Time-Series Analysis of Multivariate Manufacturing Data Sets

by

Mark Alan Rawizza

S.B., Massachusetts Institute of Technology (1994)

Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

March 12, 1996

Author ....
Department of Electrical Engineering and Computer Science
March 12, 1996

Certified by . ...
David H. Staelin
Professor of Electrical Engineering
Thesis Supervisor

Accepted by ... .
Frederic R. Morgenthaler
Chair, Department Committee on Graduate Students

Time-Series Analysis of Multivariate Manufacturing Data Sets

by

Mark Alan Rawizza

Submitted to the Department of Electrical Engineering and Computer Science
on March 12, 1996 in partial fulfillment of the
requirements for the degreee of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

The purpose of this thesis is twofold. First, data analysis methods used extensively in engineering disciplines are presented and applied to several different types of manufacturing data. Second, this thesis is intended to help bridge the gap that exists between the data analysis methods used in engineering and those used today in manufacturing. Time-series analysis, while rooted heavily in mathematics, is still fundamentally an artform. When to apply what types analysis to a particular set of data is not always clear. The introduction attempts to organize how a time-ordered data set should be studied in the context of manufacturing. Some fundamental preprocessing methods are shown to be essential for meaningful analysis. Since the processes studied in a manufacturing environment are multivariate, data reduction techniqes, such as principal components, are discussed. Improvments are made to the computation of principal components to compensate for the typically noisy environments encountered. Three important issues that analysis of time-series data should typically address in an industrial setting are detecting significant changes in a process over time, assessing the predictability of the data, and simply summarizing the time-series nature of a process. Each of these are addressed in the context of an actual manufacturing process. Hybridization of ARMA models with neural networks are shown to provide improvements in predictive power. Determining when and if a significant change in a process has occurred is extremely important to a process engineer. These changes or "surprises" can be identified using predictive models or by using in-quadrature filtering. Fourier analysis is used in unique settings to highlight patterns in a time-series via its power spectrum. Physical processes encountered in industry are very often characterized by periodically occurring patterns. An *generalized autocorrelator* is presented which can deal with assessing the predictive potential of a set of data which is sampled non-uniformly in time. In general, an attempt was made in this thesis to compromise between providing mathematical detail and presenting a wide range of different ideas found to be successful in practice.

Acknowledgments

I have completely loved my experiences at MIT over the past 6 years. Since this thesis is symbolic of the end of this exciting part of my life I would like to express my gratitude to some of the incredible people who were with me along the way. I am extremely fortunate to have been able to work with Professor Staelin. There is so much I have learned from him about how to think like an engineer. He was always supportative, helpful and generous with his time. I greatly appreciate and admire his optimistic viewpoint. No matter how bad some results may have looked he always strove to find the good side of things. Rizwan Koita helped initiate me into my graduate school years and was a source of ideas in research. I am thoroughly indebted to my fellow graduate students. In the beginning there was Michael Schwartz, who has always been more than willing to help out with problems and lend his ideas. I appreciate my many a long night working with him in the computer lab. I wish the best for him, his wife and his new baby daughter. Carlos Cabrera has been an incredible guy to bounce ideas off of and he has greatly helped me solve technical problems that I invariably encountered. Our mutual appreciation of classical music has been known to fill the computer room with interesting humming renditions of Beethoven's Seventh. And what would we do without Bill Blackwell. I am thankful for his time spent in solving the never ending stream of computer related problems. I forgive him for running his 2,000 hour long jobs and slowing the computers down to a crawl. It was great having him keep up a steady supply of cookies to the computer room. It was a pleasure to work closely with Tim Derksen on the same data sets. Our thesis complement each other. It was also great to be able to interact with Ambrose Slone, Michelle Spina, and Dr. Phil Rosenkranz. Felicia Brady has had the not always super-encouraging job of making sure I met my deadlines on time. I want to thank her for keeping me honest and for preparing the monthly RG4 newsletters.

I also want to thank some of my awesome friends who helped me through the tough times. Adam Hoyhtya and my graduate school careers paralleled each others in both classes and research. He is a great friend and helped me to finish my thesis on those last few crucial days. I love and appreciate so much: Jude, Cedric Logan, Will DeShazer, Will Potter, Marlon, Nicolas, Michael Metzger, Mike Hrnicek, Jesse Tauriac and Judi White, Piper, Natashya, Lynn Jean-Denis, Maria Sisneros, Zahra, Pete, Dan Zachary, Howard Loree, Lisa Chou, Susan Park, Aaron Cardenas, Roy and Chelly Larson, Sajjan and Lisa, Graham Morehead, John Epps, Venecia, Manish Goyal, Jose Luis Elizondo, Jim Ryan, Dean Farmer, and Danielle Dunlop. They are my best friends.

I want to hold up and honor my family with all of my heart. My mom and dad have unconditionally and completely supported me in every way. I appreciate my mom always making sure everything is in on time and pushing me when I would rather procrastinate. My dad always made sure I was eating right and checked that all of my needs were met and kept my mom from getting to excited about 'crises'. I feel blessed above all people for my mother and father. And I love my little sister Holly. We are both graduating this year together and I am so proud of her. I want her to excel in everything she does. I am greatly in debt to my Grandpa and Grandma Rawizza. They have been responsible for funding a huge portion of my education. And I thank my Grandma St. Jean and Aunt Marie for their encouragement throughout the years.

