes that the calculated maximum absolute difference (0.155) is less than the critical value (0.2675) found on the table corresponding to a P-value (0.05), so the accident data is normally distributed in this case. With the same procedure, accident data in Table 5.2 and Table 5.3 can be assumed to follow a normal distribution. Figure 5.2 and Figure 5.3 show the results of Kolmogorov-Smirnov normality test. However, accident data shown in Table 5.4 and Figure 5.4 are not normally distributed after the normality test. If the normality assumption cannot be satisfied, one should not do any analysis on this data series, as the result may be misleading.
To continue the data analysis, the analyst can run the appropriate function (Identify Pattern and Select Method) in TSIE program to determine if any seasonal or trend patterns exist in the accident data. Figure 5.5, Figure 5.6, Figure 5.7, and Figure 5.8 show the time series plot of accident data in a graphical chart. After examining these plots, no significant seasonal pattern or trend pattern can be found visually since these accident data are quite small.

Accordingly, tests for trend and seasonality can be applied here to statistically verify the existence of trend or seasonality in the plots. The results for trend tests have shown in Table 5.5, Table 5.6, Table 5.7, and Table 5.8 respectively. After comparing $F_0$ with the $F$ distribution table value ($F_{0.01, 82}$), one can reject $H_1$ and conclude that all four sets of data series have no significant trend in the plots. In examining these tables, one can even consider that the slopes (-0.017, -0.016, 0.01, and -0.0024, respectively) are very small. This means the linear trend drops off slowly and is not necessary to detrend the data (see Appendix B for more information).

The results for seasonality tests are shown in Appendix B. Since the T values in Table B.3 and Table B.4 are smaller than $\chi^2(\alpha = 0.05, d.f. = 11)$ table value, the analyst can infer that no seasonality exist in the accident data. However, T value (22.57, 29.54) in Table B.1 and Table B.2 is slightly larger than table value (19.68). One can infer that a seasonal pattern exists in the accident data. Eliminating seasonality from the data series is needed in order to study its other component uncontaminated by the seasonal pattern (see Appendix B for more information).
After satisfying the basic assumptions, the first Tabular CUSUM chart was constructed from the daytime accident data at passive railroad crossings (Non-Conrail) with \( H=5, \ K=0.2 \) parameters. The extra control limits were set to \( Z=\pm 1.65 \) based on the first three years and the results are shown in Table 5.9 and Figure 5.9, respectively. In examining this CUSUM chart, a down shift occurs in December 1992, which is the twelfth point after the control limits were set up.

The second Tabular CUSUM chart was constructed for the nighttime accident data at passive railroad crossings (Non-Conrail) shown in Table 5.2. The scheme was set up with \( H=5, \ K=0.2, \) and \( Z=\pm 1.65 \) as the first ones. The results are shown in Table 5.10 and Figure 5.10, respectively. A down shift was detected at the sixth point with \( Z = -1.866 \) and at the nineteenth point by the \( H \) value of CUSUM scheme when the \( S_L \) exceeds 5 (in July 1993, \( S_L =5.101 \)).

With the same scheme, the third Tabular CUSUM chart was built for the daytime accident data at passive railroad crossings (Conrail only). Table 5.11 and Figure 5.11 show no out-of-control signal downward after January 1992. The analyst may find some additional shifts, which are not desired, in the chart due to the setting of \( Z \) value that has \( \alpha = 0.05 \).
Table 5.1 Accident Data for Daytime at Passive RRx’s (Non-Conrail)

<table>
<thead>
<tr>
<th>YEAR</th>
<th>SUBTOTAL</th>
<th>JAN</th>
<th>FEB</th>
<th>MAR</th>
<th>APR</th>
<th>MAY</th>
<th>JUN</th>
<th>JUL</th>
<th>AUG</th>
<th>SEP</th>
<th>OCT</th>
<th>NOV</th>
<th>DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1989</td>
<td>71</td>
<td>6</td>
<td>6</td>
<td>2</td>
<td>4</td>
<td>7</td>
<td>8</td>
<td>5</td>
<td>1</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>1990</td>
<td>57</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>5</td>
<td>7</td>
<td>4</td>
<td>7</td>
<td>5</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>1991</td>
<td>65</td>
<td>2</td>
<td>2</td>
<td>7</td>
<td>2</td>
<td>5</td>
<td>11</td>
<td>11</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>1992</td>
<td>49</td>
<td>10</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>1993</td>
<td>58</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>1</td>
<td>7</td>
<td>2</td>
<td>4</td>
<td>7</td>
<td>4</td>
<td>9</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>1994</td>
<td>42</td>
<td>8</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1995</td>
<td>57</td>
<td>10</td>
<td>4</td>
<td>7</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>8</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>

Source: Based on reports from Ohio Department of Transportation (ODOT). All data analysis output are filed in Human Factors and Ergonomics Lab. at Ohio University

Table 5.2 Accident Data for Nighttime at Passive RRx’s (Non-Conrail)

<table>
<thead>
<tr>
<th>YEAR</th>
<th>SUBTOTAL</th>
<th>JAN</th>
<th>FEB</th>
<th>MAR</th>
<th>APR</th>
<th>MAY</th>
<th>JUN</th>
<th>JUL</th>
<th>AUG</th>
<th>SEP</th>
<th>OCT</th>
<th>NOV</th>
<th>DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1989</td>
<td>51</td>
<td>1</td>
<td>5</td>
<td>6</td>
<td>6</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>1990</td>
<td>40</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>1991</td>
<td>43</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>1992</td>
<td>33</td>
<td>4</td>
<td>2</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1993</td>
<td>39</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>1994</td>
<td>38</td>
<td>3</td>
<td>8</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>1995</td>
<td>31</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 5.3 Accident Data for Daytime at Passive RRx’s (Conrail Only)

<table>
<thead>
<tr>
<th>YEAR</th>
<th>SUBTOTAL</th>
<th>JAN</th>
<th>FEB</th>
<th>MAR</th>
<th>APR</th>
<th>MAY</th>
<th>JUN</th>
<th>JUL</th>
<th>AUG</th>
<th>SEP</th>
<th>OCT</th>
<th>NOV</th>
<th>DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1989</td>
<td>36</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1990</td>
<td>41</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>4</td>
<td>6</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>1991</td>
<td>36</td>
<td>5</td>
<td>0</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>3</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>6</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1992</td>
<td>42</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 5.4 Accident Data for Nighttime at Passive RRx’s (Conrail Only)

<table>
<thead>
<tr>
<th>YEAR</th>
<th>SUBTOTAL</th>
<th>JAN</th>
<th>FEB</th>
<th>MAR</th>
<th>APR</th>
<th>MAY</th>
<th>JUN</th>
<th>JUL</th>
<th>AUG</th>
<th>SEP</th>
<th>OCT</th>
<th>NOV</th>
<th>DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1989</td>
<td>42</td>
<td>2</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>1990</td>
<td>23</td>
<td>7</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>1991</td>
<td>37</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>8</td>
<td>5</td>
</tr>
<tr>
<td>1992</td>
<td>23</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
Figure 5.1 Kolmogorov-Smirnov Test for Daytime at Passive RRx’s (Non-Conrail)

Figure 5.2 Kolmogorov-Smirnov Test for Nighttime at Passive RRx’s (Non-Conrail)
Figure 5.3 Kolmogorov-Smirnov Test for Daytime at Passive RRx’s (Conrail Only)

Figure 5.4 Kolmogorov-Smirnov Test for Nighttime at Passive RRx’s (Conrail Only)
Figure 5.5  Time Series Plot for Daytime at Passive RRx's (Non-Conrail)

Figure 5.6  Time Series Plot for Nighttime at Passive RRx's (Non-Conrail)

Figure 5.7  Time Series Plot for Daytime at Passive RRx's (Conrail Only)
Table 5.5 Analysis of Variance for Daytime (Non-Conrail)

<table>
<thead>
<tr>
<th>Source of Variation</th>
<th>Sum of Squares</th>
<th>d.f.</th>
<th>Mean Square</th>
<th>F</th>
<th>Intercept(a)</th>
<th>Slope(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression</td>
<td>14.837</td>
<td>1</td>
<td>14.837</td>
<td>2.413</td>
<td>5.451</td>
<td>-0.017</td>
</tr>
<tr>
<td>Error or residual</td>
<td>504.306</td>
<td>82</td>
<td>6.150</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>519.143</td>
<td>83</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.6 Analysis of Variance for Nighttime (Non-Conrail)

<table>
<thead>
<tr>
<th>Source of Variation</th>
<th>Sum of Squares</th>
<th>d.f.</th>
<th>Mean Square</th>
<th>F</th>
<th>Intercept(a)</th>
<th>Slope(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression</td>
<td>12.589</td>
<td>1</td>
<td>12.589</td>
<td>3.287</td>
<td>3.952</td>
<td>-0.016</td>
</tr>
<tr>
<td>Error or residual</td>
<td>314.113</td>
<td>82</td>
<td>3.831</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>326.702</td>
<td>83</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.7 Analysis of Variance for Daytime (Conrail Only)

<table>
<thead>
<tr>
<th>Source of Variation</th>
<th>Sum of Squares</th>
<th>d.f.</th>
<th>Mean Square</th>
<th>F</th>
<th>Intercept(a)</th>
<th>Slope(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression</td>
<td>0.969</td>
<td>1</td>
<td>0.969</td>
<td>0.415</td>
<td>2.978</td>
<td>0.010</td>
</tr>
<tr>
<td>Error or residual</td>
<td>107.510</td>
<td>46</td>
<td>2.337</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>108.479</td>
<td>47</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.8 Analysis of Variance for Nighttime (Conrail Only)

CR only, Nighttime, Passive RRx only

<table>
<thead>
<tr>
<th>Source of Variation</th>
<th>Sum of Squares</th>
<th>d.f.</th>
<th>Mean Square</th>
<th>F</th>
<th>Intercept(a)</th>
<th>Slope(b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regression</td>
<td>5.183</td>
<td>1</td>
<td>5.183</td>
<td>1.400</td>
<td>3.185</td>
<td>-0.024</td>
</tr>
<tr>
<td>Error or residual</td>
<td>170.297</td>
<td>46</td>
<td>3.702</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>175.479</td>
<td>47</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 5.9 Tabular Cusums for Daytime at Passive RRx’s (Non-Conrail)

<table>
<thead>
<tr>
<th>No</th>
<th>Raw Data</th>
<th>Z Value</th>
<th>CUSUM High</th>
<th>CUSUM Low</th>
<th>Shift(Z)</th>
<th>Shift(H)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>6</td>
<td>0.275</td>
<td>0.075</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-89</td>
<td>6</td>
<td>0.275</td>
<td>0.15</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>1.049</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>1.335</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>7</td>
<td>0.656</td>
<td>0.456</td>
<td>0.479</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>8</td>
<td>1.037</td>
<td>1.293</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>5</td>
<td>-0.106</td>
<td>0.987</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>1</td>
<td>-1.629</td>
<td>0</td>
<td>1.429</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>8</td>
<td>1.037</td>
<td>0.837</td>
<td>0.193</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>8</td>
<td>1.037</td>
<td>1.674</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>8</td>
<td>1.037</td>
<td>2.511</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>8</td>
<td>1.037</td>
<td>3.348</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>4</td>
<td>-0.487</td>
<td>2.661</td>
<td>0.287</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>2</td>
<td>-1.249</td>
<td>1.212</td>
<td>1.335</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>4</td>
<td>-0.487</td>
<td>0.526</td>
<td>1.622</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>3</td>
<td>-0.868</td>
<td>0</td>
<td>2.29</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>6</td>
<td>0.275</td>
<td>0.075</td>
<td>1.815</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>5</td>
<td>-0.106</td>
<td>0</td>
<td>1.72</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>7</td>
<td>0.656</td>
<td>0.456</td>
<td>0.864</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>1.151</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>7</td>
<td>0.656</td>
<td>0.456</td>
<td>0.295</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>7</td>
<td>0.656</td>
<td>0.912</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>1.049</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>3</td>
<td>-0.868</td>
<td>0</td>
<td>1.716</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-91</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>2.765</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-91</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>3.813</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-91</td>
<td>7</td>
<td>0.656</td>
<td>0.456</td>
<td>2.957</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-91</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>4.006</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-91</td>
<td>5</td>
<td>-0.106</td>
<td>0</td>
<td>3.912</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-91</td>
<td>11</td>
<td>2.18</td>
<td>1.98</td>
<td>1.532</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Jul-91</td>
<td>11</td>
<td>2.18</td>
<td>3.959</td>
<td>0</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Aug-91</td>
<td>6</td>
<td>0.275</td>
<td>4.034</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-91</td>
<td>7</td>
<td>0.656</td>
<td>4.49</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-91</td>
<td>7</td>
<td>0.656</td>
<td>4.946</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-91</td>
<td>2</td>
<td>-1.249</td>
<td>3.498</td>
<td>1.049</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-91</td>
<td>3</td>
<td>-0.868</td>
<td>2.43</td>
<td>1.716</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-92</td>
<td>10</td>
<td>1.799</td>
<td>4.029</td>
<td>0</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Feb-92</td>
<td>5</td>
<td>-0.106</td>
<td>3.723</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-92</td>
<td>1</td>
<td>-1.629</td>
<td>1.894</td>
<td>1.429</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-92</td>
<td>2</td>
<td>-1.249</td>
<td>0.445</td>
<td>2.478</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-92</td>
<td>3</td>
<td>-0.868</td>
<td>0</td>
<td>3.146</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-92</td>
<td>5</td>
<td>-0.106</td>
<td>0</td>
<td>3.051</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-92</td>
<td>5</td>
<td>-0.106</td>
<td>0</td>
<td>2.957</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-92</td>
<td>3</td>
<td>-0.868</td>
<td>0</td>
<td>3.625</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-92</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>3.912</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-92</td>
<td>6</td>
<td>0.275</td>
<td>0.075</td>
<td>3.437</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-92</td>
<td>3</td>
<td>-0.868</td>
<td>0</td>
<td>4.104</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-92</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>5.153</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Continued)
<table>
<thead>
<tr>
<th>No</th>
<th>Raw Data</th>
<th>Z Value</th>
<th>CUSUM High</th>
<th>CUSUM Low</th>
<th>Shift(Z)</th>
<th>Shift(H)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-93</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>6.201</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Feb-93</td>
<td>6</td>
<td>0.275</td>
<td>0.075</td>
<td>5.726</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Mar-93</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>6.013</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Apr-93</td>
<td>1</td>
<td>-1.629</td>
<td>0</td>
<td>7.442</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>May-93</td>
<td>7</td>
<td>0.656</td>
<td>0.456</td>
<td>6.586</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Jun-93</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>7.635</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Jul-93</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>7.922</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Aug-93</td>
<td>7</td>
<td>0.656</td>
<td>0.456</td>
<td>7.066</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Sep-93</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>7.352</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Oct-93</td>
<td>9</td>
<td>1.418</td>
<td>1.218</td>
<td>5.734</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Nov-93</td>
<td>4</td>
<td>-0.487</td>
<td>0.531</td>
<td>6.021</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Dec-93</td>
<td>8</td>
<td>1.037</td>
<td>1.368</td>
<td>4.784</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Jan-94</td>
<td>8</td>
<td>1.037</td>
<td>2.205</td>
<td>3.547</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Feb-94</td>
<td>3</td>
<td>-0.868</td>
<td>1.137</td>
<td>4.215</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Mar-94</td>
<td>3</td>
<td>-0.868</td>
<td>0.07</td>
<td>4.883</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Apr-94</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>5.931</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>May-94</td>
<td>6</td>
<td>0.275</td>
<td>0.075</td>
<td>5.456</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Jun-94</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>5.743</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Jul-94</td>
<td>3</td>
<td>-0.868</td>
<td>0</td>
<td>6.41</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Aug-94</td>
<td>5</td>
<td>-0.106</td>
<td>0</td>
<td>6.316</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Sep-94</td>
<td>1</td>
<td>-1.629</td>
<td>0</td>
<td>7.746</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Oct-94</td>
<td>3</td>
<td>-0.868</td>
<td>0</td>
<td>8.413</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Nov-94</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>9.462</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Dec-94</td>
<td>2</td>
<td>-1.249</td>
<td>0</td>
<td>10.51</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Jan-95</td>
<td>10</td>
<td>1.799</td>
<td>1.599</td>
<td>8.512</td>
<td>++++</td>
<td>-----</td>
</tr>
<tr>
<td>Feb-95</td>
<td>4</td>
<td>-0.487</td>
<td>0.912</td>
<td>8.798</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Mar-95</td>
<td>7</td>
<td>0.656</td>
<td>1.368</td>
<td>7.942</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Apr-95</td>
<td>3</td>
<td>-0.868</td>
<td>0.3</td>
<td>8.61</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>May-95</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>8.897</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Jun-95</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>9.183</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Jul-95</td>
<td>3</td>
<td>-0.868</td>
<td>0</td>
<td>9.851</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Aug-95</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>10.138</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Sep-95</td>
<td>1</td>
<td>-1.629</td>
<td>0</td>
<td>11.567</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Oct-95</td>
<td>8</td>
<td>1.037</td>
<td>0.837</td>
<td>10.33</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Nov-95</td>
<td>5</td>
<td>-0.106</td>
<td>0.531</td>
<td>10.236</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>Dec-95</td>
<td>4</td>
<td>-0.487</td>
<td>0</td>
<td>10.523</td>
<td>-----</td>
<td>-----</td>
</tr>
</tbody>
</table>
Table 5.10  Tabular Cusums for Nighttime (Non-Conrail)

<table>
<thead>
<tr>
<th>No</th>
<th>Raw Data</th>
<th>Z Value</th>
<th>CUSUM High</th>
<th>CUSUM Low</th>
<th>Shift(Z)</th>
<th>Shift(H)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>1.165</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-89</td>
<td>5</td>
<td>0.641</td>
<td>0.441</td>
<td>0.324</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>6</td>
<td>1.142</td>
<td>1.383</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>6</td>
<td>1.142</td>
<td>2.325</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>5</td>
<td>0.641</td>
<td>2.765</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>2</td>
<td>-0.864</td>
<td>1.702</td>
<td>0.664</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>2</td>
<td>-0.864</td>
<td>0.638</td>
<td>1.327</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>4</td>
<td>0.139</td>
<td>0.578</td>
<td>0.988</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>1.651</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>4</td>
<td>0.139</td>
<td>0.000</td>
<td>1.312</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>7</td>
<td>1.643</td>
<td>1.443</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>7</td>
<td>1.643</td>
<td>2.887</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>4</td>
<td>0.139</td>
<td>2.826</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>5</td>
<td>0.641</td>
<td>3.267</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>3</td>
<td>-0.362</td>
<td>2.705</td>
<td>0.162</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>3</td>
<td>-0.362</td>
<td>2.143</td>
<td>0.324</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>3</td>
<td>-0.362</td>
<td>1.581</td>
<td>0.486</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>2</td>
<td>-0.864</td>
<td>0.517</td>
<td>1.150</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>4</td>
<td>0.139</td>
<td>0.456</td>
<td>0.811</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>1.474</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>3</td>
<td>-0.362</td>
<td>0.000</td>
<td>1.636</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>2.300</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>7</td>
<td>1.643</td>
<td>1.443</td>
<td>0.456</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>2</td>
<td>-0.864</td>
<td>0.380</td>
<td>1.120</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-91</td>
<td>5</td>
<td>0.641</td>
<td>0.821</td>
<td>0.279</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-91</td>
<td>4</td>
<td>0.139</td>
<td>0.760</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-91</td>
<td>3</td>
<td>-0.362</td>
<td>0.198</td>
<td>0.162</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-91</td>
<td>4</td>
<td>0.139</td>
<td>0.137</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-91</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>1.165</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-91</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>2.330</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-91</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>3.495</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-91</td>
<td>4</td>
<td>0.139</td>
<td>0.000</td>
<td>3.155</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-91</td>
<td>4</td>
<td>0.139</td>
<td>0.000</td>
<td>2.816</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-91</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>3.981</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-91</td>
<td>7</td>
<td>1.643</td>
<td>1.443</td>
<td>2.138</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-91</td>
<td>8</td>
<td>2.145</td>
<td>3.388</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-92</td>
<td>4</td>
<td>0.139</td>
<td>3.328</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-92</td>
<td>2</td>
<td>-0.864</td>
<td>2.264</td>
<td>0.664</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-92</td>
<td>6</td>
<td>1.142</td>
<td>3.206</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-92</td>
<td>1</td>
<td>-1.365</td>
<td>1.641</td>
<td>1.165</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-92</td>
<td>3</td>
<td>-0.362</td>
<td>1.079</td>
<td>1.327</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-92</td>
<td>0</td>
<td>-1.866</td>
<td>0.000</td>
<td>2.993</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-92</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>4.158</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-92</td>
<td>4</td>
<td>0.139</td>
<td>0.000</td>
<td>3.819</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-92</td>
<td>3</td>
<td>-0.362</td>
<td>0.000</td>
<td>3.981</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-92</td>
<td>5</td>
<td>0.641</td>
<td>0.441</td>
<td>3.140</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-92</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>3.804</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-92</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>4.467</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Continued)
<table>
<thead>
<tr>
<th>No</th>
<th>Raw Data</th>
<th>Z Value</th>
<th>CUSUM High</th>
<th>CUSUM Low</th>
<th>Shift(Z)</th>
<th>Shift(H)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-93</td>
<td>3</td>
<td>-0.362</td>
<td>0.000</td>
<td>4.630</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-93</td>
<td>5</td>
<td>0.641</td>
<td>0.411</td>
<td>3.789</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-93</td>
<td>4</td>
<td>0.139</td>
<td>0.380</td>
<td>3.450</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-93</td>
<td>4</td>
<td>0.139</td>
<td>0.319</td>
<td>3.110</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-93</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>3.774</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-93</td>
<td>3</td>
<td>-0.362</td>
<td>0.000</td>
<td>3.936</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-93</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>5.101</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-93</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>6.266</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-93</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>6.929</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-93</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>7.593</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-93</td>
<td>7</td>
<td>1.643</td>
<td>1.443</td>
<td>5.749</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-93</td>
<td>5</td>
<td>0.641</td>
<td>1.884</td>
<td>4.909</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-94</td>
<td>3</td>
<td>-0.362</td>
<td>1.322</td>
<td>5.071</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-94</td>
<td>8</td>
<td>2.145</td>
<td>3.267</td>
<td>2.726</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Mar-94</td>
<td>4</td>
<td>0.139</td>
<td>3.206</td>
<td>2.387</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-94</td>
<td>3</td>
<td>-0.362</td>
<td>2.644</td>
<td>2.549</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-94</td>
<td>1</td>
<td>-1.365</td>
<td>1.079</td>
<td>3.714</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-94</td>
<td>2</td>
<td>-0.864</td>
<td>0.016</td>
<td>4.377</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-94</td>
<td>0</td>
<td>-1.866</td>
<td>0.000</td>
<td>6.043</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-94</td>
<td>3</td>
<td>-0.362</td>
<td>0.000</td>
<td>6.206</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-94</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>7.371</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-94</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>8.535</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-94</td>
<td>9</td>
<td>2.646</td>
<td>2.446</td>
<td>5.699</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Dec-94</td>
<td>3</td>
<td>-0.362</td>
<td>1.884</td>
<td>5.851</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-95</td>
<td>3</td>
<td>-0.362</td>
<td>1.322</td>
<td>6.013</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-95</td>
<td>3</td>
<td>-0.362</td>
<td>0.760</td>
<td>6.176</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-95</td>
<td>4</td>
<td>0.139</td>
<td>0.699</td>
<td>5.836</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-95</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>6.500</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-95</td>
<td>4</td>
<td>0.139</td>
<td>0.000</td>
<td>6.160</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-95</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>6.824</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-95</td>
<td>0</td>
<td>-1.866</td>
<td>0.000</td>
<td>8.490</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-95</td>
<td>3</td>
<td>-0.362</td>
<td>0.000</td>
<td>8.652</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-95</td>
<td>1</td>
<td>-1.365</td>
<td>0.000</td>
<td>9.817</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-95</td>
<td>2</td>
<td>-0.864</td>
<td>0.000</td>
<td>10.481</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-95</td>
<td>3</td>
<td>-0.362</td>
<td>0.000</td>
<td>10.643</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-95</td>
<td>4</td>
<td>0.139</td>
<td>0.000</td>
<td>10.304</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 5.11  Tabular Cusums for Daytime (Conrail Only)