I also want to honor God who has blessed me continually. HE has helped me to keep perspective on what is most valuable in life.

"And we know that in all things God works for the good of those who love him."

– ROMANS 8[28]

This research was done in the Research Laboratory of Electronics (R.L.E) at M.I.T. and was supported by funds from Research Group 4 of the Leaders for Manufacturing Program at M.I.T.

Contents

List of Figures

Part I

Building a Toolkit

Chapter 1

Introduction

1.1 Background

The latter half of the 20th century is unique in history because of the way man accesses and uses information. Before the explosion of the electronics and communication industries in the last 40 years society was largely preoccupied with the relatively difficult and expensive task of gathering information. The energy expended in disseminating information was comparable to the energy needed in collecting it. The advent of powerful computers and rapid communication has created a new need for ways to organize and understand massive amounts of data, which is becoming easier and cheaper to obtain.

In fact, the incredible availability of information has far outpaced man's ability to digest it. The increasing thickness of newspapers, round the clock new channels, and the world wide web are all indicators of this fact. An interesting phenomenon has arisen. Instead of collecting data in order to answer specific questions, huge data sets are being used to actually inspire questions. The recent and fascinating topic of data mining for example uses massive amounts of data and computation power to look for patterns and special relationships in the data. This type of analysis, unfeasible in the past, has uncovered trends and idiosyncrasies that leave one wondering what else is there.

While almost every aspect of modern society has the potential for reaping large benefits from all the information available, very few have mastered the techniques to do so. Some techniques for examining and interpreting data-rich environments in the context of manufacturing are explored in this paper. In particualar, a suite of mathematical tools are developed which have proven to be very effective in studying the scientific types of data generated from a manufacturing process. These tools are then applied to actual manufacturing data sets to demonstrate their effectiveness in both generating questions and even more importantly for answering them.

1.2 Definition of Problem

This thesis will consider how to effectively analyze data in the context of large-scale manufacturing processes. These processes are data-rich in the sense that we have an abundant amount of information about a particular process under study.

Data, in this thesis, is composed of individual units called variables. Examples might be temperature, pressure, velocity, density, or viscosity. Together these variables convey information about the state of a process. There are two principal ways information is contained in each variable. One way is obvious the other is slightly less obvious. First each variable contains static information. This information is simply a recorded value. This can be seen by making a one-dimensional plot of a data vector. Simply place an x for every data value. Certainly these data values give us insight into the process under study. The parameter shown in Figure 1.1 shows data that is bifurcated. Sometimes the parameter is distributed around a mean of about 5 and at other times it is distributed around a mean of about -10 with slightly less variation.

However, in most circumstances variables will also have dynamic information. In other words, information is contained not only in the value of a variable but also in how that variable changes over time. This can be seen by expanding the variable to 2 dimensions: on one dimension is the value of a variable on the other is the ordering of that variable. Figure 1.2 shows the same variable as shown in Figure 1.1 but now it is plotted as a function

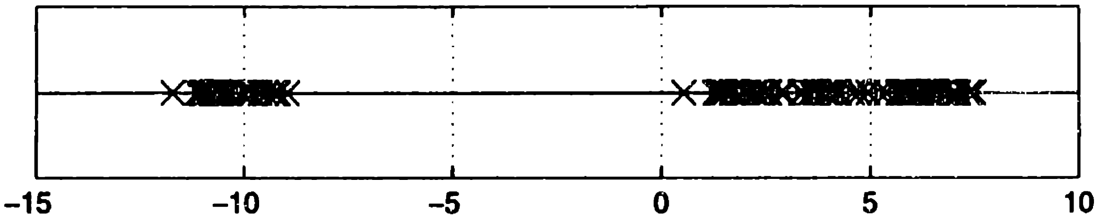


Figure 1.1: One Dimensional

of time. Data represented as a function of time can be referred to as a *signal* or as a *time-series*. These very simple plots give a basic motivation for studying time-series. Simply put, Figure 1.2 provides a lot more information about a parameter than does Figure 1.1.

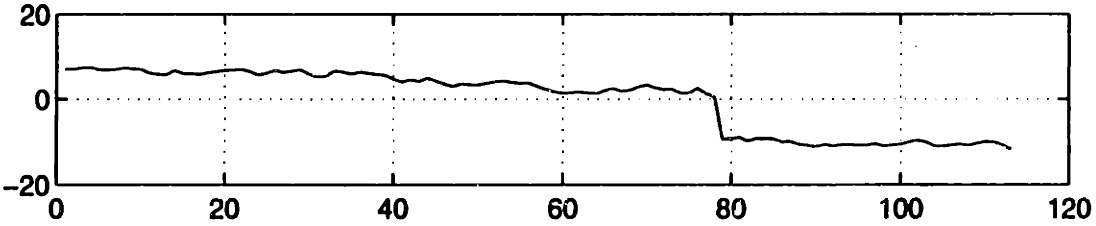


Figure 1.2: Two Dimensional

So in looking at data we want to consider both its static and dynamic qualities. This thesis will consider both, but particular emphasis will be on the fundamentally much more complicated dynamic or time-series analysis of data. A single time-series or signal can potentially contain many separate pieces of information. A major issue electrical engineers working in signal processing face is how to extract relevant parts of a signal that is embedded in other signals and in noise. Now add to this single information rich signal multiple different signals and obtain the types of data sets dealt with in this thesis. Reconciling multiple time-series with one another and with the underlying process is a formidable problem and the problem which this paper will address.