<table>
<thead>
<tr>
<th>No</th>
<th>Raw Data</th>
<th>Z value</th>
<th>CUSUM High</th>
<th>CUSUM Low</th>
<th>Shift(Z)</th>
<th>Shift(H)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>1</td>
<td>-1.332</td>
<td>0.000</td>
<td>1.132</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-89</td>
<td>3</td>
<td>-0.086</td>
<td>0.000</td>
<td>1.018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>4</td>
<td>0.536</td>
<td>0.336</td>
<td>0.282</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>3</td>
<td>-0.086</td>
<td>0.050</td>
<td>0.168</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>1</td>
<td>-1.332</td>
<td>0.000</td>
<td>1.300</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>3</td>
<td>-0.086</td>
<td>0.000</td>
<td>1.186</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>3</td>
<td>-0.086</td>
<td>0.000</td>
<td>1.073</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>1</td>
<td>-1.332</td>
<td>0.000</td>
<td>2.204</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>4</td>
<td>0.536</td>
<td>0.336</td>
<td>1.468</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>6</td>
<td>1.781</td>
<td>1.917</td>
<td>0.000</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>3</td>
<td>-0.086</td>
<td>1.631</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>4</td>
<td>0.536</td>
<td>1.967</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>4</td>
<td>0.536</td>
<td>2.303</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>2</td>
<td>-0.709</td>
<td>1.394</td>
<td>0.509</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>4</td>
<td>0.536</td>
<td>1.730</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>3</td>
<td>-0.086</td>
<td>1.444</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>2</td>
<td>-0.709</td>
<td>0.535</td>
<td>0.509</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>3</td>
<td>-0.086</td>
<td>0.248</td>
<td>0.395</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>6</td>
<td>1.781</td>
<td>1.829</td>
<td>0.000</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>4</td>
<td>0.536</td>
<td>2.165</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>6</td>
<td>1.781</td>
<td>3.747</td>
<td>0.000</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>2</td>
<td>-0.709</td>
<td>2.838</td>
<td>0.509</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>3</td>
<td>-0.086</td>
<td>2.551</td>
<td>0.395</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>2</td>
<td>-0.709</td>
<td>1.642</td>
<td>0.905</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-91</td>
<td>5</td>
<td>1.159</td>
<td>2.601</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-91</td>
<td>0</td>
<td>-1.954</td>
<td>0.447</td>
<td>1.754</td>
<td>----</td>
<td></td>
</tr>
<tr>
<td>Mar-91</td>
<td>4</td>
<td>0.536</td>
<td>0.783</td>
<td>1.018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-91</td>
<td>4</td>
<td>0.536</td>
<td>1.119</td>
<td>0.282</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-91</td>
<td>0</td>
<td>-1.954</td>
<td>0.000</td>
<td>2.036</td>
<td>----</td>
<td></td>
</tr>
<tr>
<td>Jun-91</td>
<td>3</td>
<td>-0.086</td>
<td>0.000</td>
<td>1.923</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-91</td>
<td>5</td>
<td>1.159</td>
<td>0.959</td>
<td>0.564</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-91</td>
<td>3</td>
<td>-0.086</td>
<td>0.672</td>
<td>0.450</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-91</td>
<td>3</td>
<td>-0.086</td>
<td>0.386</td>
<td>0.337</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-91</td>
<td>6</td>
<td>1.781</td>
<td>1.967</td>
<td>0.000</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Nov-91</td>
<td>1</td>
<td>-1.332</td>
<td>0.435</td>
<td>1.132</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-91</td>
<td>2</td>
<td>-0.709</td>
<td>0.000</td>
<td>1.641</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-92</td>
<td>6</td>
<td>1.781</td>
<td>1.581</td>
<td>0.000</td>
<td>++++</td>
<td></td>
</tr>
<tr>
<td>Feb-92</td>
<td>3</td>
<td>-0.086</td>
<td>1.295</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-92</td>
<td>4</td>
<td>0.536</td>
<td>1.631</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-92</td>
<td>3</td>
<td>-0.086</td>
<td>1.344</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-92</td>
<td>3</td>
<td>-0.086</td>
<td>1.058</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-92</td>
<td>1</td>
<td>-1.332</td>
<td>0.000</td>
<td>1.132</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-92</td>
<td>3</td>
<td>-0.086</td>
<td>0.000</td>
<td>1.018</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-92</td>
<td>3</td>
<td>-0.086</td>
<td>0.000</td>
<td>0.905</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-92</td>
<td>5</td>
<td>1.159</td>
<td>0.959</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-92</td>
<td>4</td>
<td>0.536</td>
<td>1.295</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-92</td>
<td>4</td>
<td>0.536</td>
<td>1.631</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-92</td>
<td>3</td>
<td>-0.086</td>
<td>1.344</td>
<td>0.000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 5.9 Tabular CUSUM Chart for Daytime at Passive RRx's (Non-Conrail)

Figure 5.10 Tabular CUSUM Chart for Nighttime at Passive RRx’s (Non-Conrail)
Figure 5.11 Tabular CUSUM Chart for Daytime at Passive RRx's (Conrail Only)

5.1.2 Case Study 2: Performance Comparison

The standardized Shewhart-CUSUM chart, Tabular CUSUM chart, and standardized Shewhart control chart (using Z values for the control chart instead of raw data) are exercised here with a simulation for comparison of speed in detecting mean shifts of various magnitudes. According to the daytime data series at passive RRx's (Non-Conrail) in Table 5.1, the 84 consecutive observations are normally distributed with \( \mu = 4.714 \) and \( \sigma = 2.501 \). Three groups (5% down, 10% down, and 20% down shift in the mean level) of simulated data, which contain 100 sets of 30 random numbers each, are generated from this normal distribution \( N(\mu = 4.714, \sigma = 2.501) \) using MS-EXCEL. Table 5.12 gives the monthly accident data from Table 5.1 and adds the first set of 30 simulated data of a 5% down shift in the mean.
Hence, the analyst can use the 84 consecutive observations and the first set of 30 random data to construct the standardized Shewhart control chart and attempt to detect an out-of-control signal. The control limits of this standardized Shewhart chart were set to \( Z = \pm 1.65 \), with \( \alpha = 0.05 \). Thus, the number of months needed to detect a shift (expected months) can always be determined from the 30 random data. After 99 repetitions with the rest of the 99 sets of random data, a cumulative frequency chart can be produced based on this expected month for the 5% down shift situation. The results are shown in Figure 5.12. In examining this chart, it can be seen that the 5% down curve takes longer to detect the shift compared with the 10% down curve and 20% down curve.

Since \( \alpha = 0.05 \), the ARL \( (ARL = \frac{1}{\alpha}) \) of this chart is 20, theoretically, which means that an out-of-control signal will be generated every 20 samples, on average.

By using the same simulated data and the same procedure, the analyst can construct the tabular CUSUM charts with \( H=5 \), \( K=0.2 \), and \( Z = \pm 1.65 \) for the 5% down, 10% down and 20% down shift in mean of different conditions. In this optimal CUSUM scheme, \( H=5 \) is selected as the threshold value when \( K = 0.2 \) (In-Control ARL = 50). If either the \( S_{Hi} \) or \( S_{Ll} \) value exceeds the \( H \) value at a certain observation of the 30 random numbers, an out-of-control signal is received. Furthermore, a series of \( Z \) values calculated from this 30 consecutive random data can be compared with the critical value, \( \pm 1.65 \) to detect the shifts. With these extra decision rules, one can expect a better performance since the tabular CUSUM chart with \( Z \) values is used. Figure 5.13 shows
the results for three different cases (5% down, 10% down, and 20% down shift in the mean) and the 20% down curve has given a shorter time to detect the right shift.

Finally, the analyst can use the same simulated data and the same procedure continually to form the standardized Shewhart-CUSUM control charts. The control limits were set with $\alpha = 0.05$ and $Z = \pm 1.65$. After 100 repetitions, one can make the same cumulative frequency chart shown in Figure 5.14 for 5% down, 10% down, and 20% down shift in the mean. This figure contains the best results, as all three curves are most bent to the upper-left corner. It is possible to conclude that this CUSUM scheme is a faster choice for detecting small mean shifts. The small ARL value (less than 10) was obtained due to the introduction of extra control limits.

From the data shown in Figure 5.12, Figure 5.13, and Figure 5.14, one can even replot the cumulative frequency chart to compare the performance among three different control charts for a 5% down, 10% down or 20% down shift in mean level. The results are shown in Figure 5.15, Figure 5.16, and Figure 5.17, respectively. In these figures, one can draw the same conclusion that the standardized Shewhart-CUSUM chart is the fastest one to detect the small mean shift. The Tabular CUSUM scheme is the second one to detect the mean shift. The Standardized Shewhart control chart traditionally is slower in detecting the mean shift, and may fail to detect a small shift. However, the analyst can set up a Tabular CUSUM scheme with appropriate FIR value to outperform the standardized Shewhart-CUSUM chart for fast shift detection.
Table 5.12 Monthly Accidents with the First Set of 30 Simulated Data

<table>
<thead>
<tr>
<th>YEAR</th>
<th>JAN</th>
<th>FEB</th>
<th>MAR</th>
<th>APR</th>
<th>MAY</th>
<th>JUN</th>
<th>JUL</th>
<th>AUG</th>
<th>SEP</th>
<th>OCT</th>
<th>NOV</th>
<th>DEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1989</td>
<td>6</td>
<td>6</td>
<td>2</td>
<td>4</td>
<td>7</td>
<td>8</td>
<td>5</td>
<td>1</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>1990</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>6</td>
<td>5</td>
<td>7</td>
<td>4</td>
<td>7</td>
<td>5</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>1991</td>
<td>2</td>
<td>2</td>
<td>7</td>
<td>2</td>
<td>5</td>
<td>11</td>
<td>11</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>1992</td>
<td>10</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>1993</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>1</td>
<td>7</td>
<td>2</td>
<td>4</td>
<td>7</td>
<td>4</td>
<td>9</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>1994</td>
<td>8</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1995</td>
<td>10</td>
<td>4</td>
<td>7</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>8</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>1996</td>
<td>4.1</td>
<td>5.2</td>
<td>4.3</td>
<td>7.8</td>
<td>3.2</td>
<td>5.5</td>
<td>6.2</td>
<td>2.1</td>
<td>5.7</td>
<td>-1.5</td>
<td>9.0</td>
<td>1.8</td>
</tr>
<tr>
<td>1997</td>
<td>0.3</td>
<td>5.9</td>
<td>2.3</td>
<td>4.6</td>
<td>8.4</td>
<td>9.4</td>
<td>5.2</td>
<td>5.0</td>
<td>4.3</td>
<td>2.2</td>
<td>4.5</td>
<td>7.1</td>
</tr>
<tr>
<td>1998</td>
<td>3.9</td>
<td>9</td>
<td>4.7</td>
<td>4.3</td>
<td>5.2</td>
<td>5.9</td>
<td>5.0</td>
<td>4.3</td>
<td>2.2</td>
<td>4.5</td>
<td>7.1</td>
<td>5.9</td>
</tr>
</tbody>
</table>

Note: All data analysis output for simulation runs are filed in Human Factors and Ergonomics Lab. at Ohio University

Figure 5.12 The Performance of Std. Shewhart Control Chart (Alpha = 5%)
Figure 5.13 The Performance of Tabular CUSUM Chart (H=5, K=0.2, Z=±1.65)

Figure 5.14 The Performance of Std. Shewhart-CUSUM Chart (α=.05, Z=±1.65)
Figure 5.15 Performance Comparison for a 5% Down Shift in Mean

Figure 5.16 Performance Comparison for a 10% Down Shift in Mean
Figure 5.17 Performance Comparison for a 20% Down Shift in Mean

5.2 Testing

Testing is a critical part of the software development process. The goal of software testing is to force the software to fail so that errors can be found and then fixed, i.e., software testing usually can be used to provide a systematic method for the elimination of errors. In traditional software development, testing usually comes at the end of the process just before the project is due. Worse yet, testing sometimes becomes an optional step, as developers focus their limited resources less on the testing process and more on completion of the project. However, because of the high cost associated with testing, research has recently begun to focus on software testing and test procedures.
In recent years, testing has been integrated into the entire software life cycle. Test strategies are generated as part of the design and implementation efforts, or at least should be developed in parallel with them. As soon as there is a running system, testing should begin. Whenever possible, a system should be designed specifically so that it is relatively easy to test. In particular, mechanisms for testing can often be designed right into the system.

Lately automated testing tools have made remarkable advances. One can use them to verify the quality of code or interface changes, at any stage of development, with little or no additional resources. This means one can catch defects earlier, when they are easier to fix. Automated testing can also help programmers meet the challenge of shorter project deadlines; where a focus on rapid development over quality testing can cause yet more errors and the need for yet more development work. Fewer maintenance upgrades can reduce significant costs.

5.3 Software Maintenance

Maintenance is the last phase of the software life cycle that begins when software is operational and released to one or more users. Object-oriented design has a different approach in software maintenance (Jacobson 1992). It organizes the program based on objects and structures them from the items which exist in the problem domain (problems come from real life). This is often a more natural way to characterize the objects of the
program. These items are very stable and do not need to change frequently. This advantage exists due to the encapsulation mechanism which has been built right into each class. Thus, each class is independent from any other so that any modification to a class is localized and does not need massive reconstruction. Therefore, software maintenance in an object-oriented design project makes it easier than in a traditional program.
Chapter 6

Discussions, Conclusions and Recommendations

6.1 Discussions and Conclusions

6.1.1 Conclusions in General

1. Based on the studies of the CUSUM control chart techniques in the field of quality control, a modified CUSUM technique (a standardized Shewhart-CUSUM scheme with extra trend identification rules) was introduced, applied successfully, and resulted in early shift detection using simulated accident data.

2. Many different process control charts were investigated in the project to find a technique which provides the shortest run length to detect down shift of 5%, 10%, and 20% in the mean level. The standardized Shewhart-CUSUM chart is considered to be the most efficient technique for monthly accident data in the range of 5 to 20 of ARL. However, the present technique is limited to a small range of significance levels. More research needs to be done before this technique can be applied to various significant levels (like 0.05, 0.01, or 0.00135).

3. Use of such a standardized Shewhart-CUSUM chart is a very good way to monitor the progress of a certain event for the purpose of evaluating the trend or impact of certain modifications that were introduced in the system. Using the example in this
thesis, a safety improvement is made to the system and the analyst can successfully detect this shift by monitoring the movement of the dynamic deviation. The speed of shift detection based on ARL is superior to other control charts.

4. Designed TSIE software provides a number of statistical functions to perform data analysis. Among supported procedures are trend removal, seasonality removal, randomness test, Tabular CUSUM chart, and the standardized Shewhart-CUSUM chart technique. The program was written in C++ with over 4000 lines of source code. With the introduction of user-friendly interfaces, this statistical software package is convenient for most practitioners.

6.1.2 Conclusions in Software Design

What is the major purpose of applying object-oriented design? Suppose one is building a modern automobile. The automobile contains more than 2,000 parts, each part should be well designed and exhibit individually high automation. Designers do not have to make all the parts they need. Designers may have different alternatives to freely choose car parts from either making them themselves or purchasing them from a vendor. The primary job in building a car is to integrate all the parts systematically into a complete product. Each parts should function properly without any ill effect to one another. On the other hand, when some specific series of parts work together, they should perform accurately without any error. Designers are involved in an extremely complex system design.
Since designers have been bombarded with enormous amounts of information in recent years, software development has become a complicated system design. Objects are created like auto parts with some high degree of automation. Data is transferred reliably among the objects through the encapsulation mechanism. If programmers combine the objects, the relations among them are adequate and function well through polymorphism and inheritance mechanisms. When the system becomes large, one can make use of a tree diagram to manage those objects through the class hierarchy mechanism. No matter how complex the system is, programmers can easily make a change to it, like changing auto parts in a car assembly. If programmers do not like any objects, they can use other substitutes, like interchangeable auto parts. Object-oriented design gives us an opportunity to gracefully handle complexity in software development.

6.2 Recommendations

This software package may need new enhanced features. Ideas for future research are:

1. Accident data may not be normally distributed due to the physical nature of accidents. It is possible to devise standardized Shewhart-CUSUM techniques for other distributions, such as Poisson, binomial, and uniform.
2. Most studies of the CUSUM chart techniques in the past were for controlling sample mean. It is possible to develop a new CUSUM scheme for controlling process variability such as ranges or standard deviations (Montgomery 1985).

3. Automatic identification of shift detection in a real time system. Using pattern recognition and a set of user-defined decision rules, a system can detect a shift automatically when a process is going to develop a trend line.

4. Multidimensional analysis with expert systems. Before applying any CUSUM chart techniques, the analyst may need an expert system for selecting an appropriate time series method to preprocess the raw data.

5. In the TSIE program two major classes (CSpreadView class and CChartView class) are applied which come from commercial components. In the future of the software design environment, the time it takes to code and test systems will shrink as one moves into an environment where applications are composed of more and more proven components. With proven objects, one can recycle systems rather than disposing of them.

6. There is no panacea in software design system. The object-oriented design method assumes a very iterative procedure in developing a design. As a result, it is not specific as to what procedures should be applied at each step. Improvements in these areas are critical to the continuing evolution of an object-oriented design system.
Bibliography


Appendix A

TSIE Screen Output
Data Analysis Functions

1. Runs Test

Runs Test can be used to test whether or not the time series data is random.

[OK] [Cancel]
2. Relative Frequency Histogram

Minimum Date: [11]  
Maximum Date: [57]

Minimum Bin Value: [10]  
Maximum Bin Value: [60]  
Bin Range: [8]

Note: No of Data Points = (Max Bin Value - Min Bin Value) / Bin Range  
(must be an integer)

OK  Cancel

3. Identify Pattern and Select a method

Legend:
- Irregular Only
- Irregular + Trend
- Irregular + Seasonal
- Irregular + Trend + Seasonal

2. Pick Up Data Pattern
- Irregular Only
- Irregular and Trend
- Irregular and Seasonal
- Irregular, Trend, Seasonal

3. Comments and Method Setup
- Need to Deserialize
- Time Series Decomposition
- Winter's Exponential Smoothing
### 6-1 Standardized Shewhart-CUSUM Chart

**Shift Detection for Before Data and After Data**

Enter the Number of Points for Before Data: **24**

(Recommend: Greater Than 24 for Good Normal Approximation)

#### Control Limits (Trend Line Rules)

- Type I Error (Alpha = 0.05)
- Type I Error (Alpha = 0.01)

#### Shewhart Control Limits (z)

- None
- Control Limits: **0.00135**

[Buttons: OK, Cancel]

---

### 6-2 Tabular CUSUM Chart (for Individual Measurement)

**Shift Detection for Before Data and After Data**

Enter the Number of Points for Before Data: **24**

(Recommend: Greater Than 24 for Good Normal Approximation)

#### Control Limits (n1)

- Reference Value K: **0.50**
- Decision Interval H: **5.0**

- Use FIR (equal to H/2)

#### Shewhart Control Limits (z)

- None
- Control Limits: **0.00135**

[Buttons: OK, Cancel]
Appendix B

Trend Line Identification Rules

Table B.1 Critical Values for $Z_{0.01}$

<table>
<thead>
<tr>
<th>Points on the Line</th>
<th>Type I Error Individually</th>
<th>Cumulative Standard Normal Probability</th>
<th>Critical Z Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.0014</td>
<td>0.9986</td>
<td>3.00</td>
</tr>
<tr>
<td>3</td>
<td>0.0016</td>
<td>0.9600</td>
<td>1.75</td>
</tr>
<tr>
<td>4</td>
<td>0.0012</td>
<td>0.8937</td>
<td>1.25</td>
</tr>
<tr>
<td>5</td>
<td>0.0012</td>
<td>0.8139</td>
<td>0.90</td>
</tr>
<tr>
<td>6</td>
<td>0.0016</td>
<td>0.7241</td>
<td>0.60</td>
</tr>
<tr>
<td>7</td>
<td>0.0017</td>
<td>0.6545</td>
<td>0.40</td>
</tr>
<tr>
<td>8</td>
<td>0.0017</td>
<td>0.5979</td>
<td>0.25</td>
</tr>
</tbody>
</table>

Table B.2 Critical Values for $Z_{0.05}$

<table>
<thead>
<tr>
<th>Points on the Line</th>
<th>Type I Error Individually</th>
<th>Cumulative Standard Normal Probability</th>
<th>Critical Z Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.0064</td>
<td>0.9936</td>
<td>2.49</td>
</tr>
<tr>
<td>3</td>
<td>0.0064</td>
<td>0.9200</td>
<td>1.41</td>
</tr>
<tr>
<td>4</td>
<td>0.0064</td>
<td>0.8143</td>
<td>0.90</td>
</tr>
<tr>
<td>5</td>
<td>0.0064</td>
<td>0.7172</td>
<td>0.58</td>
</tr>
<tr>
<td>6</td>
<td>0.0064</td>
<td>0.6359</td>
<td>0.35</td>
</tr>
<tr>
<td>7</td>
<td>0.0064</td>
<td>0.5691</td>
<td>0.15</td>
</tr>
<tr>
<td>8</td>
<td>0.0064</td>
<td>0.5140</td>
<td>0.03</td>
</tr>
<tr>
<td>more than 8</td>
<td>0.0064</td>
<td>0.5000</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Notes:

1. Calculation is Based on two Models: $\alpha_o = 1 - \prod_{i=1}^{k}(1 - \alpha_i)$ and $\phi(z) = 1 - n^{-\sqrt{\alpha}}$, where: $k = \text{number of rules}$ and $n = \text{number of points on the line}$

2. $\phi(z) = 1 - n^{-\sqrt{\alpha}}$ was provided by Dr. Richard, Gerth and $\alpha_o = 1 - \prod_{i=1}^{k}(1 - \alpha_i)$ was adapted from Montgomery (1985).