A simple problem statement is how to most effectively study the dynamic behavior of a physical process for which we have a large amount of data. Hopefully the data contains a large number of meaningful variables. However, it is not necessary to have an understanding of the mechanism by which the data is generated. We assume that the process under study is too complicated or costly to try and model.

1.3 Thesis Outline: Approach to Solving Problem

One of the major goals of this paper is to provide a general framework by which large multivariate ordered data sets can be systematically analyzed. Figure 1.3 is a flow chart showing the general approach to be taken in dealing with ordered data sets.

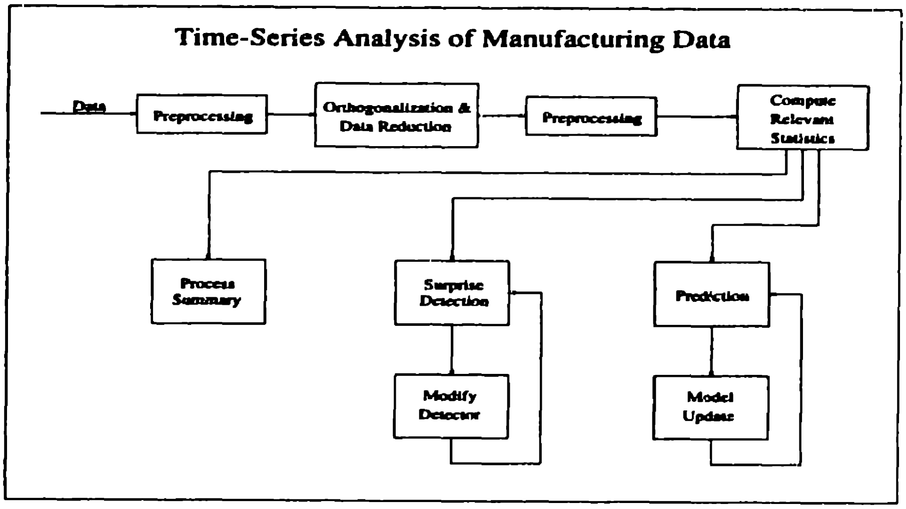

Time-Series Analysis of Manufacturing Data
Data
Preprocessing
Orthogonalization & Data Reduction
Preprocessing
Compute Relevant Statistics
Process Summary
Surprise Detection
Prediction
Modify Detector
Model Update

Figure 1.3: Flow Chart

This thesis can be broken up into 2 major parts. In the first part a set of powerful tools will be explained which will be used in the analysis of data. These tools were selected based primarily on their effectiveness working with actual data not on their novelty. Principal Component Analysis, for instance, is a commonly used method for linearly transforming a large multivariate data set into a new set of variables. It has proven to be extremely effective in digging out the important information in a large multivariate data set. Therefore we use it. The question is how to combine and modify these tools to yield the most fruitful results. Once a potent box of tools has been established the second part of this thesis will apply these tools to actual data sets. These data sets will represent a wide spectrum of possible data found in practice. The inferences drawn from the analysis in this section are geared towards manufacturing processes. However, for people wondering if they should read this thesis, the general methods discussed should apply to most type of ordered data sets. More specifically, it should apply particularly well to data that is constrained by an underlying mechanism which obeys physical laws.

Chapter 2 discusses the importance of preprocessing the data. Some essential preprocessing techniques are discussed with an emphasis being placed on how various methods influence subsequent analysis.

Chapter 3, Orthogonalization and Data Reduction, is very important when dealing with large multivariate data sets. The basic idea is how to take a huge set of data and reduce it into a smaller set which contains the most relevant information. This is relevant in graphical display of information, for instance. It is also relevant when attempting to train prediction models. It is just not feasible to work with very large numbers of input variables. Principal Component Analysis is an extremely effective ways for reducing data. It will be discussed in some detail.

The second major part of this thesis applies time-series techniques to actual data sets. Chapter's 2 and 3 are ways to condition the original raw data and get it into a form that is most useful for the analysis performed in Part II. The term *surprise* is defined and shown to be a very important characteristic to consider in manufacturing processes. A special filter-family is presented which is useful for noncausally detecting surprises over a wide range of time scales. Finally, the perhaps most sought after tools involving prediction are explained. Well organized combinations of ARIMA models and neural networks are shown to be very effective in predicting a data vector. Both univariate and multivariate models will be explored. Part II takes this arsenal of tools and applies them to a couple of manufacturing data sets. The capabilities of the tools will be demonstrated and give the reader an indication of the types of results to expect.