3. Examples:

   $\alpha_o = 1 - \prod_{i=1}^{8}(1 - 0.0064) = 0.05$

   $\phi(z = 2.49) = 1 - 2^{-\sqrt{0.0064}}$
Appendix C

Evaluation of Data Sets for Trend and Seasonality

Table C.1 Rank Test for Daytime after Detrended (Non-Conrail)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>12</td>
<td>2</td>
<td>12</td>
<td>12</td>
<td>49</td>
<td>22.57</td>
</tr>
<tr>
<td>Feb</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>8</td>
<td>8</td>
<td>5</td>
<td>4</td>
<td>34</td>
<td>( \chi^2(0.05,11) = 19.7 )</td>
</tr>
<tr>
<td>Mar</td>
<td>2</td>
<td>6</td>
<td>8</td>
<td>1</td>
<td>4</td>
<td>6</td>
<td>10</td>
<td>37</td>
<td></td>
</tr>
<tr>
<td>Apr</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>May</td>
<td>7</td>
<td>9</td>
<td>6</td>
<td>4</td>
<td>9</td>
<td>11</td>
<td>5</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>Jun</td>
<td>8</td>
<td>8</td>
<td>11</td>
<td>9</td>
<td>3</td>
<td>9</td>
<td>6</td>
<td>54</td>
<td></td>
</tr>
<tr>
<td>Jul</td>
<td>4</td>
<td>10</td>
<td>12</td>
<td>10</td>
<td>5</td>
<td>7</td>
<td>3</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>Aug</td>
<td>1</td>
<td>7</td>
<td>7</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>7</td>
<td>47</td>
<td></td>
</tr>
<tr>
<td>Sep</td>
<td>9</td>
<td>11</td>
<td>9</td>
<td>7</td>
<td>6</td>
<td>1</td>
<td>1</td>
<td>44</td>
<td></td>
</tr>
<tr>
<td>Oct</td>
<td>10</td>
<td>12</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>8</td>
<td>11</td>
<td>74</td>
<td></td>
</tr>
<tr>
<td>Nov</td>
<td>11</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>3</td>
<td>9</td>
<td>42</td>
<td></td>
</tr>
<tr>
<td>Dec</td>
<td>12</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>11</td>
<td>4</td>
<td>8</td>
<td>47</td>
<td></td>
</tr>
</tbody>
</table>

\( \chi^2(0.05,11) = 19.7 \)

Table C.2 Rank Test for Nighttime after Detrended (Non-Conrail)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>1</td>
<td>9</td>
<td>10</td>
<td>9</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>47</td>
<td>29.54</td>
</tr>
<tr>
<td>Feb</td>
<td>7</td>
<td>11</td>
<td>6</td>
<td>4</td>
<td>10</td>
<td>11</td>
<td>7</td>
<td>56</td>
<td>( \chi^2(0.05,11) = 19.7 )</td>
</tr>
<tr>
<td>Mar</td>
<td>9</td>
<td>5</td>
<td>5</td>
<td>12</td>
<td>8</td>
<td>10</td>
<td>10</td>
<td>59</td>
<td></td>
</tr>
<tr>
<td>Apr</td>
<td>10</td>
<td>6</td>
<td>7</td>
<td>2</td>
<td>9</td>
<td>7</td>
<td>3</td>
<td>44</td>
<td></td>
</tr>
<tr>
<td>May</td>
<td>8</td>
<td>7</td>
<td>1</td>
<td>7</td>
<td>3</td>
<td>2</td>
<td>11</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>Jun</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>5</td>
<td>4</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>Jul</td>
<td>3</td>
<td>10</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>Aug</td>
<td>5</td>
<td>2</td>
<td>8</td>
<td>10</td>
<td>2</td>
<td>8</td>
<td>8</td>
<td>43</td>
<td></td>
</tr>
<tr>
<td>Sep</td>
<td>4</td>
<td>8</td>
<td>9</td>
<td>8</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>38</td>
<td></td>
</tr>
<tr>
<td>Oct</td>
<td>6</td>
<td>3</td>
<td>4</td>
<td>11</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>38</td>
<td></td>
</tr>
<tr>
<td>Nov</td>
<td>11</td>
<td>12</td>
<td>11</td>
<td>5</td>
<td>12</td>
<td>12</td>
<td>9</td>
<td>72</td>
<td></td>
</tr>
<tr>
<td>Dec</td>
<td>12</td>
<td>4</td>
<td>12</td>
<td>6</td>
<td>11</td>
<td>9</td>
<td>12</td>
<td>66</td>
<td></td>
</tr>
</tbody>
</table>
### Table C.3 Rank Test for Daytime after Detrended (Conrail Only)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>3</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>36</td>
<td>18.96</td>
</tr>
<tr>
<td>Feb</td>
<td>8</td>
<td>4</td>
<td>2</td>
<td>7</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>Mar</td>
<td>11</td>
<td>9</td>
<td>9</td>
<td>10</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>Apr</td>
<td>7</td>
<td>7</td>
<td>8</td>
<td>6</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>May</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>Jun</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>1</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Jul</td>
<td>5</td>
<td>12</td>
<td>10</td>
<td>4</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>Aug</td>
<td>1</td>
<td>8</td>
<td>6</td>
<td>3</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>Sep</td>
<td>10</td>
<td>11</td>
<td>5</td>
<td>11</td>
<td>37</td>
<td></td>
</tr>
<tr>
<td>Oct</td>
<td>12</td>
<td>2</td>
<td>12</td>
<td>9</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>Nov</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>8</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Dec</td>
<td>9</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>16</td>
<td></td>
</tr>
</tbody>
</table>

\[ \chi^2(0.05,11) = 19.7 \]

### Table C.4 Rank Test for Nighttime after Detrended (Conrail Only)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan</td>
<td>2</td>
<td>12</td>
<td>10</td>
<td>1</td>
<td>25</td>
<td>14.5</td>
</tr>
<tr>
<td>Feb</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>Mar</td>
<td>9</td>
<td>1</td>
<td>6</td>
<td>6</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>Apr</td>
<td>4</td>
<td>8</td>
<td>1</td>
<td>11</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>May</td>
<td>10</td>
<td>10</td>
<td>4</td>
<td>3</td>
<td>27</td>
<td></td>
</tr>
<tr>
<td>Jun</td>
<td>5</td>
<td>9</td>
<td>9</td>
<td>4</td>
<td>27</td>
<td></td>
</tr>
<tr>
<td>Jul</td>
<td>11</td>
<td>2</td>
<td>5</td>
<td>7</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>Aug</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>5</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>Sep</td>
<td>7</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>Oct</td>
<td>12</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>36</td>
<td></td>
</tr>
<tr>
<td>Nov</td>
<td>8</td>
<td>11</td>
<td>12</td>
<td>10</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>Dec</td>
<td>6</td>
<td>3</td>
<td>11</td>
<td>12</td>
<td>32</td>
<td></td>
</tr>
</tbody>
</table>
Figure C.1 Euler Method of the Fourier Series Representation for Daytime at Passive RRx's (Non-Conrail). Model is Based on:

\[
\begin{align*}
F(t) &= 4.75 - 0.26 \cos \left( \frac{pt}{6} \right) - 0.14 \cos \left( \frac{2pt}{6} \right) + 0.38 \cos \left( \frac{3pt}{6} \right) + 0.35 \cos \left( \frac{4pt}{6} \right) + 0.15 \cos \left( \frac{5pt}{6} \right) + 0.41 \cos \left( \frac{6pt}{6} \right) \\
&\quad - 0.33 \sin \left( \frac{pt}{6} \right) - 0.41 \sin \left( \frac{2pt}{6} \right) + 0.84 \sin \left( \frac{3pt}{6} \right) - 0.08 \sin \left( \frac{4pt}{6} \right) - 0.69 \sin \left( \frac{5pt}{6} \right) + 0.01 \sin \left( \frac{6pt}{6} \right)
\end{align*}
\]

Table C.5 Euler Method for Daytime at Passive RRx's (Non-Conrail)

<table>
<thead>
<tr>
<th>Date</th>
<th>Fourier</th>
<th>Observed</th>
<th>Diff. O-F</th>
<th>Date</th>
<th>Fourier</th>
<th>Observed</th>
<th>Diff. O-F</th>
<th>mean</th>
<th>std dev.</th>
<th>max</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>3.8</td>
<td>6</td>
<td>2.2</td>
<td>Jul-92</td>
<td>3.4</td>
<td>5</td>
<td>1.6</td>
<td>0.69</td>
<td>2.72</td>
<td>7.42</td>
<td>-4.2</td>
</tr>
<tr>
<td>Feb-89</td>
<td>4.2</td>
<td>6</td>
<td>1.8</td>
<td>Aug-92</td>
<td>4.0</td>
<td>3</td>
<td>-1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>2.2</td>
<td>2</td>
<td>-0.2</td>
<td>Sep-92</td>
<td>5.6</td>
<td>4</td>
<td>-1.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>5.6</td>
<td>4</td>
<td>-1.6</td>
<td>Oct-92</td>
<td>3.8</td>
<td>6</td>
<td>2.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>5.3</td>
<td>7</td>
<td>1.7</td>
<td>Nov-92</td>
<td>3.3</td>
<td>3</td>
<td>-0.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>5.5</td>
<td>6</td>
<td>2.5</td>
<td>Dec-92</td>
<td>5.4</td>
<td>2</td>
<td>-3.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>4.0</td>
<td>5</td>
<td>1.0</td>
<td>Jan-93</td>
<td>3.0</td>
<td>2</td>
<td>-1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>4.6</td>
<td>1</td>
<td>-3.6</td>
<td>Feb-93</td>
<td>3.4</td>
<td>6</td>
<td>2.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>6.2</td>
<td>8</td>
<td>1.8</td>
<td>Mar-93</td>
<td>3.4</td>
<td>4</td>
<td>2.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>4.3</td>
<td>8</td>
<td>3.7</td>
<td>Apr-93</td>
<td>4.7</td>
<td>1</td>
<td>-3.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>4.0</td>
<td>8</td>
<td>4.0</td>
<td>May-93</td>
<td>4.5</td>
<td>7</td>
<td>2.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>6.0</td>
<td>8</td>
<td>2.0</td>
<td>Jun-93</td>
<td>4.7</td>
<td>2</td>
<td>-2.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>3.6</td>
<td>4</td>
<td>0.4</td>
<td>Jul-93</td>
<td>3.2</td>
<td>4</td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>4.0</td>
<td>2</td>
<td>-2.0</td>
<td>Aug-93</td>
<td>3.8</td>
<td>7</td>
<td>3.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>2.0</td>
<td>4</td>
<td>2.0</td>
<td>Sep-93</td>
<td>5.4</td>
<td>4</td>
<td>-1.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>5.3</td>
<td>3</td>
<td>-2.3</td>
<td>Oct-93</td>
<td>3.6</td>
<td>9</td>
<td>5.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>5.1</td>
<td>6</td>
<td>0.9</td>
<td>Nov-93</td>
<td>3.1</td>
<td>4</td>
<td>0.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>5.3</td>
<td>5</td>
<td>-0.3</td>
<td>Dec-93</td>
<td>5.2</td>
<td>8</td>
<td>2.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>3.8</td>
<td>7</td>
<td>3.2</td>
<td>Jan-94</td>
<td>2.8</td>
<td>8</td>
<td>5.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>4.4</td>
<td>4</td>
<td>-0.4</td>
<td>Feb-94</td>
<td>3.2</td>
<td>3</td>
<td>-0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>6.0</td>
<td>7</td>
<td>1.0</td>
<td>Mar-94</td>
<td>1.1</td>
<td>3</td>
<td>1.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>4.1</td>
<td>7</td>
<td>2.9</td>
<td>Apr-94</td>
<td>4.5</td>
<td>2</td>
<td>-2.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>3.8</td>
<td>2</td>
<td>-1.8</td>
<td>May-94</td>
<td>4.3</td>
<td>6</td>
<td>1.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>5.8</td>
<td>3</td>
<td>-2.8</td>
<td>Jun-94</td>
<td>4.5</td>
<td>4</td>
<td>-0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-91</td>
<td>3.4</td>
<td>2</td>
<td>-1.4</td>
<td>Jul-94</td>
<td>3.0</td>
<td>3</td>
<td>0.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-91</td>
<td>3.8</td>
<td>2</td>
<td>-1.8</td>
<td>Aug-94</td>
<td>3.5</td>
<td>5</td>
<td>1.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-91</td>
<td>1.8</td>
<td>7</td>
<td>5.2</td>
<td>Sep-94</td>
<td>5.2</td>
<td>1</td>
<td>-4.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-91</td>
<td>5.1</td>
<td>2</td>
<td>-3.1</td>
<td>Oct-94</td>
<td>3.4</td>
<td>3</td>
<td>0.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-91</td>
<td>4.9</td>
<td>5</td>
<td>0.1</td>
<td>Nov-94</td>
<td>2.9</td>
<td>2</td>
<td>-0.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-91</td>
<td>5.1</td>
<td>11</td>
<td>5.9</td>
<td>Dec-94</td>
<td>5.0</td>
<td>2</td>
<td>-3.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-91</td>
<td>3.6</td>
<td>11</td>
<td>7.4</td>
<td>Jan-95</td>
<td>2.6</td>
<td>10</td>
<td>7.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-91</td>
<td>4.2</td>
<td>6</td>
<td>1.8</td>
<td>Feb-95</td>
<td>3.0</td>
<td>4</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-91</td>
<td>5.8</td>
<td>7</td>
<td>1.2</td>
<td>Mar-95</td>
<td>0.9</td>
<td>3</td>
<td>6.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-91</td>
<td>4.0</td>
<td>7</td>
<td>3.0</td>
<td>Apr-95</td>
<td>4.2</td>
<td>3</td>
<td>-1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-91</td>
<td>3.6</td>
<td>2</td>
<td>-1.6</td>
<td>May-95</td>
<td>4.1</td>
<td>4</td>
<td>-0.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-91</td>
<td>5.6</td>
<td>3</td>
<td>-2.6</td>
<td>Jun-95</td>
<td>4.3</td>
<td>4</td>
<td>-0.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-92</td>
<td>3.2</td>
<td>10</td>
<td>6.8</td>
<td>Jul-95</td>
<td>2.8</td>
<td>3</td>
<td>0.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-92</td>
<td>3.6</td>
<td>5</td>
<td>1.4</td>
<td>Aug-95</td>
<td>3.3</td>
<td>4</td>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-92</td>
<td>1.6</td>
<td>1</td>
<td>-0.6</td>
<td>Sep-95</td>
<td>5.0</td>
<td>1</td>
<td>-4.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-92</td>
<td>4.9</td>
<td>2</td>
<td>-2.9</td>
<td>Oct-95</td>
<td>3.2</td>
<td>8</td>
<td>4.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-92</td>
<td>4.7</td>
<td>3</td>
<td>-1.7</td>
<td>Nov-95</td>
<td>2.7</td>
<td>5</td>
<td>2.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-92</td>
<td>4.9</td>
<td>5</td>
<td>0.1</td>
<td>Dec-95</td>
<td>4.8</td>
<td>4</td>
<td>-0.8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure C.2 Linear Estimation for Daytime at Passive RRx's (Non-Conrail).
Model is Based on: \( F(t) = 5.45 - 0.017t \)

Table C.6 Linear Estimation for Daytime at Passive RRx's (Non-Conrail)

<table>
<thead>
<tr>
<th>Month</th>
<th>Predicted</th>
<th>Observed</th>
<th>Diff O-P</th>
<th>Mean</th>
<th>Std dev</th>
<th>Max</th>
<th>Min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>5.43</td>
<td>6</td>
<td>0.57</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Feb-89</td>
<td>5.42</td>
<td>6</td>
<td>0.58</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Mar-89</td>
<td>5.40</td>
<td>2</td>
<td>-3.40</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Apr-89</td>
<td>5.38</td>
<td>4</td>
<td>-1.38</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>May-89</td>
<td>5.36</td>
<td>7</td>
<td>1.64</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jun-89</td>
<td>5.35</td>
<td>8</td>
<td>2.65</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jul-89</td>
<td>5.33</td>
<td>5</td>
<td>-0.33</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Aug-89</td>
<td>5.31</td>
<td>1</td>
<td>-4.31</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Sep-89</td>
<td>5.29</td>
<td>9</td>
<td>2.71</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Oct-89</td>
<td>5.28</td>
<td>8</td>
<td>2.72</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Nov-89</td>
<td>5.26</td>
<td>8</td>
<td>2.74</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Dec-89</td>
<td>5.24</td>
<td>8</td>
<td>2.76</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jan-90</td>
<td>5.23</td>
<td>4</td>
<td>-1.23</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Feb-90</td>
<td>5.21</td>
<td>2</td>
<td>-3.21</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Mar-90</td>
<td>5.19</td>
<td>4</td>
<td>-1.19</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Apr-90</td>
<td>5.17</td>
<td>3</td>
<td>-2.17</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>May-90</td>
<td>5.16</td>
<td>6</td>
<td>0.84</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jun-90</td>
<td>5.14</td>
<td>5</td>
<td>-0.14</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jul-90</td>
<td>5.12</td>
<td>7</td>
<td>1.88</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Aug-90</td>
<td>5.10</td>
<td>4</td>
<td>-1.10</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Sep-90</td>
<td>5.09</td>
<td>7</td>
<td>1.91</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Oct-90</td>
<td>5.07</td>
<td>7</td>
<td>1.93</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Nov-90</td>
<td>5.03</td>
<td>4</td>
<td>-3.05</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Dec-90</td>
<td>5.03</td>
<td>3</td>
<td>-2.03</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jan-91</td>
<td>5.02</td>
<td>2</td>
<td>-3.02</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Feb-91</td>
<td>5.00</td>
<td>2</td>
<td>-3.00</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Mar-91</td>
<td>4.98</td>
<td>7</td>
<td>2.02</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Apr-91</td>
<td>4.97</td>
<td>2</td>
<td>-2.97</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>May-91</td>
<td>4.95</td>
<td>5</td>
<td>0.05</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jun-91</td>
<td>4.93</td>
<td>11</td>
<td>5.07</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jul-91</td>
<td>4.91</td>
<td>11</td>
<td>6.09</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Aug-91</td>
<td>4.90</td>
<td>6</td>
<td>1.10</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Sep-91</td>
<td>4.88</td>
<td>7</td>
<td>2.12</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Oct-91</td>
<td>4.86</td>
<td>7</td>
<td>2.14</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Nov-91</td>
<td>4.84</td>
<td>2</td>
<td>-2.84</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Dec-91</td>
<td>4.83</td>
<td>3</td>
<td>-1.83</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jan-92</td>
<td>4.81</td>
<td>10</td>
<td>5.19</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Feb-92</td>
<td>4.79</td>
<td>5</td>
<td>0.21</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Mar-92</td>
<td>4.77</td>
<td>1</td>
<td>-3.77</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Apr-92</td>
<td>4.76</td>
<td>2</td>
<td>-2.76</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>May-92</td>
<td>4.74</td>
<td>3</td>
<td>-1.74</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
<tr>
<td>Jun-92</td>
<td>4.72</td>
<td>5</td>
<td>0.28</td>
<td>0.00</td>
<td>2.46</td>
<td>6.09</td>
<td>-4.31</td>
</tr>
</tbody>
</table>

Figures and tables are included to support the text. The table provides data for the years 1989 to 1995, showing monthly predictions and observations for RRx's.
Figure C.3 Multiple Regression Approximation for Daytime at Passive RRx's (Non-Conrail).

\[ F(t) = 5.45 - 0.017t + 0.12\cos\left(\frac{pt}{6}\right) + 0.15\cos\left(\frac{4pt}{6}\right) + 0.29\cos\left(\frac{5pt}{6}\right) - 0.44\cos\left(\frac{6pt}{6}\right) + 0.15\sin\left(\frac{2pt}{6}\right) \]

Table C.7 Multiple Regression Approximation for Daytime at Passive RRx's (Non-Conrail)

<table>
<thead>
<tr>
<th>Date</th>
<th>( F(T) )</th>
<th>Observed</th>
<th>Diff</th>
<th>Diff. ( F(T) )</th>
<th>Date</th>
<th>( F(T) )</th>
<th>Observed</th>
<th>Diff</th>
<th>Diff. ( F(T) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>5.53</td>
<td>6</td>
<td>0.47</td>
<td>5.08</td>
<td>Jul-92</td>
<td>5.08</td>
<td>5</td>
<td>-0.08</td>
<td>std. dev. 2.48</td>
</tr>
<tr>
<td>Feb-89</td>
<td>4.98</td>
<td>6</td>
<td>1.02</td>
<td>Aug-92 3.84</td>
<td>3</td>
<td>-0.84</td>
<td>max 6.77</td>
<td>min -4.37</td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>5.99</td>
<td>2</td>
<td>-3.99</td>
<td>Sep-92 5.26</td>
<td>4</td>
<td>-1.36</td>
<td>R 0.21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>4.79</td>
<td>4</td>
<td>-0.79</td>
<td>Oct-92 4.47</td>
<td>6</td>
<td>1.53</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>6.00</td>
<td>7</td>
<td>1.00</td>
<td>Nov-92 4.99</td>
<td>3</td>
<td>-1.99</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>4.65</td>
<td>8</td>
<td>3.35</td>
<td>Dec-92 4.74</td>
<td>2</td>
<td>-2.74</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>5.71</td>
<td>5</td>
<td>-0.71</td>
<td>Jan-93 4.69</td>
<td>2</td>
<td>-2.69</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>4.46</td>
<td>1</td>
<td>-3.46</td>
<td>Feb-93 4.14</td>
<td>6</td>
<td>1.86</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>5.89</td>
<td>8</td>
<td>2.11</td>
<td>Mar-93 5.16</td>
<td>4</td>
<td>-1.16</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>5.09</td>
<td>8</td>
<td>2.91</td>
<td>Apr-93 3.98</td>
<td>1</td>
<td>-2.96</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>5.61</td>
<td>8</td>
<td>2.39</td>
<td>May-93 5.17</td>
<td>7</td>
<td>1.83</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>5.36</td>
<td>8</td>
<td>2.64</td>
<td>Jun-93 3.82</td>
<td>2</td>
<td>-1.82</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>5.32</td>
<td>4</td>
<td>-1.32</td>
<td>Jul-93 4.88</td>
<td>4</td>
<td>-0.88</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>4.77</td>
<td>2</td>
<td>-2.77</td>
<td>Aug-93 3.63</td>
<td>7</td>
<td>3.37</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>5.78</td>
<td>4</td>
<td>-1.78</td>
<td>Sep-93 5.05</td>
<td>4</td>
<td>-1.05</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>4.58</td>
<td>3</td>
<td>-1.58</td>
<td>Oct-93 4.26</td>
<td>9</td>
<td>4.74</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>5.79</td>
<td>6</td>
<td>0.21</td>
<td>Nov-93 4.70</td>
<td>4</td>
<td>-0.78</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>4.44</td>
<td>5</td>
<td>0.56</td>
<td>Dec-93 4.53</td>
<td>8</td>
<td>3.47</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>5.50</td>
<td>7</td>
<td>1.50</td>
<td>Jan-94 4.49</td>
<td>8</td>
<td>3.51</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>4.26</td>
<td>4</td>
<td>-0.26</td>
<td>Feb-94 3.94</td>
<td>3</td>
<td>-0.94</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>5.68</td>
<td>7</td>
<td>1.32</td>
<td>Mar-94 4.95</td>
<td>3</td>
<td>-1.95</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>4.89</td>
<td>7</td>
<td>2.11</td>
<td>Apr-94 3.75</td>
<td>2</td>
<td>-1.75</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>5.40</td>
<td>2</td>
<td>-3.40</td>
<td>May-94 4.96</td>
<td>6</td>
<td>1.04</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>5.16</td>
<td>3</td>
<td>-2.16</td>
<td>Jun-94 3.81</td>
<td>4</td>
<td>0.39</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-91</td>
<td>5.11</td>
<td>2</td>
<td>-3.11</td>
<td>Jul-94 4.67</td>
<td>3</td>
<td>-1.57</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-91</td>
<td>4.56</td>
<td>2</td>
<td>-2.56</td>
<td>Aug-94 3.42</td>
<td>5</td>
<td>1.58</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-91</td>
<td>5.57</td>
<td>7</td>
<td>1.43</td>
<td>Sep-94 4.85</td>
<td>1</td>
<td>-3.85</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-91</td>
<td>4.37</td>
<td>2</td>
<td>-2.37</td>
<td>Oct-94 4.05</td>
<td>3</td>
<td>-1.05</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-91</td>
<td>5.58</td>
<td>5</td>
<td>-0.58</td>
<td>Nov-94 4.57</td>
<td>2</td>
<td>-2.57</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-91</td>
<td>4.23</td>
<td>11</td>
<td>6.77</td>
<td>Dec-94 4.32</td>
<td>2</td>
<td>-2.32</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-91</td>
<td>5.29</td>
<td>11</td>
<td>5.71</td>
<td>Jan-95 4.23</td>
<td>10</td>
<td>5.72</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-91</td>
<td>4.05</td>
<td>6</td>
<td>1.95</td>
<td>Feb-95 3.73</td>
<td>4</td>
<td>0.27</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-91</td>
<td>5.47</td>
<td>7</td>
<td>1.53</td>
<td>Mar-95 4.74</td>
<td>7</td>
<td>2.26</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-91</td>
<td>4.68</td>
<td>7</td>
<td>2.32</td>
<td>Apr-95 3.54</td>
<td>3</td>
<td>-0.54</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-91</td>
<td>5.19</td>
<td>2</td>
<td>-3.19</td>
<td>May-95 4.75</td>
<td>4</td>
<td>-0.75</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-91</td>
<td>4.95</td>
<td>3</td>
<td>-1.95</td>
<td>Jun-95 3.40</td>
<td>4</td>
<td>0.60</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-92</td>
<td>4.90</td>
<td>10</td>
<td>5.10</td>
<td>Jul-95 4.46</td>
<td>3</td>
<td>-1.46</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-92</td>
<td>4.35</td>
<td>5</td>
<td>0.66</td>
<td>Aug-95 3.22</td>
<td>4</td>
<td>0.78</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-92</td>
<td>5.37</td>
<td>1</td>
<td>-4.37</td>
<td>Sep-95 3.64</td>
<td>1</td>
<td>-3.64</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-92</td>
<td>4.17</td>
<td>2</td>
<td>-2.17</td>
<td>Oct-95 3.85</td>
<td>8</td>
<td>4.15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-92</td>
<td>5.38</td>
<td>3</td>
<td>-2.38</td>
<td>Nov-95 4.36</td>
<td>5</td>
<td>0.64</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-92</td>
<td>4.03</td>
<td>5</td>
<td>0.97</td>
<td>Dec-95 4.12</td>
<td>4</td>
<td>-0.12</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure C.4 Euler Method of the Fourier Series Representation for Nighttime at Passive RRx's (Non-Conrail). Model is Based on:

\[
F(t) = 3.27 - 1.39\cos\left(\frac{\pi t}{6}\right) - 0.33\cos\left(\frac{2\pi t}{6}\right) - 0.55\cos\left(\frac{3\pi t}{6}\right) - 0.70\cos\left(\frac{4\pi t}{6}\right) + 0.16\cos\left(\frac{5\pi t}{6}\right) + 0.07\cos\left(\frac{6\pi t}{6}\right)
- 0.14\sin\left(\frac{\pi t}{6}\right) - 0.13\sin\left(\frac{2\pi t}{6}\right) - 0.33\sin\left(\frac{3\pi t}{6}\right) + 0.52\sin\left(\frac{4\pi t}{6}\right) + 0.23\sin\left(\frac{5\pi t}{6}\right)
\]

Table C.8 Euler Method for Nighttime at Passive RRx's (Non-Conrail)

<table>
<thead>
<tr>
<th>Date</th>
<th>Fourier</th>
<th>Observed</th>
<th>Diff O-F</th>
<th>Date</th>
<th>Fourier</th>
<th>Observed</th>
<th>Diff O-F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>4.54</td>
<td>4</td>
<td>-3.54</td>
<td>Jul-92</td>
<td>2.94</td>
<td>1</td>
<td>-1.94</td>
</tr>
<tr>
<td>Feb-89</td>
<td>4.32</td>
<td>5</td>
<td>0.68</td>
<td>Aug-92</td>
<td>2.36</td>
<td>4</td>
<td>1.64</td>
</tr>
<tr>
<td>Mar-89</td>
<td>3.25</td>
<td>6</td>
<td>2.75</td>
<td>Sep-92</td>
<td>2.33</td>
<td>3</td>
<td>0.67</td>
</tr>
<tr>
<td>Apr-89</td>
<td>2.75</td>
<td>6</td>
<td>3.25</td>
<td>Oct-92</td>
<td>6.01</td>
<td>5</td>
<td>-1.01</td>
</tr>
<tr>
<td>May-89</td>
<td>1.68</td>
<td>5</td>
<td>3.32</td>
<td>Nov-92</td>
<td>4.44</td>
<td>2</td>
<td>-2.44</td>
</tr>
<tr>
<td>Jun-89</td>
<td>1.32</td>
<td>2</td>
<td>0.68</td>
<td>Dec-92</td>
<td>3.29</td>
<td>2</td>
<td>-1.29</td>
</tr>
<tr>
<td>Jul-89</td>
<td>2.96</td>
<td>2</td>
<td>-0.96</td>
<td>Jan-93</td>
<td>4.53</td>
<td>3</td>
<td>-1.53</td>
</tr>
<tr>
<td>Aug-89</td>
<td>2.32</td>
<td>4</td>
<td>1.68</td>
<td>Feb-93</td>
<td>4.34</td>
<td>5</td>
<td>0.68</td>
</tr>
<tr>
<td>Sep-89</td>
<td>2.39</td>
<td>2</td>
<td>-0.39</td>
<td>Mar-93</td>
<td>3.27</td>
<td>4</td>
<td>0.73</td>
</tr>
<tr>
<td>Oct-89</td>
<td>6.04</td>
<td>4</td>
<td>-2.04</td>
<td>Apr-93</td>
<td>2.76</td>
<td>4</td>
<td>1.24</td>
</tr>
<tr>
<td>Nov-89</td>
<td>4.38</td>
<td>7</td>
<td>2.62</td>
<td>May-93</td>
<td>1.71</td>
<td>2</td>
<td>0.29</td>
</tr>
<tr>
<td>Dec-89</td>
<td>3.31</td>
<td>7</td>
<td>3.69</td>
<td>Jun-93</td>
<td>1.30</td>
<td>3</td>
<td>-1.70</td>
</tr>
<tr>
<td>Jan-90</td>
<td>4.54</td>
<td>4</td>
<td>-0.54</td>
<td>Jul-93</td>
<td>2.93</td>
<td>1</td>
<td>-1.93</td>
</tr>
<tr>
<td>Feb-90</td>
<td>4.33</td>
<td>5</td>
<td>0.67</td>
<td>Aug-93</td>
<td>2.37</td>
<td>1</td>
<td>-1.37</td>
</tr>
<tr>
<td>Mar-90</td>
<td>3.26</td>
<td>3</td>
<td>-0.26</td>
<td>Sep-93</td>
<td>2.32</td>
<td>2</td>
<td>-0.32</td>
</tr>
<tr>
<td>Apr-90</td>
<td>2.75</td>
<td>3</td>
<td>0.25</td>
<td>Oct-93</td>
<td>6.00</td>
<td>2</td>
<td>-4.00</td>
</tr>
<tr>
<td>May-90</td>
<td>1.69</td>
<td>3</td>
<td>1.31</td>
<td>Nov-93</td>
<td>4.46</td>
<td>7</td>
<td>2.54</td>
</tr>
<tr>
<td>Jun-90</td>
<td>1.32</td>
<td>2</td>
<td>0.68</td>
<td>Dec-93</td>
<td>3.29</td>
<td>5</td>
<td>1.71</td>
</tr>
<tr>
<td>Jul-90</td>
<td>2.86</td>
<td>4</td>
<td>1.04</td>
<td>Jan-94</td>
<td>4.52</td>
<td>3</td>
<td>-1.52</td>
</tr>
<tr>
<td>Aug-90</td>
<td>2.34</td>
<td>2</td>
<td>-0.34</td>
<td>Feb-94</td>
<td>4.35</td>
<td>8</td>
<td>3.65</td>
</tr>
<tr>
<td>Sep-90</td>
<td>2.37</td>
<td>3</td>
<td>0.63</td>
<td>Mar-94</td>
<td>3.28</td>
<td>4</td>
<td>0.72</td>
</tr>
<tr>
<td>Oct-90</td>
<td>6.03</td>
<td>2</td>
<td>-4.03</td>
<td>Apr-94</td>
<td>2.77</td>
<td>3</td>
<td>0.23</td>
</tr>
<tr>
<td>Nov-90</td>
<td>4.40</td>
<td>7</td>
<td>2.60</td>
<td>May-94</td>
<td>1.72</td>
<td>1</td>
<td>-0.72</td>
</tr>
<tr>
<td>Dec-90</td>
<td>3.31</td>
<td>2</td>
<td>-1.31</td>
<td>Jun-94</td>
<td>1.29</td>
<td>2</td>
<td>0.71</td>
</tr>
<tr>
<td>Jan-91</td>
<td>4.35</td>
<td>5</td>
<td>0.47</td>
<td>Jul-94</td>
<td>2.92</td>
<td>0</td>
<td>-2.92</td>
</tr>
<tr>
<td>Feb-91</td>
<td>4.33</td>
<td>4</td>
<td>-0.33</td>
<td>Aug-94</td>
<td>2.38</td>
<td>3</td>
<td>0.62</td>
</tr>
<tr>
<td>Mar-91</td>
<td>3.26</td>
<td>3</td>
<td>-0.26</td>
<td>Sep-94</td>
<td>2.30</td>
<td>1</td>
<td>-1.30</td>
</tr>
<tr>
<td>Apr-91</td>
<td>2.76</td>
<td>4</td>
<td>1.24</td>
<td>Oct-94</td>
<td>5.99</td>
<td>1</td>
<td>-4.99</td>
</tr>
<tr>
<td>May-91</td>
<td>1.69</td>
<td>1</td>
<td>-0.59</td>
<td>Nov-94</td>
<td>4.48</td>
<td>9</td>
<td>4.52</td>
</tr>
<tr>
<td>Jun-91</td>
<td>1.31</td>
<td>1</td>
<td>-0.31</td>
<td>Dec-94</td>
<td>3.28</td>
<td>3</td>
<td>-0.28</td>
</tr>
<tr>
<td>Jul-91</td>
<td>2.95</td>
<td>1</td>
<td>-1.95</td>
<td>Jan-95</td>
<td>4.52</td>
<td>3</td>
<td>-1.52</td>
</tr>
<tr>
<td>Aug-91</td>
<td>2.35</td>
<td>4</td>
<td>1.65</td>
<td>Feb-95</td>
<td>4.36</td>
<td>3</td>
<td>-1.36</td>
</tr>
<tr>
<td>Sep-91</td>
<td>2.35</td>
<td>4</td>
<td>1.65</td>
<td>Mar-95</td>
<td>3.28</td>
<td>4</td>
<td>0.72</td>
</tr>
<tr>
<td>Oct-91</td>
<td>6.02</td>
<td>1</td>
<td>-5.02</td>
<td>Apr-95</td>
<td>2.77</td>
<td>2</td>
<td>-0.77</td>
</tr>
<tr>
<td>Nov-91</td>
<td>4.42</td>
<td>7</td>
<td>2.58</td>
<td>May-95</td>
<td>1.73</td>
<td>4</td>
<td>2.77</td>
</tr>
<tr>
<td>Dec-91</td>
<td>3.30</td>
<td>8</td>
<td>4.70</td>
<td>Jun-95</td>
<td>1.29</td>
<td>2</td>
<td>0.71</td>
</tr>
<tr>
<td>Jan-92</td>
<td>4.53</td>
<td>4</td>
<td>-0.53</td>
<td>Jul-95</td>
<td>2.92</td>
<td>0</td>
<td>-2.92</td>
</tr>
<tr>
<td>Feb-92</td>
<td>4.34</td>
<td>2</td>
<td>-2.34</td>
<td>Aug-95</td>
<td>2.40</td>
<td>3</td>
<td>0.60</td>
</tr>
<tr>
<td>Mar-92</td>
<td>3.27</td>
<td>6</td>
<td>2.73</td>
<td>Sep-95</td>
<td>2.28</td>
<td>1</td>
<td>-1.28</td>
</tr>
<tr>
<td>Apr-92</td>
<td>2.76</td>
<td>1</td>
<td>-1.76</td>
<td>Oct-95</td>
<td>5.98</td>
<td>2</td>
<td>-3.98</td>
</tr>
<tr>
<td>May-92</td>
<td>1.70</td>
<td>3</td>
<td>1.30</td>
<td>Nov-95</td>
<td>4.50</td>
<td>3</td>
<td>-1.50</td>
</tr>
<tr>
<td>Jun-92</td>
<td>1.31</td>
<td>0</td>
<td>-1.31</td>
<td>Dec-95</td>
<td>3.28</td>
<td>4</td>
<td>0.72</td>
</tr>
</tbody>
</table>
Figure C.5 Linear Estimation for Nighttime at Passive RRx's (Non-Conrail).

Model is Based on: \( F(t) = 3.59 - 0.016t \)

<table>
<thead>
<tr>
<th>Date</th>
<th>Predicted</th>
<th>Observed</th>
<th>Diff O-P</th>
<th>Date</th>
<th>Predicted</th>
<th>Observed</th>
<th>Diff O-P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>3.94</td>
<td>1</td>
<td>-2.94</td>
<td>Jul-92</td>
<td>3.27</td>
<td>1</td>
<td>-2.27</td>
</tr>
<tr>
<td>Feb-89</td>
<td>3.92</td>
<td>5</td>
<td>1.08</td>
<td>Aug-92</td>
<td>3.25</td>
<td>4</td>
<td>0.75</td>
</tr>
<tr>
<td>Mar-89</td>
<td>3.90</td>
<td>6</td>
<td>2.10</td>
<td>Sep-92</td>
<td>3.23</td>
<td>3</td>
<td>-0.23</td>
</tr>
<tr>
<td>Apr-89</td>
<td>3.89</td>
<td>6</td>
<td>2.11</td>
<td>Oct-92</td>
<td>3.22</td>
<td>5</td>
<td>1.78</td>
</tr>
<tr>
<td>May-89</td>
<td>3.87</td>
<td>5</td>
<td>1.13</td>
<td>Nov-92</td>
<td>3.20</td>
<td>2</td>
<td>-1.20</td>
</tr>
<tr>
<td>Jun-89</td>
<td>3.86</td>
<td>2</td>
<td>-1.86</td>
<td>Dec-92</td>
<td>3.19</td>
<td>2</td>
<td>-1.19</td>
</tr>
<tr>
<td>Jul-89</td>
<td>3.84</td>
<td>2</td>
<td>-1.84</td>
<td>Jan-93</td>
<td>3.17</td>
<td>3</td>
<td>-0.17</td>
</tr>
<tr>
<td>Aug-89</td>
<td>3.82</td>
<td>4</td>
<td>0.18</td>
<td>Feb-93</td>
<td>3.15</td>
<td>5</td>
<td>1.85</td>
</tr>
<tr>
<td>Sep-89</td>
<td>3.81</td>
<td>2</td>
<td>-1.81</td>
<td>Mar-93</td>
<td>3.14</td>
<td>4</td>
<td>0.86</td>
</tr>
<tr>
<td>Oct-89</td>
<td>3.79</td>
<td>4</td>
<td>0.21</td>
<td>Apr-93</td>
<td>3.12</td>
<td>4</td>
<td>0.88</td>
</tr>
<tr>
<td>Nov-89</td>
<td>3.78</td>
<td>7</td>
<td>3.22</td>
<td>May-93</td>
<td>3.11</td>
<td>2</td>
<td>-1.11</td>
</tr>
<tr>
<td>Dec-89</td>
<td>3.76</td>
<td>7</td>
<td>3.24</td>
<td>Jun-93</td>
<td>3.09</td>
<td>3</td>
<td>-0.09</td>
</tr>
<tr>
<td>Jan-90</td>
<td>3.74</td>
<td>4</td>
<td>0.26</td>
<td>Jul-93</td>
<td>3.07</td>
<td>1</td>
<td>-2.07</td>
</tr>
<tr>
<td>Feb-90</td>
<td>3.73</td>
<td>5</td>
<td>1.27</td>
<td>Aug-93</td>
<td>3.06</td>
<td>1</td>
<td>-2.06</td>
</tr>
<tr>
<td>Mar-90</td>
<td>3.71</td>
<td>3</td>
<td>-0.71</td>
<td>Sep-93</td>
<td>3.04</td>
<td>2</td>
<td>-1.04</td>
</tr>
<tr>
<td>Apr-90</td>
<td>3.70</td>
<td>3</td>
<td>-0.70</td>
<td>Oct-93</td>
<td>3.03</td>
<td>2</td>
<td>-1.03</td>
</tr>
<tr>
<td>May-90</td>
<td>3.68</td>
<td>3</td>
<td>-0.68</td>
<td>Nov-93</td>
<td>3.01</td>
<td>7</td>
<td>3.99</td>
</tr>
<tr>
<td>Jun-90</td>
<td>3.66</td>
<td>2</td>
<td>-1.66</td>
<td>Dec-93</td>
<td>2.99</td>
<td>5</td>
<td>2.01</td>
</tr>
<tr>
<td>Jul-90</td>
<td>3.65</td>
<td>4</td>
<td>0.35</td>
<td>Jan-94</td>
<td>2.98</td>
<td>3</td>
<td>0.02</td>
</tr>
<tr>
<td>Aug-90</td>
<td>3.63</td>
<td>2</td>
<td>-1.63</td>
<td>Feb-94</td>
<td>2.96</td>
<td>8</td>
<td>5.04</td>
</tr>
<tr>
<td>Sep-90</td>
<td>3.62</td>
<td>3</td>
<td>-0.62</td>
<td>Mar-94</td>
<td>2.95</td>
<td>4</td>
<td>1.05</td>
</tr>
<tr>
<td>Oct-90</td>
<td>3.60</td>
<td>2</td>
<td>-1.60</td>
<td>Apr-94</td>
<td>2.93</td>
<td>3</td>
<td>0.07</td>
</tr>
<tr>
<td>Nov-90</td>
<td>3.59</td>
<td>7</td>
<td>3.41</td>
<td>May-94</td>
<td>2.91</td>
<td>1</td>
<td>-1.91</td>
</tr>
<tr>
<td>Dec-90</td>
<td>3.57</td>
<td>2</td>
<td>-1.57</td>
<td>Jun-94</td>
<td>2.90</td>
<td>2</td>
<td>-0.90</td>
</tr>
<tr>
<td>Jan-91</td>
<td>3.55</td>
<td>5</td>
<td>1.45</td>
<td>Jul-94</td>
<td>2.88</td>
<td>0</td>
<td>-2.88</td>
</tr>
<tr>
<td>Feb-91</td>
<td>3.54</td>
<td>4</td>
<td>0.46</td>
<td>Aug-94</td>
<td>2.87</td>
<td>3</td>
<td>0.13</td>
</tr>
<tr>
<td>Mar-91</td>
<td>3.52</td>
<td>3</td>
<td>-0.52</td>
<td>Sep-94</td>
<td>2.85</td>
<td>1</td>
<td>-1.85</td>
</tr>
<tr>
<td>Apr-91</td>
<td>3.51</td>
<td>4</td>
<td>0.49</td>
<td>Oct-94</td>
<td>2.83</td>
<td>1</td>
<td>-1.83</td>
</tr>
<tr>
<td>May-91</td>
<td>3.49</td>
<td>1</td>
<td>-2.49</td>
<td>Nov-94</td>
<td>2.82</td>
<td>9</td>
<td>6.18</td>
</tr>
<tr>
<td>Jun-91</td>
<td>3.47</td>
<td>1</td>
<td>-2.47</td>
<td>Dec-94</td>
<td>2.80</td>
<td>3</td>
<td>0.20</td>
</tr>
<tr>
<td>Jul-91</td>
<td>3.46</td>
<td>1</td>
<td>-2.46</td>
<td>Jan-95</td>
<td>2.79</td>
<td>3</td>
<td>0.21</td>
</tr>
<tr>
<td>Aug-91</td>
<td>3.44</td>
<td>4</td>
<td>0.56</td>
<td>Feb-95</td>
<td>2.77</td>
<td>3</td>
<td>0.23</td>
</tr>
<tr>
<td>Sep-91</td>
<td>3.43</td>
<td>4</td>
<td>0.57</td>
<td>Mar-95</td>
<td>2.75</td>
<td>4</td>
<td>1.25</td>
</tr>
<tr>
<td>Oct-91</td>
<td>3.41</td>
<td>1</td>
<td>-2.41</td>
<td>Apr-95</td>
<td>2.74</td>
<td>2</td>
<td>-0.74</td>
</tr>
<tr>
<td>Nov-91</td>
<td>3.39</td>
<td>7</td>
<td>3.61</td>
<td>May-95</td>
<td>2.72</td>
<td>4</td>
<td>1.28</td>
</tr>
<tr>
<td>Dec-91</td>
<td>3.38</td>
<td>6</td>
<td>4.62</td>
<td>Jun-95</td>
<td>2.71</td>
<td>2</td>
<td>-0.71</td>
</tr>
<tr>
<td>Jan-92</td>
<td>3.36</td>
<td>4</td>
<td>0.64</td>
<td>Jul-95</td>
<td>2.69</td>
<td>0</td>
<td>-2.69</td>
</tr>
<tr>
<td>Feb-92</td>
<td>3.35</td>
<td>2</td>
<td>-1.35</td>
<td>Aug-95</td>
<td>2.67</td>
<td>3</td>
<td>0.33</td>
</tr>
<tr>
<td>Mar-92</td>
<td>3.33</td>
<td>6</td>
<td>2.67</td>
<td>Sep-95</td>
<td>2.66</td>
<td>1</td>
<td>-1.68</td>
</tr>
<tr>
<td>Apr-92</td>
<td>3.31</td>
<td>1</td>
<td>-2.31</td>
<td>Oct-95</td>
<td>2.64</td>
<td>2</td>
<td>-0.66</td>
</tr>
<tr>
<td>May-92</td>
<td>3.30</td>
<td>3</td>
<td>-0.30</td>
<td>Nov-95</td>
<td>2.63</td>
<td>3</td>
<td>0.37</td>
</tr>
<tr>
<td>Jun-92</td>
<td>3.28</td>
<td>0</td>
<td>-3.28</td>
<td>Dec-95</td>
<td>2.61</td>
<td>4</td>
<td>1.39</td>
</tr>
</tbody>
</table>
Figure C.6 Multiple Regression Approximation for Nighttime at Passive RRx's (Non-Conrail).

\[ f(t) = 3.95 - 0.016t + 0.19\cos\left(\frac{2\pi t}{6}\right) + 0.14\cos\left(\frac{2\pi t}{5}\right) + 0.05\sin\left(\frac{2\pi t}{4}\right) - 0.47\cos\left(\frac{2\pi t}{5}\right) - 0.12\cos\left(\frac{2\pi t}{6}\right) - 0.12\sin\left(\frac{2\pi t}{6}\right) \]

Table C.10 Multiple Regression Approximation for Nighttime at Passive RRx's (Non-Conrail)