Chapter 2

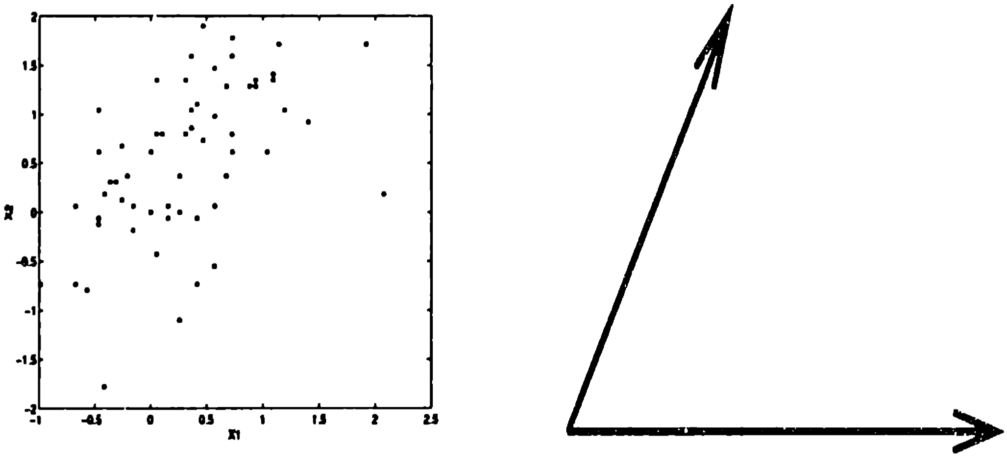
Preprocessing

2.1 Introduction

Experience has shown that preprocessing can make or break any subsequent analysis. Very rarely will just raw data values be used when working with large multivariate data sets. Ultimately the data will be used to train prediction models or as inputs to filters. As a result it is well worth the time to put forth some real effort to assemble a meaningful collection of data vectors. What this means will become clearer as the different preprocessing techniques are discussed. The procedures discussed are transformations, scaling, centering, and variable encoding.

2.2 Packaging the Data

Before delving into the most important preprocessing methods this brief section will explain exactly how raw data was prepared or packaged for all subsequent manipulations. Before any analysis begins the data is placed in a $n \times p$ matrix. The p columns correspond to each variable or parameter from which data is recorded and the n rows correspond to each observation of a particular variable ordered typically in time. Geometrically the matrix can be thought of in two important ways. One way is to think of the matrix as a p-dimensional vector space, $\mathcal{V}_{\mathsf{X}}^{p}$, where each variable represents a degree of freedom or a unique axis in this space. This will be called *variable space*. A variable space representation of data, often referred to as a scatterplot in two-dimensions, is an excellent way to visualize patterns in a data set and quickly assess correlations among variables, outliers, linear relationships, etc. However, in variable space the emphasis is on the individual observations and not on the variables themselves. This leads to a complementary way of thinking about the $n \times p$ matrix of observations in *subject space*, $\mathcal{S}_{\mathsf{X}}^{n}$. Each column of our matrix locates a point in n-dimensional space. Where in variable space each row of our matrix is perhaps best thought of as a point, in subject space each column should be thought of as a vector. These vectors have two very nice mathematical properties. First, the length of the vector is the variability of the corresponding variable. And second, the angle between vectors in subject space is directly related to correlations between variables. Vectors with a small angle between them are highly correlated whereas uncorrelated variables are at right angles or orthogonal to one another. To motivate future discussion note here that many analysis techniques benefit from having a set of variables which are orthogonal to one another.



(a) Variable Space, $\mathcal{V}_{\mathsf{X}}^{2}$ (b) Subject Space, $\mathcal{S}_{\mathsf{X}}^{60}$

Figure 2.1: Two Interpretations of 60×2 Data Matrix

2.3 Transformation

Hopefully the data we collect from a process is representative of the actual state of the process. Transformations can be used to enhance important information or to yield a set of numbers that have desirable mathematical properties. For instance, a certain variable may usually be measured at small values every half-hour but every 8 hours a very large value is recorded. Any prediction model given this data will likely be dominated by the effect of the large values and the relatively important half-hour variation in the data will be ignored. In this case some type of compressive mapping function may be useful to move the large values closer to the majority. This ensures that important information is not wiped out. Some methods of analysis require that the set of data satisfies some set of assumptions. If this is not the case a transformation can often be applied so that the data will satisfy the assumptions. The transformation can be applied to tne transformed data and, if desired, the results can be back-transformed so that conclusions can be based on the original units of measurement. Some analysis methods are optimal in a particular way for normally distributed data, for example. An appropriately selected transformation can map the original data to values that are more normally distributed. Important transformations include the log, square root, and inverse transformations. Analytical methods exist for determining appropriate transformations.

There are a family of univariate power transformations that can be applied to create a new set of observations that are nearly as normal as possible [6].

$$x^{(\lambda)} = \begin{cases} x^\lambda & \lambda \neq 0, \\ \ln x & \lambda = 0 \text{ and } x > 0. \end{cases}$$

This family was slightly modified by Box and Cox to avoid a discontinuity at $\lambda = 0$:

$$x^{(\lambda)} = \begin{cases} \frac{x^\lambda - 1}{\lambda} & \lambda \neq 0, \\ \ln x & \lambda = 0 \text{ and } x > 0. \end{cases}$$

This family is characterized by a parameter that can be calculated by a maximized value of the logarithm of a normal likelihood function (2.3), after maximizing it with respect to the population mean and variance parameters. Given a set of n observations the appropriate λ is the one that maxamizes

$$\mathcal{L}(\lambda) = -\frac{n}{2}\ln[\frac{1}{n}\sum_{i=1}^{n}(x_i^{(\lambda)} - \overline{x^{(\lambda)}})^2] + (\lambda - 1)\sum_{i=1}^{n}\ln x_i \tag{2.1}$$

where

$$\overline{x^{(\lambda)}} = \frac{1}{n}\sum_{i=1}^{n}x_i^{(\lambda)} = \frac{1}{n}\sum_{i=1}^{n}(\frac{x_i^{\lambda} - 1}{\lambda}). \tag{2.2}$$

This family of power transformations only works for positive values of x. But this can be overcome by simply adding a constant to each observation. An alternative way to deal with negative valued observations is to try odd power roots such as the cube root.