<table>
<thead>
<tr>
<th>Date</th>
<th>( \text{F(T)} )</th>
<th>( \text{Observed} )</th>
<th>Diff</th>
<th>( \text{F(T)} )</th>
<th>( \text{Observed} )</th>
<th>Diff</th>
<th>( \text{F(T)} )</th>
<th>( \text{Observed} )</th>
<th>Diff</th>
<th>Average</th>
<th>Std. Dev.</th>
<th>Max</th>
<th>Min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>4.59</td>
<td>-3.59</td>
<td></td>
<td>Jul-92</td>
<td>2.87</td>
<td>-1.17</td>
<td></td>
<td></td>
<td></td>
<td>0.00</td>
<td>1.92</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-89</td>
<td>3.71</td>
<td>1.29</td>
<td></td>
<td>Aug-92</td>
<td>3.51</td>
<td>0.49</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5.83</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>4.11</td>
<td>1.89</td>
<td></td>
<td>Sep-92</td>
<td>3.65</td>
<td>-0.65</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.59</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>3.76</td>
<td>2.24</td>
<td></td>
<td>Oct-92</td>
<td>2.99</td>
<td>2.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.01</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>3.17</td>
<td>1.83</td>
<td></td>
<td>Nov-92</td>
<td>3.76</td>
<td>-1.76</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.76</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>3.93</td>
<td>-1.93</td>
<td></td>
<td>Dec-92</td>
<td>2.69</td>
<td>-0.59</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.59</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>3.45</td>
<td>-1.45</td>
<td></td>
<td>Jan-93</td>
<td>3.82</td>
<td>-0.82</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.82</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>4.08</td>
<td>-0.08</td>
<td></td>
<td>Feb-93</td>
<td>2.95</td>
<td>2.05</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>4.22</td>
<td>-2.22</td>
<td></td>
<td>Mar-93</td>
<td>3.34</td>
<td>0.68</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.68</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>3.57</td>
<td>0.43</td>
<td></td>
<td>Apr-93</td>
<td>3.00</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>4.33</td>
<td>2.67</td>
<td></td>
<td>May-93</td>
<td>2.40</td>
<td>-0.40</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>3.27</td>
<td>3.73</td>
<td></td>
<td>Jun-93</td>
<td>3.16</td>
<td>-0.16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>4.00</td>
<td>-0.40</td>
<td></td>
<td>Jul-93</td>
<td>2.68</td>
<td>-1.68</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.68</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>3.52</td>
<td>1.48</td>
<td></td>
<td>Aug-93</td>
<td>3.31</td>
<td>-2.31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>3.82</td>
<td>-0.92</td>
<td></td>
<td>Sep-93</td>
<td>3.45</td>
<td>-1.45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.45</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>3.57</td>
<td>-0.57</td>
<td></td>
<td>Oct-93</td>
<td>2.80</td>
<td>-0.80</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.80</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>2.98</td>
<td>0.02</td>
<td></td>
<td>Nov-93</td>
<td>3.57</td>
<td>3.43</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.43</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>3.74</td>
<td>-1.74</td>
<td></td>
<td>Dec-93</td>
<td>2.50</td>
<td>2.50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>3.25</td>
<td>0.75</td>
<td></td>
<td>Jan-94</td>
<td>3.63</td>
<td>-0.63</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.63</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>3.89</td>
<td>-1.89</td>
<td></td>
<td>Feb-94</td>
<td>2.75</td>
<td>5.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5.25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>4.03</td>
<td>-1.03</td>
<td></td>
<td>Mar-94</td>
<td>3.15</td>
<td>0.85</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.85</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>3.37</td>
<td>-1.37</td>
<td></td>
<td>Apr-94</td>
<td>2.80</td>
<td>0.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>4.14</td>
<td>2.86</td>
<td></td>
<td>May-94</td>
<td>2.21</td>
<td>-1.21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.21</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>3.08</td>
<td>-1.08</td>
<td></td>
<td>Jun-94</td>
<td>2.97</td>
<td>-0.97</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-91</td>
<td>4.20</td>
<td>0.80</td>
<td></td>
<td>Jul-94</td>
<td>2.49</td>
<td>-2.49</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.49</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-91</td>
<td>3.33</td>
<td>0.67</td>
<td></td>
<td>Aug-94</td>
<td>3.12</td>
<td>-0.12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-91</td>
<td>3.72</td>
<td>-0.72</td>
<td></td>
<td>Sep-94</td>
<td>3.26</td>
<td>-2.26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-91</td>
<td>3.38</td>
<td>0.62</td>
<td></td>
<td>Oct-94</td>
<td>2.61</td>
<td>-1.61</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-91</td>
<td>2.79</td>
<td>-1.79</td>
<td></td>
<td>Nov-94</td>
<td>3.37</td>
<td>5.63</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>5.63</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-91</td>
<td>3.55</td>
<td>-2.55</td>
<td></td>
<td>Dec-94</td>
<td>2.31</td>
<td>0.69</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.69</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-91</td>
<td>3.26</td>
<td>-2.06</td>
<td></td>
<td>Jan-95</td>
<td>3.44</td>
<td>-0.44</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.44</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-91</td>
<td>3.70</td>
<td>0.30</td>
<td></td>
<td>Feb-95</td>
<td>2.56</td>
<td>0.44</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.44</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-91</td>
<td>3.84</td>
<td>0.16</td>
<td></td>
<td>Mar-95</td>
<td>2.86</td>
<td>1.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.04</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-91</td>
<td>3.61</td>
<td>-2.18</td>
<td></td>
<td>Apr-95</td>
<td>2.61</td>
<td>-0.61</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-91</td>
<td>3.95</td>
<td>3.05</td>
<td></td>
<td>May-95</td>
<td>2.02</td>
<td>1.98</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.98</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-91</td>
<td>2.88</td>
<td>5.12</td>
<td></td>
<td>Jun-95</td>
<td>2.78</td>
<td>-0.78</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.78</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-92</td>
<td>4.01</td>
<td>-0.01</td>
<td></td>
<td>Jul-95</td>
<td>2.30</td>
<td>-2.30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-92</td>
<td>3.14</td>
<td>-1.14</td>
<td></td>
<td>Aug-95</td>
<td>2.93</td>
<td>0.07</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.07</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-92</td>
<td>3.53</td>
<td>2.47</td>
<td></td>
<td>Sep-95</td>
<td>3.07</td>
<td>-2.07</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.07</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-92</td>
<td>3.19</td>
<td>-2.19</td>
<td></td>
<td>Oct-95</td>
<td>2.42</td>
<td>-0.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.42</td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-92</td>
<td>2.60</td>
<td>0.40</td>
<td></td>
<td>Nov-95</td>
<td>3.18</td>
<td>-0.16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-92</td>
<td>3.35</td>
<td>-3.35</td>
<td></td>
<td>Dec-95</td>
<td>2.12</td>
<td>1.88</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.88</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure C.7 Euler Method of the Fourier Series Representation for Daytime at Passive RRx’s (Conrail-Only). Model is Based on:

\[ F(t) = 3.23 - 0.06 \cos \left( \frac{\pi t}{6} \right) - 0.14 \cos \left( \frac{2\pi t}{6} \right) - 0.17 \cos \left( \frac{3\pi t}{6} \right) + 0.77 \cos \left( \frac{4\pi t}{6} \right) + 0.09 \cos \left( \frac{5\pi t}{6} \right) + 0.54 \cos \left( \frac{6\pi t}{6} \right) \]

\[ - 0.54 \sin \left( \frac{\pi t}{6} \right) + 0.54 \sin \left( \frac{2\pi t}{6} \right) + 0.04 \sin \left( \frac{3\pi t}{6} \right) - 0.69 \sin \left( \frac{4\pi t}{6} \right) - 0.03 \sin \left( \frac{5\pi t}{6} \right) \]

Table C.11 Euler Method for Daytime at Passive RRx’s (Conrail-Only)

| Date  | Fourier | Observed | Diff. | O-F | Date  | Fourier | Observed | Diff. | O-F | mean | std dev | max | min |
|-------|---------|----------|-------|-----|-------|---------|----------|-------|-----|-----|--------|------|-----|-----|
| Jan-89 | 1.72    | 1        | -0.72 |     | Jan-91 | 1.32    | 5        | 3.68  |     |     | 0.42   | 1.99 | 4.89 | -3.8 |
| Feb-89 | 4.23    | 3        | -1.23 |     | Feb-91 | 3.80    | 0        | -3.80 |     |     |        |      |      |     |
| Mar-89 | 2.93    | 4        | 1.07  |     | Mar-91 | 2.56    | 4        | 1.44  |     |     |        |      |      |     |
| Apr-89 | 1.70    | 3        | 1.30  |     | Apr-91 | 1.30    | 4        | 2.70  |     |     |        |      |      |     |
| May-89 | 2.15    | 1        | -1.15 |     | May-91 | 1.71    | 0        | -1.71 |     |     |        |      |      |     |
| Jun-89 | 4.42    | 3        | -1.42 |     | Jun-91 | 4.02    | 3        | -1.02 |     |     |        |      |      |     |
| Jul-89 | 2.36    | 3        | 0.64  |     | Jul-91 | 1.96    | 5        | 3.04  |     |     |        |      |      |     |
| Aug-89 | 4.64    | 1        | -3.64 |     | Aug-91 | 4.21    | 3        | -1.21 |     |     |        |      |      |     |
| Sep-89 | 4.07    | 4        | -0.07 |     | Sep-91 | 3.68    | 3        | -0.68 |     |     |        |      |      |     |
| Oct-89 | 2.86    | 6        | 3.14  |     | Oct-91 | 2.47    | 6        | 3.53  |     |     |        |      |      |     |
| Nov-89 | 2.27    | 3        | 0.73  |     | Nov-91 | 1.84    | 1        | -0.84 |     |     |        |      |      |     |
| Dec-89 | 4.06    | 4        | -0.06 |     | Dec-91 | 3.67    | 2        | -1.67 |     |     |        |      |      |     |
| Jan-90 | 1.52    | 4        | 2.48  |     | Jan-92 | 1.11    | 6        | 4.89  |     |     |        |      |      |     |
| Feb-90 | 4.02    | 2        | -2.02 |     | Feb-92 | 3.59    | 3        | -0.59 |     |     |        |      |      |     |
| Mar-90 | 2.74    | 4        | 1.26  |     | Mar-92 | 2.37    | 4        | 1.63  |     |     |        |      |      |     |
| Apr-90 | 1.50    | 3        | 1.50  |     | Apr-92 | 1.10    | 3        | 1.90  |     |     |        |      |      |     |
| May-90 | 1.93    | 2        | 0.07  |     | May-92 | 1.49    | 3        | 1.51  |     |     |        |      |      |     |
| Jun-90 | 4.22    | 3        | -1.22 |     | Jun-92 | 3.82    | 1        | -2.82 |     |     |        |      |      |     |
| Jul-90 | 2.16    | 6        | 3.84  |     | Jul-92 | 1.76    | 3        | 1.24  |     |     |        |      |      |     |
| Aug-90 | 4.42    | 4        | -0.42 |     | Aug-92 | 3.99    | 3        | -0.99 |     |     |        |      |      |     |
| Sep-90 | 3.88    | 6        | 2.12  |     | Sep-92 | 3.49    | 5        | 1.51  |     |     |        |      |      |     |
| Oct-90 | 2.67    | 2        | -0.67 |     | Oct-92 | 2.28    | 4        | 1.72  |     |     |        |      |      |     |
| Nov-90 | 2.06    | 3        | 0.94  |     | Nov-92 | 1.63    | 4        | 2.37  |     |     |        |      |      |     |
| Dec-90 | 3.86    | 2        | -1.86 |     | Dec-92 | 3.47    | 3        | -0.47 |     |     |        |      |      |     |
Figure C.8 Linear Estimation for Daytime at Passive RRx’s (Conrail-Only)

Model is Based on: $F(t) = 3.53 - 0.018t$

Table C.12 Linear Estimation for Daytime at Passive RRx’s (Conrail-Only)

<table>
<thead>
<tr>
<th>Date</th>
<th>Predicted</th>
<th>Observed</th>
<th>Diff. O-P</th>
<th>Date</th>
<th>Predicted</th>
<th>Observed</th>
<th>Diff. O-P</th>
<th>Date</th>
<th>Predicted</th>
<th>Observed</th>
<th>Diff. O-P</th>
<th>Mean</th>
<th>Std. Dev.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>3.51</td>
<td>1</td>
<td>-2.51</td>
<td>Jan-91</td>
<td>3.08</td>
<td>5</td>
<td>1.92</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.14</td>
</tr>
<tr>
<td>Feb-89</td>
<td>3.49</td>
<td>3</td>
<td>-0.49</td>
<td>Feb-91</td>
<td>3.06</td>
<td>0</td>
<td>-3.06</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>3.48</td>
<td>4</td>
<td>0.52</td>
<td>Mar-91</td>
<td>3.04</td>
<td>4</td>
<td>0.96</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>3.46</td>
<td>3</td>
<td>-0.46</td>
<td>Apr-91</td>
<td>3.03</td>
<td>4</td>
<td>0.97</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>3.44</td>
<td>1</td>
<td>-2.44</td>
<td>May-91</td>
<td>3.01</td>
<td>0</td>
<td>-3.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>3.42</td>
<td>3</td>
<td>-0.42</td>
<td>Jun-91</td>
<td>2.99</td>
<td>3</td>
<td>0.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>3.40</td>
<td>3</td>
<td>-0.40</td>
<td>Jul-91</td>
<td>2.97</td>
<td>5</td>
<td>2.03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>3.39</td>
<td>1</td>
<td>-2.39</td>
<td>Aug-91</td>
<td>2.95</td>
<td>3</td>
<td>0.05</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>3.37</td>
<td>4</td>
<td>0.63</td>
<td>Sep-91</td>
<td>2.94</td>
<td>3</td>
<td>0.06</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>3.35</td>
<td>6</td>
<td>2.65</td>
<td>Oct-91</td>
<td>2.92</td>
<td>6</td>
<td>3.08</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>3.33</td>
<td>3</td>
<td>-0.33</td>
<td>Nov-91</td>
<td>2.90</td>
<td>1</td>
<td>-1.90</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>3.31</td>
<td>4</td>
<td>0.69</td>
<td>Dec-91</td>
<td>2.88</td>
<td>2</td>
<td>-0.88</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>3.30</td>
<td>4</td>
<td>0.70</td>
<td>Jan-92</td>
<td>2.86</td>
<td>6</td>
<td>3.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>3.28</td>
<td>2</td>
<td>-1.28</td>
<td>Feb-92</td>
<td>2.85</td>
<td>3</td>
<td>0.15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>3.26</td>
<td>4</td>
<td>0.74</td>
<td>Mar-92</td>
<td>2.83</td>
<td>4</td>
<td>1.17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>3.24</td>
<td>3</td>
<td>-0.24</td>
<td>Apr-92</td>
<td>2.81</td>
<td>3</td>
<td>0.19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>3.22</td>
<td>2</td>
<td>-1.22</td>
<td>May-92</td>
<td>2.79</td>
<td>3</td>
<td>0.21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>3.21</td>
<td>3</td>
<td>-0.21</td>
<td>Jun-92</td>
<td>2.77</td>
<td>1</td>
<td>-1.77</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>3.19</td>
<td>6</td>
<td>2.81</td>
<td>Jul-92</td>
<td>2.76</td>
<td>3</td>
<td>0.24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>3.17</td>
<td>4</td>
<td>0.83</td>
<td>Aug-92</td>
<td>2.74</td>
<td>3</td>
<td>0.26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>3.15</td>
<td>6</td>
<td>2.85</td>
<td>Sep-92</td>
<td>2.72</td>
<td>5</td>
<td>2.28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>3.13</td>
<td>2</td>
<td>-1.13</td>
<td>Oct-92</td>
<td>2.70</td>
<td>4</td>
<td>1.30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>3.12</td>
<td>3</td>
<td>-0.12</td>
<td>Nov-92</td>
<td>2.68</td>
<td>4</td>
<td>1.32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>3.10</td>
<td>2</td>
<td>-1.10</td>
<td>Dec-92</td>
<td>2.67</td>
<td>3</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure C.9 Multiple Regression Approximation for Daytime at Passive RRx's (Conrail-Only).

\[
F(t) = 3.53 - 0.018t - 0.06\cos\left(\frac{pt}{6}\right) + 0.09\cos\left(\frac{2pt}{6}\right) + 0.2\cos\left(\frac{3pt}{6}\right) + 0.47\cos\left(\frac{4pt}{6}\right) + 0.45\cos\left(\frac{5pt}{6}\right)
- 0.22\sin\left(\frac{pt}{6}\right) - 0.51\sin\left(\frac{2pt}{6}\right)
\]

Table C.13 Multiple Regression Approximation for Daytime at Passive RRx's (Conrail-Only)

<table>
<thead>
<tr>
<th>Date</th>
<th>F(T)</th>
<th>Observed</th>
<th>Diff. O-F(T)</th>
<th>Date</th>
<th>F(T)</th>
<th>Observed</th>
<th>Diff. O-F(T)</th>
<th>average</th>
<th>std. dev</th>
<th>max</th>
<th>min</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>2.33</td>
<td>1</td>
<td>-1.33</td>
<td>Jan-91</td>
<td>1.89</td>
<td>5</td>
<td>3.11</td>
<td>0.14</td>
<td>1.66</td>
<td>4.32</td>
<td>-3.60</td>
<td>0.388</td>
</tr>
<tr>
<td>Feb-89</td>
<td>2.58</td>
<td>3</td>
<td>0.42</td>
<td>Feb-91</td>
<td>2.14</td>
<td>0</td>
<td>-2.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>3.64</td>
<td>4</td>
<td>0.36</td>
<td>Mar-91</td>
<td>3.21</td>
<td>4</td>
<td>0.79</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>3.44</td>
<td>3</td>
<td>-0.44</td>
<td>Apr-91</td>
<td>3.01</td>
<td>4</td>
<td>0.99</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>4.03</td>
<td>1</td>
<td>-3.03</td>
<td>May-91</td>
<td>3.60</td>
<td>0</td>
<td>-3.60</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>3.39</td>
<td>3</td>
<td>-0.39</td>
<td>Jun-91</td>
<td>2.96</td>
<td>3</td>
<td>0.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>3.31</td>
<td>3</td>
<td>-0.31</td>
<td>Jul-91</td>
<td>2.88</td>
<td>5</td>
<td>2.12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>2.84</td>
<td>1</td>
<td>-1.84</td>
<td>Aug-91</td>
<td>2.41</td>
<td>3</td>
<td>0.59</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>3.96</td>
<td>4</td>
<td>0.04</td>
<td>Sep-91</td>
<td>3.53</td>
<td>3</td>
<td>-0.53</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>3.71</td>
<td>6</td>
<td>2.29</td>
<td>Oct-91</td>
<td>3.28</td>
<td>6</td>
<td>2.72</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>3.26</td>
<td>3</td>
<td>-0.26</td>
<td>Nov-91</td>
<td>2.83</td>
<td>1</td>
<td>-1.83</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>4.47</td>
<td>4</td>
<td>-0.47</td>
<td>Dec-91</td>
<td>4.04</td>
<td>2</td>
<td>-2.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>2.11</td>
<td>4</td>
<td>1.89</td>
<td>Jan-92</td>
<td>1.68</td>
<td>6</td>
<td>4.32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>2.36</td>
<td>2</td>
<td>-0.36</td>
<td>Feb-92</td>
<td>1.93</td>
<td>3</td>
<td>1.07</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>3.42</td>
<td>4</td>
<td>0.58</td>
<td>Mar-92</td>
<td>2.99</td>
<td>4</td>
<td>1.01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>3.23</td>
<td>3</td>
<td>-0.23</td>
<td>Apr-92</td>
<td>2.79</td>
<td>3</td>
<td>0.21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>3.82</td>
<td>2</td>
<td>-1.82</td>
<td>May-92</td>
<td>3.39</td>
<td>3</td>
<td>-0.39</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>3.17</td>
<td>3</td>
<td>-0.17</td>
<td>Jun-92</td>
<td>2.74</td>
<td>1</td>
<td>-1.74</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>3.09</td>
<td>6</td>
<td>2.91</td>
<td>Jul-92</td>
<td>2.66</td>
<td>3</td>
<td>0.34</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>2.62</td>
<td>4</td>
<td>1.38</td>
<td>Aug-92</td>
<td>2.19</td>
<td>3</td>
<td>0.81</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>3.75</td>
<td>6</td>
<td>2.25</td>
<td>Sep-92</td>
<td>3.31</td>
<td>5</td>
<td>1.69</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>3.49</td>
<td>2</td>
<td>-1.49</td>
<td>Oct-92</td>
<td>3.06</td>
<td>4</td>
<td>0.94</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>3.05</td>
<td>3</td>
<td>-0.05</td>
<td>Nov-92</td>
<td>2.62</td>
<td>4</td>
<td>1.38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>4.25</td>
<td>2</td>
<td>-2.25</td>
<td>Dec-92</td>
<td>3.82</td>
<td>3</td>
<td>-0.82</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure C.10 Euler Method of the Fourier Series Representation for Nighttime at Passive RRx’s (Conrail-Only). Model is Based on:

\[
F(t) = 2.61 + 0.59 \cos\left(\frac{\pi t}{6}\right) - 0.29 \cos\left(\frac{2\pi t}{6}\right) + 0.004 \cos\left(\frac{3\pi t}{6}\right) + 0.21 \cos\left(\frac{4\pi t}{6}\right) + 0.16 \cos\left(\frac{5\pi t}{6}\right) + 0.96 \cos\left(\frac{6\pi t}{6}\right) - 0.43 \sin\left(\frac{\pi t}{6}\right) - 0.79 \sin\left(\frac{2\pi t}{6}\right) + 0.21 \sin\left(\frac{3\pi t}{6}\right) - 0.07 \sin\left(\frac{4\pi t}{6}\right) + 0.01 \sin\left(\frac{5\pi t}{6}\right)
\]

<table>
<thead>
<tr>
<th>Date</th>
<th>Fourier</th>
<th>Observed</th>
<th>Diff. O-F</th>
<th>Date</th>
<th>Fourier</th>
<th>Observed</th>
<th>Diff. O-F</th>
<th>mean</th>
<th>std dev</th>
<th>max</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>1.01</td>
<td>2</td>
<td>0.99</td>
<td>Jan-91</td>
<td>0.61</td>
<td>5</td>
<td>4.39</td>
<td>0.41</td>
<td>2.31</td>
<td>6.19</td>
<td>-3.85</td>
</tr>
<tr>
<td>Feb-89</td>
<td>2.94</td>
<td>2</td>
<td>-0.94</td>
<td>Feb-91</td>
<td>2.53</td>
<td>2</td>
<td>-0.53</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>1.47</td>
<td>5</td>
<td>3.53</td>
<td>Mar-91</td>
<td>1.07</td>
<td>3</td>
<td>1.93</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>3.41</td>
<td>2</td>
<td>-1.41</td>
<td>Apr-91</td>
<td>3.00</td>
<td>0</td>
<td>-3.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>1.69</td>
<td>5</td>
<td>3.31</td>
<td>May-91</td>
<td>1.28</td>
<td>2</td>
<td>0.72</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>2.63</td>
<td>2</td>
<td>-0.63</td>
<td>Jun-91</td>
<td>2.23</td>
<td>4</td>
<td>1.77</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>0.15</td>
<td>5</td>
<td>4.85</td>
<td>Jul-91</td>
<td>-0.26</td>
<td>2</td>
<td>2.26</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>2.85</td>
<td>1</td>
<td>-1.85</td>
<td>Aug-91</td>
<td>2.42</td>
<td>0</td>
<td>-2.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>2.61</td>
<td>4</td>
<td>1.39</td>
<td>Sep-91</td>
<td>2.19</td>
<td>3</td>
<td>0.81</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>4.82</td>
<td>7</td>
<td>2.18</td>
<td>Oct-91</td>
<td>4.42</td>
<td>3</td>
<td>-1.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>2.31</td>
<td>4</td>
<td>1.69</td>
<td>Nov-91</td>
<td>1.91</td>
<td>8</td>
<td>6.09</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>4.05</td>
<td>3</td>
<td>-1.05</td>
<td>Dec-91</td>
<td>3.65</td>
<td>5</td>
<td>1.35</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>0.81</td>
<td>7</td>
<td>6.19</td>
<td>Jan-92</td>
<td>0.42</td>
<td>1</td>
<td>0.58</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>2.74</td>
<td>1</td>
<td>-1.74</td>
<td>Feb-92</td>
<td>2.32</td>
<td>1</td>
<td>-1.32</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>1.27</td>
<td>0</td>
<td>-1.27</td>
<td>Mar-92</td>
<td>0.86</td>
<td>2</td>
<td>1.14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>3.21</td>
<td>2</td>
<td>-1.21</td>
<td>Apr-92</td>
<td>2.79</td>
<td>4</td>
<td>1.21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>1.49</td>
<td>4</td>
<td>2.51</td>
<td>May-92</td>
<td>1.08</td>
<td>1</td>
<td>-0.08</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>2.43</td>
<td>2</td>
<td>-0.43</td>
<td>Jun-92</td>
<td>2.04</td>
<td>1</td>
<td>-1.04</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>-0.05</td>
<td>0</td>
<td>0.05</td>
<td>Jul-92</td>
<td>-0.46</td>
<td>2</td>
<td>2.46</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>2.63</td>
<td>1</td>
<td>-1.63</td>
<td>Aug-92</td>
<td>2.21</td>
<td>1</td>
<td>-1.21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>2.40</td>
<td>1</td>
<td>-1.40</td>
<td>Sep-92</td>
<td>1.98</td>
<td>2</td>
<td>0.02</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>2.11</td>
<td>4</td>
<td>1.89</td>
<td>Nov-92</td>
<td>1.71</td>
<td>2</td>
<td>0.29</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>3.85</td>
<td>0</td>
<td>-3.85</td>
<td>Dec-92</td>
<td>3.45</td>
<td>4</td>
<td>0.55</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure C.11 Linear Estimation for Nighttime at Passive RRx’s (Conrail-Only).
Model is Based on: $F(t) = 3.185 - 0.024t$

Table C.15 Linear Estimation for Nighttime at Passive RRx’s (Conrail-Only)