2.4 Scaling

Most of the analytical techniques used in analyzing data take into consideration variation. Information in a set of data is contained in the variation. The overall variation can be adjusted by scaling. When using multivariate techniques scaling must be considered in all cases. Ideally, variables should be scaled so that their variation represents their respective importance. Lacking better information, variables are usually scaled to equal variance. Scaling is a critical factor in most nonlinear models. Nonlinear models usually assume that variables with large variation are more important than variables with small variation. As a general rule, variables input to a multivariate model should have roughly comparable variances. There should be a very good reason for scaling data to have significantly different variances. How the individual vectors comprising a data matrix are scaled is critical in principal component analysis.

2.5 Centering

When dealing with large multivariate data sets one of the most common things to do is to automatically center the data, usually about its mean. While this is often a very good procedure to apply to data some real thought should be made as to whether centering is always appropriate. Centering appears to be harmless. Mathematically when the joint probability distribution for a set of random variables is centered the location of the distribution is moved but the shape and orientation of the distribution is unaffected. The problem is that there may be a difference between the observed mean and the true population mean of the data. In an actual set of data, deviations from the mean may be a significant source of information. If this is the case, then centering the data can be deleterious; the information content of the series is altered. Models built using the altered data will obviously be adversely affected. The point here is that centering should not be done blindly. Whether or not to center a set of data is an important decision.

One way to entirely avoid the problem presented in the above discussion is to pick a value that approximately centers a set of data and always use that value for every center operation on a particular vector. The objective is just to get data approximately around zero.

As a general rule centering and scaling will always be applied in some shape or form to the data in this paper. Neither of these operations alter the type of information contained in a signal that is critical to analysis. But even more importantly, normalization of data in some way is essential for good results.

2.6 Variable Encoding

Variables are not always numeric. Some variable may be qualitative or it may describe an event. For instance, different types of defects at the end-of-line process may be described qualitatively. Or a product may follow one of several different routes in its assembly. What is an effective way to represent defects or production routes? Some practical encoding methods are as follows:

- To represent a classification scheme that has some type of implied order relationship use one less variable than there are classes. Use all -1's to represent the lowest class and all 1's to represent the highest class. Flip variables from -1 to 1 as the different classes are traversed. So assuming we have a serious defect, a moderate defect, and a minor defect, this can be encoded with two variables, d_1 and d_2. d_1 = -1 and d_2 = -1 represents the minor defect; d_1 = -1 and d_2 = 1 represents the moderate defect and d_1 = 1, d_2 = 1 represents the serious defect.

- To represent class information that has no implied order use one variable for each class. Set the variable corresponding to a particular class equal to 1 and all other variables equal to zero. So if we have a product coming off an assembly line which could have taken 4 different routes use 4 variables: x_1, x_2, x_3, x_4 and encode route 3, for example, to be x_1=0, x_2=0, x_3=1, x_4=0. Do not encode the 4 different routes using 4 values of one variable.

- To encode a rare event set a variable to one for this event and zero otherwise.

- New variables can be created as flags for special relationships between other variables. For instance, a multivariable data set may have data points that cluster into several distinct regions. A new set of variables could be used to flag which region a point is in.

Chapter 3

Orthogonalization and Data Reduction

3.1 Introduction

As the title implies, the goal of this chapter is two-fold. First, when dealing with large multivariate data sets some method of reducing the number of variables is needed. The prediction models implemented later are trained based on the input data given. Typically it is not feasible to train the models using a very large number of variables. One simple approach might be to simply select some small subset of variables and ignore the rest. Some engineering knowledge could be used to perhaps select a relevant subset of variables. The technique presented in this chapter, however, demonstrate a way in which information from all of the variables can be used to some degree in obtaining a reduced set of data. A second important consideration is orthogonalization. It is preferable to use data variables which are uncorrelated with one another. In this way variables can be fed to the training models each of which are unique up to their second moment. Orthogonalization produces a set of uncorrelated variables. The technique presented in this chapter, Principal Component Analysis (PCA), orthogonalizes the data and provides a convenient way to reduce the dimensionality of the data set.

3.2 Principal Component Analysis

Principal Component Analysis exploits relationships among variables. Given a multivariate data set one can be almost certain that correlations exist among the variables. Intuitively this can be seen in manufacturing processes by understanding that different variables are often constrained in some way to obey the laws of physics. If we have taken measurements on a particular process which include temperature, pressure, viscosity, and velocity there is likely to be physical relationships. Temperature and pressure are physical variables that are often related in some way to one another. If measurements were taken of some heated substance the tempterature is sure to influence the viscosity. The point is that given any large set of data correlations among variables are sure to exist. If p variables have correlations among themselves this implies that there is really less than p pieces of information contained in the variables. PCA is an excellent way to get an indication of the true dimensionality of a set of variables.

3.2.1 Mathematical and Geometrical Description

Most of the tools presented in this thesis are well rooted in a solid mathematical foundation. In examing these tools our primary focus will be on extracting their usefulness in practical application and in developing a working intuition of how they work. A list of excellent refrences for more rigorous study of these tools will be provided in the Bibliography. However, Principal Components will be looked at here in some degree of detail for several reasons. First, PC's are immensely useful in many types of multivariate analysis techniques. Second, the mathematics behind Principal Component is relatively easy to understand and it has a very nice geometrical interpretation. Third, a method to improve the information containing quality of Principal Components will be presented for which a basic mathematical prerequisite will be helpful.

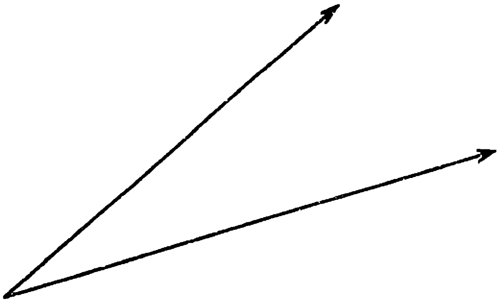


Figure 3.1: Correlated Data Points

It can be very useful to consider PCA geometrically. Consider again the interpretation of data as points in a p-dimensional space discussed in Section 2.2. A *linearly independent* set of p variables $X_1, X_2, X_3, ..., X_p$ span a space of p-dimensions. In this case, due to linear independence, the location of data points in one dimension give us no indication of the location of data points in another dimension. However, as is almost always the case, correlations exist among variables. This implies that our data vectors are concentrated more in some directions than in others. Principal Component Analysis takes advantage of these correlations. To start off with let's consider some simple examples in 2-dimensional subject space. Vectors representing the data points are shown by solid lines and the principal component is the dotted line.

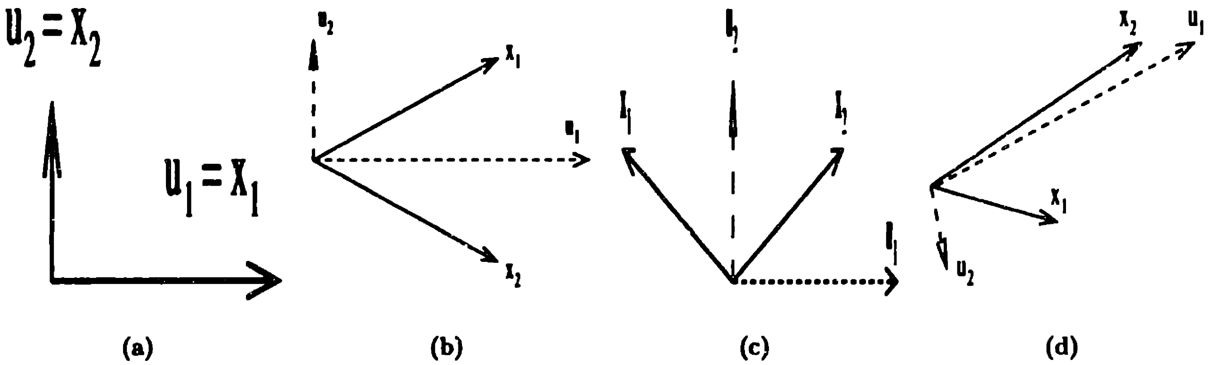


Figure 3.2: Some Examples of Principal Components in $\mathcal{S}_X^2$

In actual data sets not dominated by noise it is very often the case that variability is concentrated in a relatively small subspace of $\mathcal{S}_X$. PCA takes advantage of correlations among variables to find a new set of variables (or new vector directions in $\mathcal{S}_X$) which concentrate their variabililty into as few dimensions as possible. This new set of variables,

$U_1, U_2, U_3, ..., U_p$ is a linear combination of the original variables, $X_1, X_2, X_3, ..., X_p$:

$$\vec{u}_k = \theta_{1k}\vec{x}_1 + \theta_{2k}\vec{x}_2 + \cdots + \theta_{pk}\vec{x}_p \tag{3.1}$$

This can be also thought of as a dot product of the original variable matrix, X with an orthogonal matrix X.

$$U = X\Theta \tag{3.2}$$

The orthogonality condition implies that

$$\theta_{1k}^2 + \theta_{2k}^2 + \cdots + \theta_{pk}^2 = 1 \tag{3.3}$$

which will preserve distances.

The matrix Θ will transform the original data matrix. The following 4 properties constrain this transformation process:

1. The vectors $\vec{u}_k$ span the same space as the original vectors $\vec{x}_k$. In other words no information is lost in the transformation.

2. The total variability is the same between the two sets of variables

$$\text{var}(U_1) + \text{var}(U_2) + \cdots + \text{var}(U_p) = \text{var}(X_1) + \text{var}(X_2) + \cdots + \text{var}(X_p) \tag{3.4}$$

   Geometrically this means that the squared lengths are the same:

$$|\vec{u}_1|^2 + |\vec{u}_2|^2 + \cdots + |\vec{u}_p|^2 = |\vec{x}_1|^2 + |\vec{x}_2|^2 + \cdots + |\vec{x}_p|^2 \tag{3.5}$$

3. The transformed variables are uncorrelated. Geometrically this implies that the principal components are orthogonal as seen in Figure 3.2.

4. Each principal component captures as much of the variability of X_l as possible. So

principal component U_1 is chosen so that it has the maximum possible variance subject to the orthogonality constraint of Equation 3.3 (eg. $\vec{u}_1$ is as long as possible). U_2 is chosen so that $\vec{u}_2$ is as long as possible and subject to the contraint that it be orthogonal to $\vec{u}_1$. This same pattern is continued until we build up the full set of Principal Components.

3.2.2 Information Enhanced Principal Components

Principal Components are used extensively in this paper. In many cases, large multivariate data sets can be reduced down to a significantly smaller set of transformed variables which capture most of the variation in a process. The only potential problem is that variation is not necessarily equivalent to information. If many of the original variables are mostly noise this will find its way into some or all of the principal components. The ideas discussed in Section 2.4 can be applied here in attempt to make variation proportional to information content. This goal can realized if a signal-to-noise (SNR) is computed.