<table>
<thead>
<tr>
<th>Date</th>
<th>Predicted</th>
<th>Observed</th>
<th>Diff. O-P</th>
<th>Date</th>
<th>Predicted</th>
<th>Observed</th>
<th>Diff. O-P</th>
<th>mean</th>
<th>std. dev.</th>
<th>max</th>
<th>min</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>3.16</td>
<td>2</td>
<td>-1.16</td>
<td>Jan-91</td>
<td>2.59</td>
<td>5</td>
<td>2.42</td>
<td>0.01</td>
<td>1.9</td>
<td>5.66</td>
<td>-2.83</td>
</tr>
<tr>
<td>Feb-89</td>
<td>3.14</td>
<td>2</td>
<td>-1.14</td>
<td>Feb-91</td>
<td>2.56</td>
<td>2</td>
<td>-0.56</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>3.11</td>
<td>5</td>
<td>1.89</td>
<td>Mar-91</td>
<td>2.54</td>
<td>3</td>
<td>0.46</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>3.09</td>
<td>2</td>
<td>-1.09</td>
<td>Apr-91</td>
<td>2.51</td>
<td>0</td>
<td>-2.51</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>3.07</td>
<td>5</td>
<td>1.94</td>
<td>May-91</td>
<td>2.49</td>
<td>2</td>
<td>-0.49</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>3.04</td>
<td>2</td>
<td>-1.04</td>
<td>Jun-91</td>
<td>2.47</td>
<td>4</td>
<td>1.54</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>3.02</td>
<td>5</td>
<td>1.98</td>
<td>Jul-91</td>
<td>2.44</td>
<td>2</td>
<td>-0.44</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>2.99</td>
<td>1</td>
<td>-1.99</td>
<td>Aug-91</td>
<td>2.42</td>
<td>0</td>
<td>-2.42</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>2.97</td>
<td>4</td>
<td>1.03</td>
<td>Sep-91</td>
<td>2.39</td>
<td>3</td>
<td>0.61</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>2.95</td>
<td>7</td>
<td>4.06</td>
<td>Oct-91</td>
<td>2.37</td>
<td>3</td>
<td>0.63</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>2.92</td>
<td>4</td>
<td>1.08</td>
<td>Nov-91</td>
<td>2.35</td>
<td>8</td>
<td>5.66</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>2.90</td>
<td>3</td>
<td>0.10</td>
<td>Dec-91</td>
<td>2.32</td>
<td>5</td>
<td>2.68</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>2.87</td>
<td>7</td>
<td>4.13</td>
<td>Jan-92</td>
<td>2.30</td>
<td>1</td>
<td>-1.30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>2.85</td>
<td>1</td>
<td>-1.85</td>
<td>Feb-92</td>
<td>2.27</td>
<td>1</td>
<td>-1.27</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>2.83</td>
<td>0</td>
<td>-2.83</td>
<td>Mar-92</td>
<td>2.25</td>
<td>2</td>
<td>-0.25</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>2.80</td>
<td>2</td>
<td>-0.80</td>
<td>Apr-92</td>
<td>2.23</td>
<td>4</td>
<td>1.78</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>2.78</td>
<td>4</td>
<td>1.22</td>
<td>May-92</td>
<td>2.20</td>
<td>1</td>
<td>-1.20</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>2.75</td>
<td>2</td>
<td>-0.75</td>
<td>Jun-92</td>
<td>2.18</td>
<td>1</td>
<td>-1.18</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>2.73</td>
<td>0</td>
<td>-2.73</td>
<td>Jul-92</td>
<td>2.15</td>
<td>2</td>
<td>-0.15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>2.71</td>
<td>1</td>
<td>-1.71</td>
<td>Aug-92</td>
<td>2.13</td>
<td>1</td>
<td>-1.13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>2.68</td>
<td>1</td>
<td>-1.68</td>
<td>Sep-92</td>
<td>2.11</td>
<td>2</td>
<td>-0.11</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>2.66</td>
<td>1</td>
<td>-1.66</td>
<td>Oct-92</td>
<td>2.08</td>
<td>2</td>
<td>-0.08</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>2.63</td>
<td>4</td>
<td>1.37</td>
<td>Nov-92</td>
<td>2.06</td>
<td>2</td>
<td>-0.06</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>2.61</td>
<td>0</td>
<td>-2.61</td>
<td>Dec-92</td>
<td>2.03</td>
<td>4</td>
<td>1.97</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure C.12 Multiple Regression Approximation for Nighttime at Passive RRx's (Conrail-Only).

\[
F(t) = 3.19 - 0.024t + 0.29\cos\left(\frac{\pi t}{6}\right) - 0.9\cos\left(\frac{2\pi t}{6}\right) + 0.14\cos\left(\frac{3\pi t}{6}\right) + 0.81\cos\left(\frac{4\pi t}{6}\right) \\
+ 0.36\cos\left(\frac{5\pi t}{6}\right) - 0.27\cos\left(\frac{6\pi t}{6}\right) + 0.27\sin\left(\frac{2\pi t}{6}\right)
\]

Table C.16 Multiple Regression Approximation for Nighttime at Passive RRx's (Conrail-Only)

<table>
<thead>
<tr>
<th>Date</th>
<th>F(T)</th>
<th>Observed</th>
<th>Diff. O-F(T)</th>
<th>Date</th>
<th>F(T)</th>
<th>Observed</th>
<th>Diff. O-F(T)</th>
<th>mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan-89</td>
<td>2.75</td>
<td>2</td>
<td>-0.75</td>
<td>Jan-91</td>
<td>2.17</td>
<td>5</td>
<td>2.83</td>
<td>0.01</td>
</tr>
<tr>
<td>Feb-89</td>
<td>3.34</td>
<td>2</td>
<td>-1.34</td>
<td>Feb-91</td>
<td>2.76</td>
<td>2</td>
<td>-0.76</td>
<td></td>
</tr>
<tr>
<td>Mar-89</td>
<td>5.09</td>
<td>5</td>
<td>-0.09</td>
<td>Mar-91</td>
<td>4.51</td>
<td>3</td>
<td>-1.51</td>
<td></td>
</tr>
<tr>
<td>Apr-89</td>
<td>2.46</td>
<td>2</td>
<td>-0.46</td>
<td>Apr-91</td>
<td>1.88</td>
<td>0</td>
<td>-1.88</td>
<td></td>
</tr>
<tr>
<td>May-89</td>
<td>2.30</td>
<td>5</td>
<td>2.70</td>
<td>May-91</td>
<td>1.73</td>
<td>2</td>
<td>0.27</td>
<td></td>
</tr>
<tr>
<td>Jun-89</td>
<td>1.89</td>
<td>2</td>
<td>0.11</td>
<td>Jun-91</td>
<td>1.32</td>
<td>4</td>
<td>2.68</td>
<td></td>
</tr>
<tr>
<td>Jul-89</td>
<td>2.72</td>
<td>5</td>
<td>2.28</td>
<td>Jul-91</td>
<td>2.14</td>
<td>2</td>
<td>-0.14</td>
<td></td>
</tr>
<tr>
<td>Aug-89</td>
<td>2.83</td>
<td>1</td>
<td>-1.83</td>
<td>Aug-91</td>
<td>2.25</td>
<td>0</td>
<td>-2.25</td>
<td></td>
</tr>
<tr>
<td>Sep-89</td>
<td>4.94</td>
<td>4</td>
<td>-0.94</td>
<td>Sep-91</td>
<td>4.37</td>
<td>3</td>
<td>-1.37</td>
<td></td>
</tr>
<tr>
<td>Oct-89</td>
<td>2.68</td>
<td>7</td>
<td>4.32</td>
<td>Oct-91</td>
<td>2.10</td>
<td>3</td>
<td>0.90</td>
<td></td>
</tr>
<tr>
<td>Nov-89</td>
<td>2.04</td>
<td>4</td>
<td>1.96</td>
<td>Nov-91</td>
<td>1.46</td>
<td>8</td>
<td>6.54</td>
<td></td>
</tr>
<tr>
<td>Dec-89</td>
<td>3.32</td>
<td>3</td>
<td>-0.32</td>
<td>Dec-91</td>
<td>2.75</td>
<td>5</td>
<td>2.25</td>
<td></td>
</tr>
<tr>
<td>Jan-90</td>
<td>2.46</td>
<td>7</td>
<td>4.54</td>
<td>Jan-92</td>
<td>1.88</td>
<td>1</td>
<td>-0.88</td>
<td></td>
</tr>
<tr>
<td>Feb-90</td>
<td>3.05</td>
<td>1</td>
<td>-2.05</td>
<td>Feb-92</td>
<td>2.47</td>
<td>1</td>
<td>-1.47</td>
<td></td>
</tr>
<tr>
<td>Mar-90</td>
<td>4.80</td>
<td>0</td>
<td>-4.80</td>
<td>Mar-92</td>
<td>4.22</td>
<td>2</td>
<td>-2.22</td>
<td></td>
</tr>
<tr>
<td>Apr-90</td>
<td>2.17</td>
<td>2</td>
<td>-0.17</td>
<td>Apr-92</td>
<td>1.59</td>
<td>4</td>
<td>2.41</td>
<td></td>
</tr>
<tr>
<td>May-90</td>
<td>2.01</td>
<td>4</td>
<td>1.99</td>
<td>May-92</td>
<td>1.44</td>
<td>1</td>
<td>-0.44</td>
<td></td>
</tr>
<tr>
<td>Jun-90</td>
<td>1.61</td>
<td>2</td>
<td>0.39</td>
<td>Jun-92</td>
<td>1.03</td>
<td>1</td>
<td>-0.03</td>
<td></td>
</tr>
<tr>
<td>Jul-90</td>
<td>2.43</td>
<td>0</td>
<td>-2.43</td>
<td>Jul-92</td>
<td>1.86</td>
<td>2</td>
<td>0.14</td>
<td></td>
</tr>
<tr>
<td>Aug-90</td>
<td>2.54</td>
<td>1</td>
<td>-1.54</td>
<td>Aug-92</td>
<td>1.96</td>
<td>1</td>
<td>-0.96</td>
<td></td>
</tr>
<tr>
<td>Sep-90</td>
<td>4.65</td>
<td>1</td>
<td>-3.65</td>
<td>Sep-92</td>
<td>4.08</td>
<td>2</td>
<td>-2.08</td>
<td></td>
</tr>
<tr>
<td>Oct-90</td>
<td>2.39</td>
<td>1</td>
<td>-1.39</td>
<td>Oct-92</td>
<td>1.81</td>
<td>2</td>
<td>0.19</td>
<td></td>
</tr>
<tr>
<td>Nov-90</td>
<td>1.75</td>
<td>4</td>
<td>2.25</td>
<td>Nov-92</td>
<td>1.18</td>
<td>2</td>
<td>0.82</td>
<td></td>
</tr>
<tr>
<td>Dec-90</td>
<td>3.04</td>
<td>0</td>
<td>-3.04</td>
<td>Dec-92</td>
<td>2.46</td>
<td>4</td>
<td>1.54</td>
<td></td>
</tr>
</tbody>
</table>
Appendix D

TSIE Source Code
// TSIE.h : main header file for the TSIE application
#ifndef AFXWIN_H
#endif
#include "resource.h"    // main symbols

// See TSIE.cpp for the implementation of this class

class CTSIEApp : public CWinApp
{
public:
    CTSIEApp();

    // Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CTSIEApp)
    public:
        virtual BOOL InitInstance();
    //}}AFX_VIRTUAL

    // Implementation
    //{{AFX_MSG(CTSIEApp)
    afx_msg void OnAppAbout();
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};
// TSIE.cpp : Defines the class behaviors for the application.

#include "stdafx.h"
#include "TSIE.h"
#include "MainFrm.h"
#include "ChildFrm.h"
#include "TSIEDoc.h"
#include "SpreadView.h"
ifdef _DEBUG
#define new DEBUG_NEW
#endif
static char THIS_FILE[] = __FILE__;
#endif

// CTSIEApp
BEGIN_MESSAGE_MAP(CTSIEApp, CWinApp)
    // Standard file based document commands
    ON_COMMAND(ID_FILE_NEW, CWinApp::OnFileNew)
    ON_COMMAND(ID_FILE_OPEN, CWinApp::OnFileOpen)
    // Standard print setup command
    ON_COMMAND(ID_FILE_PRINT_SETUP, CWinApp::OnFilePrintSetup)
END_MESSAGE_MAP()

// CTSIEApp construction
CTSIEApp::CTSIEApp()
{
}
}

BOOL CTSIEApp::InitInstance()
{
   AfxEnableControlContainer();

   // Standard initialization

default
   Enable3dControls(); // Call this when using MFC in a shared DLL
#else
   Enable3dControlsStatic(); // Call this when linking to MFC statically
#endif
LoadStdProfileSettings(); // Load standard INI file options
CMultiDocTemplate* pDocTemplate = new CMultiDocTemplate{
   IDR_TSIETYPE,
   RUNTIME_CLASS(CTSIEDoc),
   RUNTIME_CLASS(CSplitterFrame), // custom MDI child frame
   RUNTIME_CLASS(CSpreadView));
AddDocTemplate(pDocTemplate);
// create main MDI Frame window
CMainFrame* pMainFrame = new CMainFrame;
if (!pMainFrame->LoadFrame(IDR_MAINFRAME))
    return FALSE;

m_pMainWnd = pMainFrame;
CCommandLineInfo cmdInfo;
ParseCommandLine(cmdInfo);
if (!ProcessShellCommand(cmdInfo))
    return FALSE;

pMainFrame->ShowWindow(m_nCmdShow);
pMainFrame->UpdateWindow();
return TRUE;
}

class CActiveDlg : public CDialog
{
public:
    CActiveDlg();
    ////{{AFX_DATA CActiveDlg
    enum { IDD = IDD_ABOUTBOX }
    ////}}AFX_DATA
    CActiveDlg() : CActiveDlg::IDD)
    {
    ////{{AFX_DATA_INIT CActiveDlg
    ////}}AFX_DATA_INIT
    CActiveDlg::CAboutDlg()
    {
        CActiveDlg aboutDlg;
        aboutDlg.DoModal();
    }

    CActiveDlg::CAboutDlg() : CActiveDlg(CActiveDlg::IDD)
    {
        CActiveDlg aboutDlg;
        CActiveDlg DoModal;
    }

    CActiveDlg::DoDataExchange(CDataExchange* pDX)
    {
        CActiveDlg::CActiveDlg
    }

    CActiveDlg::OnAppAbout()
    {
        CActiveDlg aboutDlg;
        aboutDlg.DoModal();
    }
// MainFrm.h : interface of the CMainFrame class

class CMainFrame : public CMDIFrameWnd
{
   DECLARE_DYNAMIC(CMainFrame)
public:
   CMainFrame();

   // Attributes
public:

   // Operations
public:

   // Overrides
   // ClassWizard generated virtual function overrides
   //{{AFX_VIRTUAL(CMainFrame)
   virtual BOOL PreCreateWindow(CREATESTRUCT& cs);
   //}}AFX_VIRTUAL

   // Implementation
public:
   virtual ~CMainFrame();
   
   #ifdef _DEBUG
   virtual void AssertValid() const;
   virtual void Dump(CDumpContext& dc) const;
   #endif

   protected: // control bar embedded members
   CStatusBar m_wndStatusBar;
   CToolBar m_wndToolBar;

   // Generated message map functions
protected:
   //{{AFX_MSG(CMainFrame)
   afx_msg int OnCreate(LPCREATESTRUCT lpCreateStruct);
   //}}AFX_MSG
   DECLARE_MESSAGE_MAP()
};
MainFrm.cpp : implementation of the CMainFrame class

#include "stdafx.h"
#include "TSIE.h"
#include "MainFrm.h"

#ifndef _DEBUG
#define new DEBUG_NEW
#endif

static char THIS_FILE[] = __FILE__;

BEGIN_MESSAGE_MAP(CMainFrame, CMDIFrameWnd)
ON_WM_CREATE()
//}}AFX_MSG_MAP
// Global help commands
ON_COMMAND(ID_HELP_FINDER, CMDIFrameWnd::OnHelpFinder)
ON_COMMAND(ID_HELP, CMDIFrameWnd::OnHelp)
ON_COMMAND(ID_CONTEXT_HELP, CMDIFrameWnd::OnContextHelp)
ON_COMMAND(ID_DEFAULT_HELP, CMDIFrameWnd::OnHelpFinder)
END_MESSAGE_MAP()

static UINT indicators[] =
{
   ID_SEPARATOR,      // status line indicator
   ID_INDICATOR_CAPS,
   ID_INDICATOR_NUM,
   ID_INDICATOR_SCRL,
};

CMainFrame::CMainFrame()
{
}

CMainFrame::~CMainFrame()
{
}

int CMainFrame::OnCreate(LPCREATESTRUCT lpCreateStruct)
{
   if (CMDIFrameWnd::OnCreate(lpCreateStruct) == -1)
      return -1;
   if (!m_wndToolBar.Create(this) ||
       !m_wndToolBar.LoadToolBar(IDR_MAINFRAME))
   { TRACE0("Failed to create toolbar\n");
      return -1; 
   } return 0;
return -1;    // fail to create
}

if (!m_wndStatusBar.Create(this) ||
    !m_wndStatusBar.SetIndicators(indicators,
            sizeof(indicators)/sizeof(UINT)) )
{
    TRACEO("Failed to create status bar\n");
    return -1;    // fail to create
}

m_wndToolBar.SetBarStyle(m_wndToolBar.GetBarStyle () |
            CBRS_TOOLTIPS | CBRS_FLYBY | CBRS_SIZE_DYNAMIC);

m_wndToolBar.EnableDocking(CBRS_ALIGN_ANY);

EnableDocking(CBRS_ALIGN_ANY);
DockControlBar(&m_wndToolBar);
return 0;
}

BOOL CMainFrame::PreCreateWindow(CREATESTRUCT& cs)
{
    return CMDIFrameWnd::PreCreateWindow(cs);
}

#endif

// CMainFrame diagnostics
#endif

void CMainFrame::AssertValid() const
{
    CMDIFrameWnd::AssertValid();
}

void CMainFrame::Dump(CDumpContext& dc) const
{
    CMDIFrameWnd::Dump(dc);
}

#endif // _DEBUG
// ChildFrm.h : interface of the CChildFrame class

class CChildFrame : public CMDIChildWnd
{
    DECLARE_DYNCREATE(CChildFrame)
public:
    CChildFrame();

    Operations
public:

    Overrides
    // ClassWizard generated virtual function overrides
    //{{AFX_VIRTUAL(CChildFrame)
    virtual BOOL PreCreateWindow(CREATESTRUCT& cs);
    //}}AFX_VIRTUAL

    // Implementation
public:
    virtual ~CChildFrame();
    #ifdef DEBUG
    virtual void AssertValid() const;
    virtual void Dump(CDumpContext& dc) const;
    #endif

    // Generated message map functions
protected:
    //{{AFX_MSG(CChildFrame)
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
};

CSplitterFrame frame

class CSplitterFrame : public CMDIChildWnd
{
    DECLARE_DYNCREATE(CSplitterFrame)
protected:
    CSplitterFrame(); // protected constructor used by dynamic creation

    // Attributes
protected:
    CSplitterWnd m_wndSplitter;

    // Operations
public:

protected:
    virtual BOOL OnCreateClient(LPCREATESTRUCT lpcs, CCreateContext* pContext);
    //{{AFX_VIRTUAL
    protected:
    virtual ~CSplitterFrame();
    DECLARE_MESSAGE_MAP()
};
// ChildFrm.cpp : implementation of the CChildFrame class

#include "stdafx.h"
#include "TSIE.h"
#include "ChildFrm.h"
#include "SpreadView.h"
#include "GraphView.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

IMPLEMENT_DYNCREATE(CChildFrame, CMDIChildWnd)
BEGIN_MESSAGE_MAP(CChildFrame, CMDIChildWnd)
//}}AFX_MSG_MAP(CChildFrame)
END_MESSAGE_MAP()

CChildFrame::CChildFrame()
{
}

CChildFrame::~CChildFrame()
{
}

BOOL CChildFrame::PreCreateWindow(CREATESTRUCT& cs)
{
    return CMDIChildWnd::PreCreateWindow(cs);
}

// CChildFrame diagnostics
#ifdef _DEBUG
void CChildFrame::AssertValid() const
{
    CMDIChildWnd::AssertValid();
}

void CChildFrame::Dump(CDumpContext& dc) const
{
    CMDIChildWnd::Dump(dc);
}
#endif // _DEBUG
IMPLEMENT_DYNCREATE(CSplitterFrame, CMDIChildWnd)

CSplitterFrame::CSplitterFrame()
{
}

CSplitterFrame::~CSplitterFrame()
{
}

BEGIN_MESSAGE_MAP(CSplitterFrame, CMDIChildWnd)

//}}AFX_MSG_MAP

END_MESSAGE_MAP()

BOOL CSplitterFrame::OnCreateClient(LPCREATESTRUCT lpcs,
CCreateContext* pContext)
{
    CRect rect;
    GetClientRect(&rect);
    CSize size = rect.Size();
    size.cy /= 2;

    if (!m_wndSplitter.CreateStatic(this, 2, 1))
    {
        TRACE0("Failed to CreateStaticSplitter\n");
        return FALSE;
    }

    if (!m_wndSplitter.CreateView(1, 0,
RUNTIME_CLASS(CSpreadView), size, pContext))
    {
        TRACE0("Failed to create first pane\n");
        return FALSE;
    }

    if (!m_wndSplitter.CreateView(0, 0,
RUNTIME_CLASS(CGraphView), size, pContext))
    {
        TRACE0("Failed to create second pane\n");
        return FALSE;
    }

    // activate the spreadsheet view
   SetActiveView((CSpreadView*)m_wndSplitter.GetPane(1,0));
    return TRUE;
}
class CTSIEDoc : public CDocument
{
protected: // create from serialization only
    CTSIEDoc();
    DECLARE_DYNCREATE(CTSIEDoc)

// Attributes
public:

// Operations
public:

// Overrides
// ClassWizard generated virtual function overrides
//}}AFX_VIRTUAL
public:
    virtual void Serialize(CArchive& ar);
    virtual BOOL OnOpenDocument(LPCTSTR lpszPathName);
    virtual BOOL OnSaveDocument(LPCTSTR lpszPathName);
//}}AFX_VIRTUAL

// Implementation
public:
    virtual ~CTSIEDoc();
    #ifdef _DEBUG
        virtual void AssertValid() const;
        virtual void Dump(CDumpContext& dc) const;
    #endif

protected:

// Generated message map functions
protected:
    //{{AFX_MSG(CTSIEDoc)
    //}}AFX_MSG
    DECLARE_MESSAGE_MAP()
implementation of the CTSIEDoc class

#include "view.h"
#include "sheet.h"

DECLARE_STANDARD_MFC
S_FILE[] = __FILE__;

P(0)

:EDoc()

CTSIEDoc::~CTSIEDoc()
{
}

void CTSIEDoc::Serialize(CArchive& ar)
{
    if (ar.IsStoring())
    {
    }
    else
    {
    }

    #ifdef __DEBUG
    void CTSIEDoc::AssertValid() const
    {
        CDocument::AssertValid();
    }

    void CTSIEDoc::Dump(CDumpContext& dc) const
    {
        CDocument::Dump(dc);
    }
    #endif //__DEBUG

BOOL CTSIEDoc::OnOpenDocument(LPCTSTR lpszPathName)
{
    if (!CDocument::OnOpenDocument(lpszPathName))
        return FALSE;
    short* pFileType = &a;
    POSITION pos = GetFirstViewPosition();
    CSpreadView* pView = (CSpreadView*) GetNextView(pos);
    pView->m_ss.Read(lpszPathName, pFileType);

    return TRUE;
}
// SpreadView.h : header file
#
#include "vcfl.h"

///
/// CSpreadView view
class CGraphView;
class CSpreadView : public CView
{
protected:
    CSpreadView();  //protected constructor used by dynamic creation
    DECLARE_DYNCREATE(CSpreadView)

// Attributes
public:
    CVCFL m_ss;
    CGraphView* pChartView;
    BOOL m_gridline;
    BOOL m_ColHeading;
    BOOL m_RowHeading;
    BOOL m_EditBar;
    BOOL m_CellProtection;

// Operations
public:
    void ClearSheet();