The SNR is computed using spectral techniques. Information in a signal is usually contained in a certain range of frequencies. Noise is typically represented at every frequency but there are usually bands of frequency which are only noise. These are typically found in flat regions of a spectrum. A ratio between signal and noise can be estimated using this spectrum. This is the SNR. Each of the original variables can be weighted by the SNR. When the Principal Components are calculated from the weighted data vectors they are more likely to contain variation that is meaningful.

So to summarize the calculation of the Information-Enhanced Principal Components:

1. Normalize the original data set, X, to zero mean, unit variance, $\tilde{X}$. Geometrically this is equivalent to setting the lengths of each variable in subject space to unit length and setting the origin to zero.

2. Calculate the power spectrum of each variable to estimate the SNR.

3. Weight each column of X by its corresponding SNR.

4. **Calculate the Principal Components on this adjusted data matrix, $\tilde{\mathsf{X}}$.**

Part II

Application to Real Data Sets

Chapter 4

Web Process

4.1 Introduction

The second part of this thesis applies the tools previously introduced to actual data sets. The analysis is "blind" to the mechanism generating data. Constraints imposed on measurements due to the underlying physics in manufacturing processes, for instance, are not factored into the analysis. Modeling decisions and inferences are made based purly on the data. Indeed one of the purposes of this thesis is to demonstrate how general techniques can be applied to a wide range of different processes. As a result, a complete and exhaustive analysis of the various data sets is not necessary. Each of the processes will be used to highlight the use of specific tools. Interesting features usually become apparent as the relationships between multivariate time sequenced data are geometrically and mathematically explored. Detecting, characterizing and modeling these features are the major goals of the subsequent analysis techniques.

The Web process will be used to demonstrate methods of time-series prediction. First, traditional ARMA models will be used as a benchmark for predictability. The main challenge in ARMA modeling is selecting the correct model to use. The autocorrelation function and frequency domain analysis of the time-series to be modeled will be used to facilitate making this selection. Finding a good ARMA model inevitability involves some iterations.

The predicted signal an ARMA model generates along with a shock series can be used to suggest new models to try. ARMA models will also be hybridized with Neural Networks in an attempt to improve results. Multivariate ARMA models involving the use of several time-series simultaneously, are explored as well. Finally, this chapter will close by considering pure neural network prediction. The ability of neural networks to fit to nonlinear and nonstationary processes will repiesent another method for prediction.

4.2 ARMA Modeling

A univariate parameter, x, from the Web process was selected for the subsequent analysis. A portion of this parameter is shown in Figure 4.1. Notice a high frequency component with a period of about 20 samples superposed on a gradual trend upward. The autocorrelation function and the power spectral density will help to formalize some of the important characteristics of the signal, x.

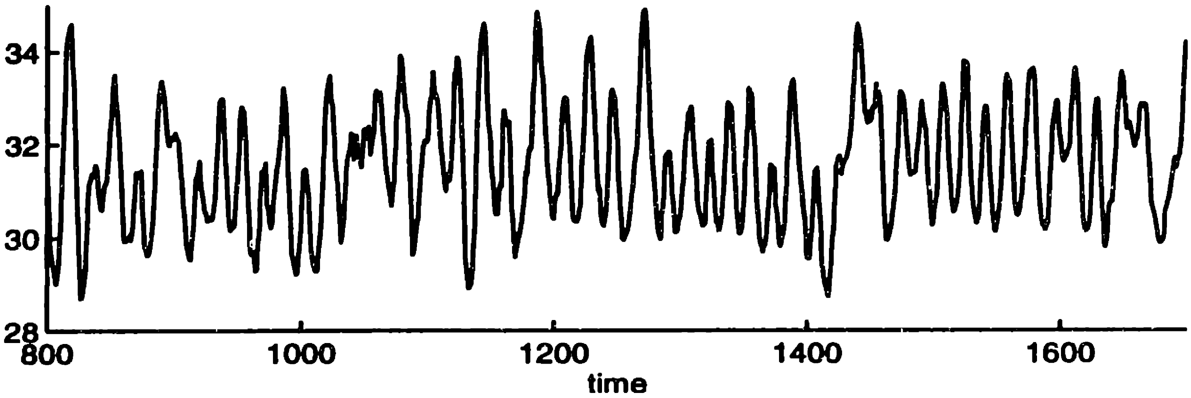
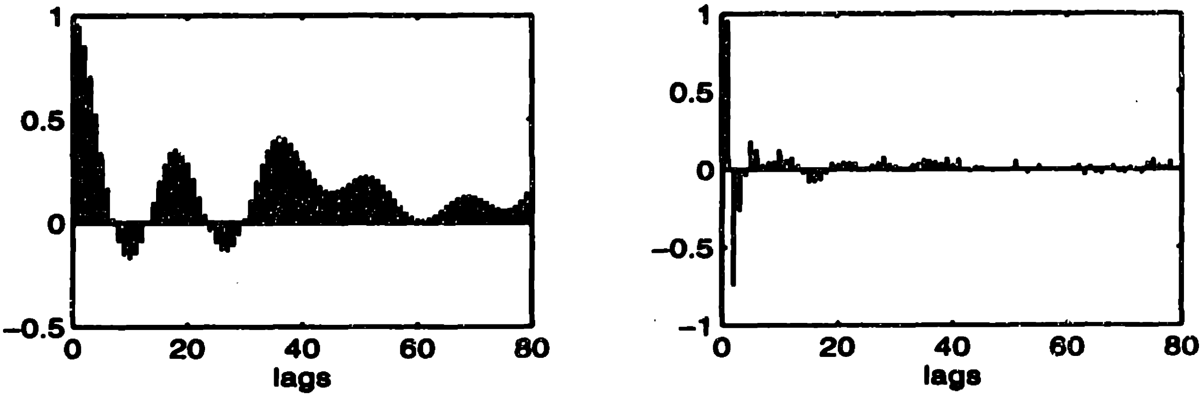