// Overrides
// ClassWizard generated virtual function overrides
//{{AFX_VIRTUAL(CSpreadView)
public:

    virtual BOOL Create(LPCTSTR lpszClassName, LPCTSTR lpszWindowName, DWORD dwStyle, const RECT& rect, CWnd* pParentWnd, UINT nID, CCreateContext* pContext = NULL);
    virtual void OnInitialUpdate();
    virtual void OnDraw(CDC* pDC);  // overridden to draw this view
//}}AFX_VIRTUAL

// Implementation
protected:
    virtual ~CSpreadView();
    #ifdef _DEBUG
        virtual void AssertValid() const;
        virtual void Dump(CDumpContext& dc) const;
    #endif

    // Generated message map functions
protected:
//{{AFX_MSG(CSpreadView)
afx_msg void OnSize(UINT nType, int cx, int cy);
afx_msg void OnViewGridlines();
afx_msg void OnUpdateViewGridlines(CCmdUI* pCmdUI);
afx_msg void OnViewColHeading();
afx_msg void OnUpdateViewColHeading(CCmdUI* pCmdUI);
afx_msg void OnViewRowHeading();
afx_msg void OnUpdateViewRowHeading(CCmdUI* pCmdUI);
afx_msg void OnViewEditbar();
//}}AFX_MSG
}
afx_msg void OnUpdateViewEditbar(CCmdUI* pCmdUI);
afx_msg void OnFormatAlignment();
afx_msg void OnFormatFont();
afx_msg void OnFormatBorder();
afx_msg void OnFormatPattern();
afx_msg void OnFormatNumber();
afx_msg void OnFormatProtection();
afx_msg void OnUpdateFormatProtection(CCmdUI* pCmdUI);
afx_msg void OnFormatColWidth();
afx_msg void OnFormatRowHeight();
afx_msg void OnEditCut();
afx_msg void OnEditCopy();
afx_msg void OnEditPaste();
afx_msg void OnUpdateEditPaste(CCmdUI* pCmdUI);
afx_msg void OnEditClear();
afx_msg void OnEditInsColrow();
afx_msg void OnEditDelColrow();
afx_msg void OnEditInsertSheet();
afx_msg void OnEditDeleteSheet();
afx_msg void OnEditSort();
afx_msg void OnFilePrint();
afx_msg void OnFilePrintSetup();
afx_msg void OnAnalysisSmoving();
afx_msg void OnAnalysisDecompo();
afx_msg void OnAnalysisCusum();
afx_msg void OnAnalysisSexpo();
afx_msg void OnAnalysisHexpo();
afx_msg void OnAnalysisWexpo();
afx_msg void OnAnalysisIdentifier();
afx_msg void OnBtCurrency();
afx_msg void OnBtGeneral();
afx_msg void OnBtPercent();
afx_msg void OnBtScientific();
afx_msg void OnAnalysisRun();
afx_msg void OnAnalysisDeseasonal();
afx_msg void OnAnalysisDetrend();
afx_msg void OnAnalysisRfh();
//}AFX_MSG
DECLARE_MESSAGE_MAP()
// SpreadView.cpp : implementation file
//

#include "stdafx.h"
#include "TSIE.h"
#include "SpreadView.h"
#include "GraphView.h"
#include "AnalysisDlg.h"
#include <math.h>

#ifdef _DEBUG
#define new DEBUG_NEW
#endif

static char * THIS_FILE[] = __FILE__;
#endif

const int rows = 200;
const int cols = 10;

DECLARE DYNCREATE(CSpreadView, CView)

CSpreadView::CSpreadView()
{
    m_gridline = true;
    m_COLHeading = true;
    m_ROWHeading = true;
    m_EditBar = false;
    m_CellProtection = false;
}

CSpreadView::~CSpreadView()
{
}

BEGIN_MESSAGE_MAP(CSpreadView, CView)
    ON_WM_SIZE() {afx_msg_MAP(CSpreadView)
        ON_COMMAND(ID_VIEW_GRIDLINES, OnViewGridlines)
        ON_UPDATE_COMMAND_UI(ID_VIEW_GRIDLINES, OnUpdateViewGridlines)
        ON_COMMAND(ID_VIEW_COL_HEADING, OnViewColHeading)
        ON_UPDATE_COMMAND_UI(ID_VIEW_COL_HEADING, OnUpdateViewColHeading)
        ON_COMMAND(ID_VIEW_ROW_HEADING, OnViewRowHeading)
        ON_UPDATE_COMMAND_UI(ID_VIEW_ROW_HEADING, OnUpdateViewRowHeading)
        ON_COMMAND(ID_VIEW_EDITBAR, OnViewEditbar)
        ON_UPDATE_COMMAND_UI(ID_VIEW_EDITBAR, OnUpdateViewEditbar)
        ON_COMMAND(ID_FORMAT_ALIGNMENT, OnFormatAlignment)
        ON_COMMAND(ID_FORMAT_FONT, OnFormatFont)
        ON_COMMAND(ID_FORMAT_BORDER, OnFormatBorder)
        ON_COMMAND(ID_FORMAT_PATTERN, OnFormatPattern)
        ON_COMMAND(ID_FORMAT_NUMBER, OnFormatNumber)
        ON_COMMAND(ID_FORMAT_PROTECTION, OnFormatProtection)
        ON_UPDATE_COMMAND_UI(ID_FORMAT_PROTECTION, OnUpdateFormatProtection)
        ON_COMMAND(ID_FORMAT_COL_WIDTH, OnFormatColWidth)
    }

END_MESSAGE_MAP()
void CSpreadView::OnDraw(CDC* pDC)
{
    CDocument* pDoc = GetDocument();
}

#ifdef _DEBUG
void CSpreadView::AssertValid() const
{
    CView::AssertValid();
}
#endif // _DEBUG

void CSpreadView::Dump(CDumpContext& dc) const
{
    CView::Dump(dc);
}

BOOL CSpreadView::Create(LPCSTR lpszClassName, LPCTSTR lpszWindowName, DWORD dwStyle, const RECT& rect, CWnd* pParentWnd, UINT nID, CCreatedContext* pContext)
{
    if (CWnd::Create(lpszClassName, lpszWindowName, dwStyle, rect, pParentWnd, nID, pContext) == 0)
    return 0;
    return 1;
}

BEGIN_MESSAGE_MAP(CSpreadView, CView)
    ON_COMMAND(ID_FORMAT_ROW_HEIGHT, OnFormatRowHeight)
    ON_COMMAND(ID_EDIT_CUT, OnEditCut)
    ON_COMMAND(ID_EDIT_COPY, OnEditCopy)
    ON_COMMAND(ID_EDIT_PASTE, OnEditPaste)
    ON_UPDAT_COMMAND_UI(ID_EDIT_PASTE, OnUpdateEditPaste)
    ON_COMMAND(ID_EDIT_CLEAR, OnEditClear)
    ON_COMMAND(ID_EDIT_INS_COLROW, OnEditInsColrow)
    ON_COMMAND(ID_EDIT_DEL_COLROW, OnEditDelColrow)
    ON_COMMAND(ID_EDIT_INSERT_SHEET, OnEditInsertSheet)
    ON_COMMAND(ID_EDIT_DELETE_SHEET, OnEditDeleteSheet)
    ON_COMMAND(ID_EDIT_SORT, OnEditSort)
    ON_COMMAND(ID_FILE_PRINT, OnFilePrint)
    ON_COMMAND(ID_FILE_PRINT_SETUP, OnFilePrintSetup)
    ON_COMMAND(ID_ANALYSIS_SMOVING, OnAnalysisSmoving)
    ON_COMMAND(ID_ANALYSIS_DECOMPO, OnAnalysisDecompo)
    ON_COMMAND(ID_ANALYSIS_CUSUM, OnAnalysisCusum)
    ON_COMMAND(ID_ANALYSIS_SEXPO, OnAnalysisSexpo)
    ON_COMMAND(ID_ANALYSIS_HEXPO, OnAnalysisHexpo)
    ON_COMMAND(ID_ANALYSIS_WEXPO, OnAnalysisWexpo)
    ON_COMMAND(ID_ANALYSIS_IDENTIFY, OnAnalysisIdentify)
    ON_COMMAND(ID_BT_CURRENCY, OnBtCurrency)
    ON_COMMAND(ID_BT_GENERAL, OnBtGeneral)
    ON_COMMAND(ID_BT_PERCENT, OnBtPercent)
    ON_COMMAND(ID_BT_SCIENTIFIC, OnBtScientific)
    ON_COMMAND(ID_ANALYSIS_RUN, OnAnalysisRun)
    ON_COMMAND(ID_ANALYSIS_DESEASONAL, OnAnalysisDeseasonal)
    ON_COMMAND(ID_ANALYSIS_DETREND, OnAnalysisDetrend)
    ON_COMMAND(ID_ANALYSIS_RFH, OnAnalysisRfh)

    //AFX_MSG_MAP
    END_MESSAGE_MAP()
if (m_ss.Create("Formula One Book", dwStyle, rect, this, 1) == 0)
    return 0;
m_ss.SetNumSheets(2);
m_ss.SetSheetName(1, "Data Entry");
m_ss.SetSheetName(2, "Result");
return TRUE;
}

void CSpreadView::OnInitialUpdate()
{
    CView::OnInitialUpdate();

    CRect rect;
    GetClientRect(rect);
    if (m_ss)
        m_ss.MoveWindow(0, 0, rect.Width(), rect.Height());

    // Try to get the ChartView pointer from CSplitterFrame
    CSplitterWnd* pParent = (CSplitterWnd*)GetParent();
    ASSERT (pParent->IsKindOf(RUNTIME_CLASS(CSplitterWnd)));
    pChartView = (CGraphView*)pParent->GetPane(0, 0);
    ASSERT (pChartView->IsKindOf(RUNTIME_CLASS(CGraphView)));
}

void CSpreadView::OnSize(UINT nType, int cx, int cy)
{
    CView::OnSize(nType, cx, cy);
    if (m_ss)
        m_ss.MoveWindow(0, 0, cx, cy);
}

void CSpreadView::OnViewEditbar()
{
    if (m_EditBar)
        m_ss.SetShowEditBar(FALSE);
    else
        m_ss.SetShowEditBar(TRUE);
    m_EditBar = !m_EditBar;
}

void CSpreadView::OnFormatPattern()
{
    m_ss.FormatPatternDlg();
}

void CSpreadView::OnFormatNumber()
{
    m_ss.FormatNumberDlg();
}

void CSpreadView::OnFormatProtection()
{
    if (m_CellProtection)
        m_ss.SetEnableProtection(FALSE);
    else
m_ss.SetEnableProtection(TRUE);
   m_CellProtection = !m_CellProtection;

void CSpreadView::OnUpdateFormatProtection(CCmdUI* pCmdUI)
{
   if (m_CellProtection)
      pCmdUI->SetText("Disable Protection");
   else
      pCmdUI->SetText("Enable Protection");
}

void CSpreadView::OnFormatColWidth()
{
   m_ss.ColWidthDlg();
}

void CSpreadView::OnFormatRowHeight()
{
   m_ss.RowHeightDlg();
}

void CSpreadView::OnEditCut()
{
   m_ss.EditCut();
}

void CSpreadView::OnEditCopy()
{
   m_ss.EditCopy();
}

void CSpreadView::OnEditPaste()
{
   m_ss.EditPaste();
}

void CSpreadView::OnUpdateEditPaste(CCmdUI* pCmdUI)
{
}

void CSpreadView::OnEditClear()
{
   m_ss.EditClear(1);
}

void CSpreadView::OnEditInsColrow()
{
   long  max_col = 0;
   long  r1 = 0, c1 = 0, r2 = 0, c2 = 0;
   max_col = m_ss.GetMaxCol();
   m_ss.GetSelection( 0, (long*)&r1, (long*)&c1, (long*)&r2,
                        (long*)&c2);
   if (r1 == 1 && r2 >= max_col)
      m_ss.EditInsert( ITP_RIGHT);
   else
      m_ss.EditInsert( ITP_DOWN);
void CSpreadView::OnEditDelColrow()
{
    long max_col = 0;
    long r1 = 0, c1 = 0, r2 = 0, c2 = 0;
    max_col = m_ss.GetMaxCol();
    m_ss.GetSelection(0, (long*)&r1, (long*)&c1, (long*)&r2, (long*)&c2);
    if (r1 == 1 && r2 >= max_col)
        m_ss.EditDelete( ITP_RIGHT);
    else
        m_ss.EditDelete( ITP_DOWN);
}

void CSpreadView::OnEditInsertSheet()
{
    m_ss.EditInsertSheets();
}

void CSpreadView::OnAnalysisCusum()
{
    CCuSum cusumdlg;
    if (cusumdlg.DoModal() == IDOK)
    {
        long nRows;
        double Enter[rows][cols];
        if (m_ss.GetSheet() != 1)
            { AfxMessageBox("Data Entry Should Start with the Data Entry Sheet!", MB_OK);
        return;
        }
        nRows = m_ss.GetSelEndRow()-m_ss.GetSelStartRow()+1;
        if (m_ss.GetSelStartRow() != 1)
            { AfxMessageBox("Data Entry Should Start with the First Row!", MB_OK);
        return;
        }
        if (nRows < 24)
            { AfxMessageBox("Data must be Entered at least 24 Sets!", MB_OK);
        return;
        }
        m_ss.SetSheet(2);
        ClearSheet();
        m_ss.SetColText(1, "NO");
        m_ss.SetColWidth(1, 1, 900, FALSE);
        m_ss.SetColText(2, "Raw Data");
        m_ss.SetColWidth(2, 2, 2600, FALSE);
        m_ss.SetColText(3, "Normalized Data");
        m_ss.SetColWidth(3, 3, 4000, FALSE);
        m_ss.SetColText(4, "Sum");
        m_ss.SetColWidth(4, 4, 3000, FALSE);
        m_ss.SetColText(5, "CUSUM");
        m_ss.SetColWidth(5, 5, 3000, FALSE);
        m_ss.SetColText(6, "Difference");
        m_ss.SetColWidth(6, 6, 3000, FALSE);
m_ss.SetColText(7, "Shift Size");
m_ss.SetColWidth(7, 7, 3000, FALSE);
m_ss.SetColText(8, "Shift Detected");
m_ss.SetColWidth(8, 8, 4000, FALSE);
m_ss.SetActiveCell(1, 1);
m_ss.SetText(" Cumulative Sum Control Chart");

for (long x=1; x<=nRows; x++) //arrangement for 1st col
{
    m_ss.SetActiveCell(x+1, 1);
    m_ss.SetNumber(double(x));
}

for (long y=1; y<=nRows; y++) //arrangement for 2th col
{
    m_ss.SetSheet(1);
    m_ss.SetActiveCell(y, m_ss.GetSelStartCol());
    Enter[y][2] = m_ss.GetNumber();
    m_ss.SetSheet(2);
    m_ss.SetActiveCell(y+1, 2);
    m_ss.SetNumber(Enter[y][2]);
}

double totall=0, total2=0, stdv=0, avg=0;
long templ;

if (cusumdlg.m_option == 0) // Determine which option you choose
{
    templ = long(cusumdlg.m_beforedata);
    for (long z=1; z<=templ; z++) //calculation for average
    {
        totall=totall+Enter[z][2];
    }
    avg=totall/templ;

    for (long i=1; i<=templ; i++) //calculation for standard deviation
    {
        total2=total2+(Enter[i][2]-avg)*(Enter[i][2]-avg);
    }
    stdv=sqrt(total2/(templ-1));
}
else
{
    for (long z5=1; z5<=nRows; z5++) //calculation for average
    {
        totall=totall+Enter[z5][2];
    }
    avg=totall/nRows;

    for (long i5=1; i5<=nRows; i5++) //calculation for standard deviation
    {
        total2 = total2+(Enter[i5][2]-avg)*
            (Enter[i5][2]-avg);
    }
    stdv=sqrt(total2/(nRows-1));
}
for (long j=1; j<=nRows; j++)  //Calculation for CUSUM
{
    Enter[0][4] = 0;
Enter[j][3] = (Enter[j][2]-avg) / stdv;
Enter[j][4] = Enter[j-1][4] + Enter[j][3];Enter[j][5] = Enter[j][4] / sqrt(double(j));m_ss.SetActiveCell(j+1, 3);
m_ss.SetNumber(Enter[j][3]);
m_ss.SetActiveCell(j+1, 4);
m_ss.SetNumber(Enter[j][4]);m_ss.SetActiveCell(j+1, 5);
m_ss.SetNumber(Enter[j][5]);
}

for (long k=1; k<=nRows-1; k++)   //Calculation for Difference
{
    Enter[k][6] = Enter[k+1][5] - Enter[k][5];
m_ss.SetActiveCell(k+2, 6);
m_ss.SetNumber(Enter[k][6]);
}

int counter=2;
double shift=Enter[1][6];
for (long l=1; l<=nRows-1; l++)  //Calculation for Shift
{
    if (Enter[l+1][6]*Enter[l][6]>0)
    {
        counter=counter+1;
        shift=shift+Enter[l+1][6];
    }
    else
    {
        //shift=fabs(shift);
        switch(counter) // check the critical limit (8 rules)
        {
            case 2:
                if (shift>=3 || shift<=-3)
                {
                    m_ss.SetActiveCell(l+1, 8);
m_ss.SetText(" Start of Trend");
m_ss.SetActiveCell(l+2, 8);
m_ss.SetText(" End of Trend");
m_ss.SetActiveCell(l+2, 7);
m_ss.SetNumber(shift);
                } break;

            case 3:
                if (shift>=1.75 || shift<=-1.75)
                {
                    m_ss.SetActiveCell(l, 8);
m_ss.SetText(" Start of Trend");
m_ss.SetActiveCell(l+2, 8);
m_ss.SetText(" End of Trend");
m_ss.SetActiveCell(l+2, 7);
m_ss.SetNumber(shift);
                } break;

            case 4:
                
        }
if (shift>=1.25 | shift<=-1.25)
  { m_ss.SetActiveCell(l-1, 8);
    m_ss.SetText(" Start of Trend");
    m_ss.SetActiveCell(l+2, 8);
    m_ss.SetText(" End of Trend");
    m_ss.SetActiveCell(l+2, 7);
    m_ss.SetNumber(shift);
  }
  break;

  case 5:
    if (shift>=0.9 | shift<=-0.9)
      { m_ss.SetActiveCell(l-2, 8);
        m_ss.SetText(" Start of Trend");
        m_ss.SetActiveCell(l+2, 8);
        m_ss.SetText(" End of Trend");
        m_ss.SetActiveCell(l+2, 7);
        m_ss.SetNumber(shift);
      }
    break;

    case 6:
      if (shift>=0.6 | shift<=-0.6)
      { m_ss.SetActiveCell(l-3, 8);
        m_ss.SetText(" Start of Trend");
        m_ss.SetActiveCell(l+2, 8);
        m_ss.SetText(" End of Trend");
        m_ss.SetActiveCell(l+2, 7);
        m_ss.SetNumber(shift);
      }
    break;

    case 7:
      if (shift>=0.4 | shift<=-0.4)
      { m_ss.SetActiveCell(l-4, 8);
        m_ss.SetText(" Start of Trend");
        m_ss.SetActiveCell(l+2, 8);
        m_ss.SetText(" End of Trend");
        m_ss.SetActiveCell(l+2, 7);
        m_ss.SetNumber(shift);
      }
    break;

    case 8:
      if (shift>=0.25 | shift<=-0.25)
      { m_ss.SetActiveCell(l-counter+3, 8);
        m_ss.SetText(" Start of Trend");
        m_ss.SetActiveCell(l+2, 8);
        m_ss.SetText(" End of Trend");
        m_ss.SetActiveCell(l+2, 7);
        m_ss.SetNumber(shift);
      }
    break;

default:
   { m_ss.SetActiveCell(l-counter+3, 8);
     m_ss.SetText(" Start of Trend");
     m_ss.SetActiveCell(l+2, 8);
     m_ss.SetText(" End of Trend");
     m_ss.SetActiveCell(l+2, 7);
     m_ss.SetNumber(shift);
   }
// end of switch

shift=Enter[l+1][6];
counter=2;
} // end of else
if (for
Handle chart here
char buffer[40];
pChartView->m_graph.Refresh();
pChartView->m_graph.SetChartType(16);
pChartView->m_graph.SetRandomFill(FALSE);
pChartView->m_graph.SetColumnCount(6);
pChartView->m_graph.SetRowCount(short(nRows));
pChartView->m_graph.SetColumn(1);
for (long m1=1; m1<nRows; m1++)
{
    pChartView->m_graph.SetRow(short(m1));
    _ltoa(m1, buffer, 10);
pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetColumn(2);
for (long m2=1; m2<=nRows; m2++)
{
    pChartView->m_graph.SetRow(short(m2));
    _gcvt(Enter[m2][5], 12, buffer);
pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetColumn(3);
for (long m3=1; m3<=nRows; m3++)
{
    pChartView->m_graph.SetRow(short(m3));
    _ltoa(m3, buffer, 10);
pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetColumn(4);
for (long m4=1; m4<=nRows; m4++)
{
    pChartView->m_graph.SetRow(short(m4));
pChartView->m_graph.SetData("3");
}
pChartView->m_graph.SetColumn(5);
for (long m5=1; m5<=nRows; m5++)
{
    pChartView->m_graph.SetRow(short(m5));
    _ltoa(m5, buffer, 10);
pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetColumn(6);
for (long m6=1; m6<=nRows; m6++)
{
    pChartView->m_graph.SetRow(short(m6));
pChartView->m_graph.SetData("-3");
}
for (long m7=1; m7<=nRows; m7++)
{
    pChartView->m_graph.SetRow(short(m7));
    _ltoa(m7, buffer, 10);
pChartView->m_graph.SetRowLabel(buffer);
pChartView->m_graph.SetTitleText("CUSUM Chart");
}

void CSpreadView::OnAnalysisIdentify()
{
    CSelectDlg seldlg;
    long nRows;
    double Enter[300][1];
    if (m_ss.GetSheet() != 1)
    {
        AfxMessageBox("Data Entry Should Start with the Data
Entry Sheet!", MB_OK);
        return;
    }
    nRows = m_ss.GetSelEndRow() - m_ss.GetSelStartRow() + 1;
    if (m_ss.GetSelStartRow() != 1)
    {
        AfxMessageBox("Data Entry Should Start with the First
Row!", MB_OK);
        return;
    }
    if (nRows < 24)
    {
        AfxMessageBox("Data must be Entered at least 24 sets!",
MB_OK);
        return;
    }
    for (long x = 1; x <= nRows; x++) //get data from spreadsheet1
    {
        m_ss.SetActiveCell(x, m_ss.GetSelStartCol());
        Enter[x][1] = m_ss.GetNumber();
    }
    m_ss.SetSheet(2); //clean spreadsheet2 then
    ClearSheet();
    m_ss.SetSheet(1);

    // Draw chart here
    char buffer[40];
    pChartView->m_graph.Refresh();
    pChartView->m_graph.SetChartType(3);
    pChartView->m_graph.SetAutoIncrement(1);
    pChartView->m_graph.SetRowCount(short(nRows));
    pChartView->m_graph.SetColumnCount(1);
    for (long y = 1; y <= nRows; y++)
    {
        gcvt(Enter[y][1], 12, buffer);
        pChartView->m_graph.SetData(buffer);
    }
    pChartView->m_graph.SetTitleText("Determine Data Pattern");
    if (seldlg.DoModal() == IDCANCEL)
        return;
}

void CSpreadView::OnAnalysisDeseasonal()
{
    CDeSeasonal deseasonalDlg;
    long nRows, period;
    double Enter[rows][cols];
if ( m_ss.GetSheet() != 1 )
    { AfxMessageBox("Data Entry Should Start with the Data
Entry Sheet!"); return; }

nRows = m_ss.GetSelEndRow() - m_ss.GetSelStartRow() + 1;
if (m_ss.GetSelStartRow() != 1 )
    { AfxMessageBox("Data Entry Should Start with the First Row
!"); return; }

if (long(deseasonaldlg.m_period) == 0) //if m_period==1 then
    { period = 12;
        if (nRows % 12 != 0)
            { AfxMessageBox("Data Sets must be Entered by Year with 12
months!"); return; }
    }
else //Assign period for monthly or quarterly
    { period = 4;
        if (nRows % 4 != 0)
            { AfxMessageBox("Data Sets must be Entered Quarterly!",
MB_OK); return; }
    }

if (deseasonaldlg.DoModal() == IDOK)
{
    if (deseasonaldlg.m_method == 0)
        { 7/ which method we choose? winter's or
decomposition?

        m_ss.SetSheet(2);
        ClearSheet();
        m_ss.SetColText(1, "NO");
        m_ss.SetColWidth(1, 1, 900, FALSE);
        m_ss.SetColText(2, "Raw Data");
        m_ss.SetColWidth(2, 2, 2600, FALSE);
        m_ss.SetColText(3, "Deseasonalized Data");
        m_ss.SetColWidth(3, 3, 5200, FALSE);
        m_ss.SetColText(4, "Squared Error");
        m_ss.SetColWidth(4, 4, 3600, FALSE);
        m_ss.SetColText(5, "RMSE");
        m_ss.SetColWidth(5, 5, 3200, FALSE);
        m_ss.SetActiveCell(1, 1);
        m_ss.SetText("Deseasonalize Data through Time Series
- Decompositions");

        for (long x=1; x<=nRows; x++) // 1st column arrangement
            { m_ss.SetActiveCell(x+1, 1);
                m_ss.SetNumber(double(x));
            }
for (long i=1; i<=nRows; i++)  // 2nd col arrangement
{
    m_ss.SetSheet(1);
    m_ss.SetActiveCell(i, m_ss.GetSelStartCol());
    Enter[i][2] = m_ss.GetNumber();
    m_ss.SetSheet(2);
    m_ss.SetActiveCell(i+1, 2);
    m_ss.SetNumber(Enter[i][2]);
}

for (long j=1; j<=nRows-period; j++)
{
    double subsum = 0.0;
    for (long k=j+1; k<=j+period-1; k++)
    {
        subsum += Enter[k][2];
    }
    Enter[j][3]=(0.5*Enter[j][2]+subsum+0.5*
        Enter[j+period][2])/period;
}

for (long l=1; l<=nRows-period; l++)  // 4th column arrangement
{
    Enter[1][4] = Enter[1+period/2][2]/Enter[1][3];
}

for (long a1=1; a1<=period/2; a1++)  // 5th column arrangement
{
    Enter[a1][10] =
        (Enter[a1+period/2][4]+Enter[a1+3*period/2][4])
        /((nRows-period)/period);  // period change 12
    Enter[a1+period/2][10] =
        (Enter[a1][4]+Enter[a1+period][4])
        /((nRows-period)/period);
}

double subsuma = 0.0;
for (long a2=1; a2<=period; a2++)
{ subsuma += Enter[a2][10]; }

for (long a3=1; a3<=nRows/period; a3++)
{
    for (long a4=1; a4<=period; a4++)
    {
        Enter[(a3-1)*period+a4][5] =
            Enter[a4][10]*period/subsuma;
    }
}

for (long a6=1; a6<=nRows; a6++)  // 3th column for
{
    Enter[a6][6] = Enter[a6][2] / Enter[a6][5];
    m_ss.SetActiveCell(a6+1, 3);
    m_ss.SetNumber(Enter[a6][6]);
}

for (long a7=1; a7<=nRows; a7++)  // squared error cal for
                                   // 4th col
{
    Enter[a7][7] = (Enter[a7][2]-Enter[a7][6])*%
        (Enter[a7][2]-Enter[a7][6]);
    m_ss.SetActiveCell(a7+1, 4);
    m_ss.SetNumber(Enter[a7][7]);
}

double rmse=0, templ=0;
for (long a8=1; a8<=nRows; a8++)  // calculation for RMSE

{ template = templ + Enter[a8][7];
} rmse = sqrt(templ/(nRows));
m_s.SetActiveCell(1, 5);
m_s.SetNumber(rmse);

// Handle chart here
char buffer[40];
pChartView->m_graph.Refresh();
pChartView->m_graph.SetChartType(3);
pChartView->m_graph.SetRandomFill(FALSE);
pChartView->m_graph.SetRowCount(short(nRows));
pChartView->m_graph.SetColumnType(2);

pChartView->m_graph.SetColumn(1);
for (long z=1; z<=nRows; z++)
{ pChartView->m_graph.SetRow(short(z));
  gcvt(Enter[z][2], 12, buffer);
  pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetColumn(2);
for (long y=1; y<=nRows; y++)
{ pChartView->m_graph.SetRow(short(y));
  gcvt(Enter[y][6], 12, buffer);
  pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetTitleText("Deseasonalized Data through Time Series Decomposition");
else
{
  m_s.SetSheet(2);
  ClearSheet();
  m_s.SetColText(1, "NO");
  m_s.SetColWidth(1, 1, 900, FALSE);
  m_s.SetColText(2, "Raw Data");
  m_s.SetColWidth(2, 2, 2600, FALSE);
  m_s.SetColText(3, "Seasonal Adj.");
  m_s.SetColWidth(3, 3, 3700, FALSE);
  m_s.SetColText(4, "Deseasonalized Data");
  m_s.SetColWidth(4, 4, 5000, FALSE);
  m_s.SetColText(5, "Sqr. Error");
  m_s.SetColWidth(5, 5, 3000, FALSE);
  m_s.SetColText(6, "RMSE");
  m_s.SetColWidth(6, 6, 3200, FALSE);
  m_s.SetActiveCell(1, 1);
  m_s.SetText("Deseasonalize data through Winter's Exponential Smoothing");

  for (long x=1; x<=nRows; x++) //calculation for 1st col
  {
    m_s.SetActiveCell(x+1, 1);
    m_s.SetNumber(double(x));
  }
}
for (long i=period+1; i<=period+nRows; i++) //calculation for 2th col
{
    m_ss.SetSheet(1);
    m_ss.SetActiveCell(i-period, m_ss.GetSelStartCol());
    Enter[i][2] = m_ss.GetNumber();
    m_ss.SetSheet(2);
    m_ss.SetActiveCell(i+1-period, 2);
    m_ss.SetNumber(Enter[i][2]);
}

cchar *p1, *p2, *p3;
double q1, q2, q3;
q1 = strtod(deseasonaldlg.m_dwexpol, &p1);
q2 = strtod(deseasonaldlg.m_dwexpol2, &p2);
q3 = strtod(deseasonaldlg.m_dwexpol3, &p3);
Enter[period][3] = Enter[period+1][2];

for (long j=1; j<=period; j++) //calculation for 3rd col
{
    Enter[j][4] = 1.00;
}

for (long k=period+1; k<=nRows+period; k++)
{
    Enter[k][3] = q1*Enter[k][2]/Enter[k-period][4]+
                  (1-q1)*(Enter[k-1][3]+Enter[k-1][5]);
    Enter[k][4] = q2*Enter[k][2]/Enter[k][3]+(1.0-q2)*
                  Enter[k-period][4];
    Enter[k][5] = q3*(Enter[k][3]-Enter[k-1][3]) +
                  (1-q3)*Enter[k-1][5];
    Enter[k+period][6] = (Enter[k][3]+period*Enter[k][5])*Enter[k][4];
}

for (long m1=period+1; m1<=nRows+period; m1++)
{
    m_ss.SetActiveCell(m1+1-period, 3);
    m_ss.SetNumber(Enter[m1][4]);
}

for (long m2=period+1; m2<=nRows+period; m2++)//deseasonalize data in 3rd cols
{
    Enter[m2][7] = Enter[m2][2]*Enter[m2][4];
    m_ss.SetActiveCell(m2+1-period, 4);
    m_ss.SetNumber(Enter[m2][7]);
}

for (long k1=period+1; k1<=nRows+period; k1++)
{
    Enter[k1][8] = (Enter[k1][2]-Enter[k1][7])*
                  (Enter[k1][2]-Enter[k1][7]);
    m_ss.SetActiveCell(k1+1-period, 5);
    m_ss.SetNumber(Enter[k1][8]);
}

double rmse=0, temp1=0;
for (long k2=period+1; k2<=nRows+period; k2++)//calculation for 6 col(RMSE)
temp1 = temp1 + Enter[k2][8];
rmse = sqrt(temp1/(nRows));
m_ss.SetActiveCell(1, 6);
m_ss.SetNumber(rmse);

// Handle chart here
char buffer[40];
pChartView->m_graph.Refresh();
pChartView->m_graph.SetChartType(3);
pChartView->m_graph.SetRandomFill(FALSE);
pChartView->m_graph.setColumnCount(2);
pChartView->m_graph.setColumn(1);
pChartView->m_graph.SetRowCount(short(nRows));
for (long z=period+1; z<=nRows+period; z++)
{
    pChartView->m_graph.SetRow(short(z-period));
    _gct(Enter[z][2], 12, buffer);
pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetColumn(2);
pChartView->m_graph.SetRowCount(short(nRows));
for (long y=period+1; y<=nRows+period; y++)
{
    pChartView->m_graph.SetRow(short(y-period));
    _gct(Enter[y][7], 12, buffer);
pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetTitleText("Deseasonalize Data through
  Winter's Exponential Smoothing");
}
}

void CSpreadView::OnAnalysisDetrend()
{
    CDeTrend detrenddlg;
    if (detrenddlg.DoModal() == IDOK)
    {
        long nRows;
        double Enter[rows][cols];
        if (m_ss.GetSheet() != 1 )
        {
            AfxMessageBox("Data Entry Should Start with the Data
Entry Sheet!", MB_OK);
            return;
        }
        nRows = m_ss.GetSelEndRow() - m_ss.GetSelStartRow() + 1;
        if (m_ss.GetSelStartRow() != 1 )
        {
            AfxMessageBox("Data Entry Should Start with the First
Row !", MB_OK);
            return;
        }
        if (nRows < 24)
        {
            AfxMessageBox("Data Sets must be Entered at least 24 sets!",
            MB_OK);
            return;
        }
    }
m_ss.SetSheet(2);
ClearSheet();
m_ss.SetColText(1, "NO");
m_ss.SetColWidth(1, 1, 900, FALSE);
m_ss.SetColText(2, "Raw Data");
m_ss.SetColWidth(2, 2, 2600, FALSE);
m_ss.SetColText(3, "Trend Estimate");
m_ss.SetColWidth(3, 3, 3800, FALSE);
m_ss.SetColText(4, "Detrend Data");
m_ss.SetColWidth(4, 4, 3800, FALSE);
m_ss.SetColText(5, "Sqr. Error");
m_ss.SetColWidth(5, 5, 3000, FALSE);
m_ss.SetColText(6, "RMSE");
m_ss.SetColWidth(6, 6, 3200, FALSE);
m_ss.SetActiveCell(1, 1);
m_ss.SetText("Detrend data through Holt's Exponential Smoothing");

for (long x=1; x<=nRows; x++)  // arrangement for 1st col
{
    m_ss.SetActiveCell(x+1, 1);
    m_ss.SetNumber(double(x));
}

for (long y=1; y<=nRows; y++)  // calculation for 2th col
{
    m_ss.SetSheet(1);
    m_ss.SetActiveCell(y, m_ss.GetSelStartCol());
Enter[y][2] = m_ss.GetNumber();
m_ss.SetSheet(2);
    m_ss.SetActiveCell(y+1, 2);
m_ss.SetNumber(Enter[y][2]);
}

char *p1, *p2;
double q1, q2;
q1 = strtod(detrenddlg.m_dtholt1, &p1);
q2 = strtod(detrenddlg.m_dtholt2, &p2);

m_ss.SetActiveCell(2, 3);
Enter[1][3] = Enter[1][2];
m_ss.SetNumber(Enter[1][4]);
m_ss.SetActiveCell(2, 4);
Enter[1][4] = 0;
m_ss.SetActiveCell(2, 5);
Enter[1][5] = Enter[1][2];

for (long z=2; z<=nRows+1; z++)  //calculation for 3rd col
{
    Enter[z][3] = q1*Enter[z-1][2]+(1-q1)*
    (Enter[z-1][3]+Enter[z-1][4]);
Enter[z][4] = q2*(Enter[z][3]-Enter[z-1][3])+(1-q2)*
    Enter[z-1][4];
Enter[z][5] = Enter[z][3]+Enter[z][4];
}

for (long j=1; j<=nRows; j++)  //arrangement for 4th col
{
m_ss.SetActiveCell(j+2, 3);
m_ss.SetNumber(Enter[j+1][4]);

for (long k=1; k<=nRows; k++) //5th col
{
    Enter[k][7] = Enter[k][2] - Enter[k][4];
m_ss.SetActiveCell(k+1, 4);
m_ss.SetNumber(Enter[k][7]);
}

for (long kl=1; kl<=nRows; kl++) //squared error cal for 6th col
{
    Enter[kl][6] = (Enter[kl][2] - Enter[kl][7])*
    (Enter[kl][2] - Enter[kl][7]);
m_ss.SetActiveCell(kl+1, 5);
m_ss.SetNumber(Enter[kl][6]);
}

double rmse=0, templ=0;
for (long k2=1; k2<=nRows; k2++) //calculation for 7 col(RMSE)
{
    templ = templ + Enter[k2][6];
}
rmse = sqrt(templ/(nRows));
m_ss.SetActiveCell(1, 6);
m_ss.SetNumber(rmse);

// Handle chart here
char buffer[40];
pChartView->m_graph.Refresh();
pChartView->m_graph.SetChartType(3);
pChartView->m_graph.SetRandomFill(FALSE);
pChartView->m_graph.SetColumnCount(2);
pChartView->m_graph.SetColumn(1);
pChartView->m_graph.SetRowCount(short(nRows));
for (long m1=1; m1<=nRows; m1++)
{
    pChartView->m_graph.SetRow(short(m1));
    gcvt(Enter[m1][2], 12, buffer);
pChartView->m_graph.SetData(buffer);
}

pChartView->m_graph.SetColumn(2);
pChartView->m_graph.SetRowCount(short(nRows));
for (long m2=1; m2<=nRows; m2++)
{
    pChartView->m_graph.SetRow(short(m2));
    gcvt(Enter[m2][7], 12, buffer);
pChartView->m_graph.SetData(buffer);
}
pChartView->m_graph.SetTitleText("Detrend Data throuth Holt's
Exponential Smoothing");
}
// GraphView.h : header file

#include "vtchart.h"

class CGraphView : public CView
{
   protected:
      CGraphView(); // protected constructor used by dynamic creation
      DECLARE_DYNCREATE(CGraphView)
   
   Attributes
   public:
      CVtChart m_graph;
   
   // Operations
   public:
   
   // Overrides
   // ClassWizard generated virtual function overrides
  //{{AFX_VIRTUAL(CGraphView)
      public:
         virtual BOOL Create(LPCTSTR IpszClassName, LPCTSTR lpszWindowName, DWORD dwStyle, canst RECT& rect, CWnd* pParentWnd, UINT nID, CCreateContext* pContext = NULL);
         virtual void OnInitialUpdate();
      protected:
         virtual void OnDraw(CDC* pOe);
      overridden to draw this view
   }}AFX_VIRTUAL
   
   // Implementation
   protected:
      virtual ~CGraphView();
      #ifdef _DEBUG
         virtual void AssertValid() const;
         virtual void Dump(CDumpContext& dc) const;
      #endif
   
   // Generated message map functions
   protected:
  //{{AFX_MSG(CGraphView)
      afx_msg void OnSize(UINT nType, int cx, int cy);
   }}AFX_MSG
   
   DECLARE_MESSAGE_MAP()}


// CCuSum dialog

class CCuSum : public CDialog
{
    // Construction
    public:
        CCuSum(CWnd* pParent = NULL);   // standard constructor

    // Dialog Data
        //{{AFX_DATA(CCuSum)
        enum { IDD = IDD_CUSUM };
        UINT m_beforedata;
        int  m_option;
        //}}AFX_DATA

    // Overrides
        // ClassWizard generated virtual function overrides
        //{{AFX_VIRTUAL(CCuSum)
        protected:
            virtual void DoDataExchange(CDataExchange* pDX);
        //}}AFX_VIRTUAL

    // Implementation
    protected:

        // Generated message map functions
        //{{AFX_MSG(CCuSum)
        afx_msg void OnRdAllData();
        afx_msg void OnRdBeforea();
        //}}AFX_MSG
        DECLARE_MESSAGE_MAP()
    
};

// CSelectDlg dialog
class CSelectDlg : public CDialog
{
    // Construction
    public:
        CSelectDlg(CWnd* pParent = NULL);   // standard constructor

    // Dialog Data
        //{{AFX_DATA(CSelectDlg)
        enum { IDD = IDD_SELECT };
        CEdit m_comment;
        CEdit m_select;
        //}}AFX_DATA

    // Overrides
        // ClassWizard generated virtual function overrides
        //{{AFX_VIRTUAL(CSelectDlg)
        protected:
            virtual void DoDataExchange(CDataExchange* pDX);
        //}}AFX_VIRTUAL

    // Implementation
    protected:

// Generated message map functions
//
afx_msg void OnRdlrregular();
afx_msg void OnRdlIt();
afx_msg void OnRdlIs();
afx_msg void OnRdlIts();
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
}
class CRunTest : public CDialog
{
    // Construction
    public:
    CRunTest(CWnd* pParent = NULL); // standard constructor

    // Dialog Data
    //

    // Overrides
    // ClassWizard generated virtual function overrides
    //
    // Implementation
    protected:

    // Generated message map functions
    //

    // CDeSeasonal dialog
    class CDeSeasonal : public CDialog
    {
        // Construction
        public:
        CDeSeasonal(CWnd* pParent = NULL); // standard constructor

        // Dialog Data
        //

        // Overrides
        // ClassWizard generated virtual function overrides
protected:
virtual void DoDataExchange(CDataExchange* pDX);

// Implementation
protected:

// Generated message map functions
//{{AFX_MSG(CDeSeasonal)
afx_msg void OnRdDecompo();
afx_msg void OnRdWinter();
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
};

// COeTrend dialog
class CDeTrend : public CDialog
{

// Construction
public:
CDeTrend(CWnd* pParent = NULL); // standard constructor

// Dialog Data
//{{AFX_DATA(CDeTrend)
enum { IDD = IDD_DETREND };
int m_method;
CString m_dtholt1;
CString m_dtholt2;
//}}AFX_DATA

protected:

// Generated message map functions
//{{AFX_MSG(CDeTrend)
//}}AFX_MSG
DECLARE_MESSAGE_MAP()

};
// AnalysisDlg.cpp : implementation file
#include "stdafx.h"
#include "TSIE.h"
#include "AnalysisDlg.h"

CCuSum::CCuSum(CWnd* pParent /*=NULL*/) :
    CDialog(CCuSum::IDD, pParent)
{
    m_beforedata = 24;
    m_option = 0;
}

void CCuSum::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    DDX_Text(pDX, IDC_EDIT BEFORE, m_beforedata);
    DDX_Radio(pDX, IDC_RD_ALLDATA, m_option);
}

BEGIN_MESSAGE_MAP(CCuSum, CDialog)
    //{{AFX_MSG_MAP(CCuSum)
    ON_BN_CLICKED(IDC_RD_ALLDATA, OnRdAlldata)
    ON_BN_CLICKED(IDC_RD_BEFORE, OnRdBeforea)
   //}}AFX_MSG_MAP
END_MESSAGE_MAP()

void CCuSum::OnRdBeforea()
{
    UpdateData();
    GetDlgItem(IDC_EDIT BEFORE)->EnableWindow();
}

void CCuSum::OnRdAlldata()
{
    UpdateData();
    GetDlgItem(IDC_EDIT BEFORE)->EnableWindow(FALSE);
}

// CSelectDlg dialog
CSelectDlg::CSelectDlg(CWnd* pParent /*=NULL*/) :
    CDialog(CSelectDlg::IDD, pParent)
{
}

void CSelectDlg::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    //{{AFX_DATA_MAP(CSelectDlg)
    DDX_Control(pDX, IDC_EDIT COMMENT, m_comment);
    DDX_Control(pDX, IDC_EDIT_SEL, m_select);
    //}}AFX_DATA_MAP
}

BEGIN_MESSAGE_MAP(CSelectDlg, CDialog)
    //{{AFX_MSG_MAP(CSelectDlg)
    ON_BN_CLICKED(IDC_RD_IRREGULAR, OnRdIrregular)
    ON_BN_CLICKED(IDC_RD_IT, OnRdIt)
   //}}AFX_MSG_MAP
END_MESSAGE_MAP()
ON_BN_CLICKED(IDC_RD_IS, OnRdIs)
ON_BN_CLICKED(IDC_RD_ITS, OnRdIts)
//}}AFX_MSG_MAP
END_MESSAGE_MAP()

// CSelectDlg message handlers
void CSelectDlg::OnRdIrregular()
{
    m_comment.SetWindowText("No need to Deseasonalize or Detrend");
    m_select.SetWindowText("Do Nothing");
}

void CSelectDlg::OnRdIt()
{
    m_comment.SetWindowText("Need to Detrend");
    m_select.SetWindowText("Holt's Exponential Smoothing or Simple Linear Regression");
}

void CSelectDlg::OnRdIs()
{
    m_comment.SetWindowText("Need to Deseasonalize");
    m_select.SetWindowText("Time Series Decomposition or Winter's Exponential Smoothing");
}

void CSelectDlg::OnRdIts()
{
    m_comment.SetWindowText("Need to Deseasonalize and Detrend Both");
    m_select.SetWindowText("Winter's Exponential Smoothing");
}

// CRunTest dialog
CRunTest::CRunTest(CWnd* pParent /*=NULL*/)
: CDialog(CRunTest::IDD, pParent)
{
}

void CRunTest::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
}

BEGIN_MESSAGE_MAP(CRunTest, CDialog)
END_MESSAGE_MAP()

// CDeSeasonal dialog
CDeSeasonal::CDeSeasonal(CWnd* pParent /*=NULL*/)
: CDialog(CDeSeasonal::IDD, pParent)
{
    m_period = 0;
    m_method = 0;
    m_dwexpo1 = \_T("0.40");
    m_dwexpo2 = \_T("0.30");
    m_dwexpo3 = \_T("0.10");
}

void CDeSeasonal::DoDataExchange(CDataExchange* pDX)
BEGIN_MESSAGE_MAP(CDeSeasonal, CDialog)
    // (AFX_MSG_MAP(CDeSeasonal)
    ON_BN_CLICKED(IDC_RD_DECOMPO, OnRdDecompo)
    ON_BN_CLICKED(IDC_RD_WINTER, OnRdWinter)
    //)AFX_MSG_MAP
END_MESSAGE_MAP()

// CDeSeasonal message handlers
void CDeSeasonal::OnRdDecompo()
{
    UpdateData();
    GetDlgItem(IDC_DSWEXP01)->EnableWindow(FALSE);
    GetDlgItem(IDC_DSWEXP02)->EnableWindow(FALSE);
    GetDlgItem(IDC_DSWEXP03)->EnableWindow(FALSE);
}

void CDeSeasonal::OnRdWinter()
{
    UpdateData();
    GetDlgItem(IDC_DSWEXP01)->EnableWindow();
    GetDlgItem(IDC_DSWEXP02)->EnableWindow();
    GetDlgItem(IDC_DSWEXP03)->EnableWindow();
}

// CDeTrend dialog
CDeTrend::CDeTrend(CWnd* pParent /*=NULL*/)
: CDialog(CDeTrend::IDD, pParent)
{
    // (AFX_DATA_INIT(CDeTrend)
    m_method = 0;
    m_dtholt1 = T("0.30");
    m_dtholt2 = T("0.10");
    //)AFX_DATA_INIT
}

void CDeTrend::DoDataExchange(CDataExchange* pDX)
{
    CDialog::DoDataExchange(pDX);
    // (AFX_DATA_MAP(CDeTrend)
    DDX_Radio(pDX, IDC_RD_HOLTS, m_method);
    DDX_CBString(pDX, IDC_DT_HOLT1, m_dtholt1);
    DDX_CBString(pDX, IDC_DT_HOLT2, m_dtholt2);
    //)AFX_DATA_MAP
}