Figure 4.1: Original Signal

Effective ARMA modeling requires that the signal to be modeled is stationary, or can be made to look stationary through an appropriate transformation which can then be easily reversed to obtain back the original signal. The autocorrelation function, shown in Figure 4.2, is used to check whether this requirement is satisfied. In most normal cases a stationary series will drop quite rapidly to zero. In this case, the autocorrelation does drop to zero fairly rapidly but it has a very obvious periodic component. Each peak is separated by about 17 lags sugesting a periodic component in x with a period of 17 samples. Significant periodic components such as this one need to be removed from the signal.



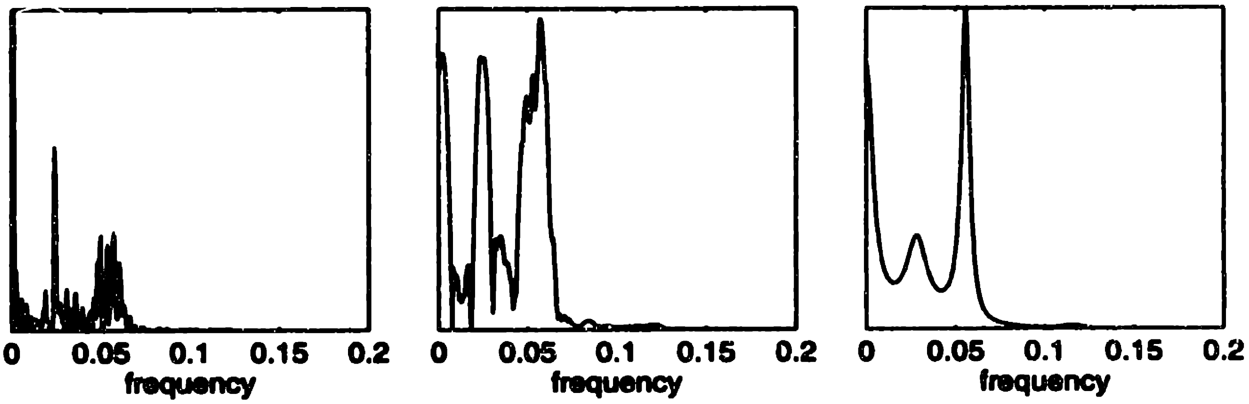
(a) Autocorrelation

(b) Partial Autocorrelation

Figure 4.2: Time Domain Correlation Analysis

The power spectrum verifies what was learned from the autocorrelation function. A period of 17 corresponds to a frequency of $1/17 = .0588$ which is the highest peak in both the smoothed spectrum and the maximum entropy spectrum. But there are some other prominent frequency components in the power spectrum that can't be easily deduced from the autocorrelation. There is a peak at about .023. This suggests a periodic component on the order of 43 samples.

As a first cut at modeling, the signal is differenced to produce, $\tilde{x}$. The differencing operation has the effect of a highpass filter which will kill low frequency terms as well as the DC constant offset. An ARMA model is trained on the first 1580 points of $\tilde{x} \mapsto \tilde{x}_T$. Training refers to the process of iteratively refining the ARMA coefficients so as to minimize a given cost function. The AR terms affect the prediction in a linear manner and can be solved for explicitly. But solving for the coefficients of the MA terms involves undoing a convolution. This means the MA terms have a nonlinear influence on the predictions. In addition, MA parameters interact with AR parameters. As a result the AR and MA parameters must be



(a) Raw Power Spectrum

(b) Smoothed Power Spectrum

(c) Maximum Entropy Spectrum

Figure 4.3: Spectral Analysis

optimized together in an iterative fashion. The cost function used in this analysis is the sum-squared prediction errors. Powell's method [8] is the algorithm implemented for this minimization.

The ARMA model selected used 20 lagged AR terms and 3 MA terms. This will be notated as ARMA(20,3). The training set can be created once the model is specified. For clarity ARMA(20,3) is a model of the form

$$x[n] = \phi_0+\phi_1 x[n-1]+\phi_2 x[n-2]+\cdots+\phi_{20} x[n-20]+\theta_1\varepsilon[n-1]+\cdots+\theta_3\varepsilon[n-3]+\varepsilon[n] \quad (4.1)$$

The model is trained and is validated using $x[1581]$ to $x[1640]$. A flow chart of this process is shown in Figure 4.4.

The results of applying the operations shown in the flow chart to the signal, x, is shown in Figure 4.5 as the ARMA(20,3) line. Figure 4.6 shows a plot of the magnitude of the residual between the actual signal and the predicted signal. The predicted signal, shown by the dotted line, tracks the actual signal well but it is offset by a DC component. This is due to the differencing operation. Results will be improved if the periodicity can be removed.

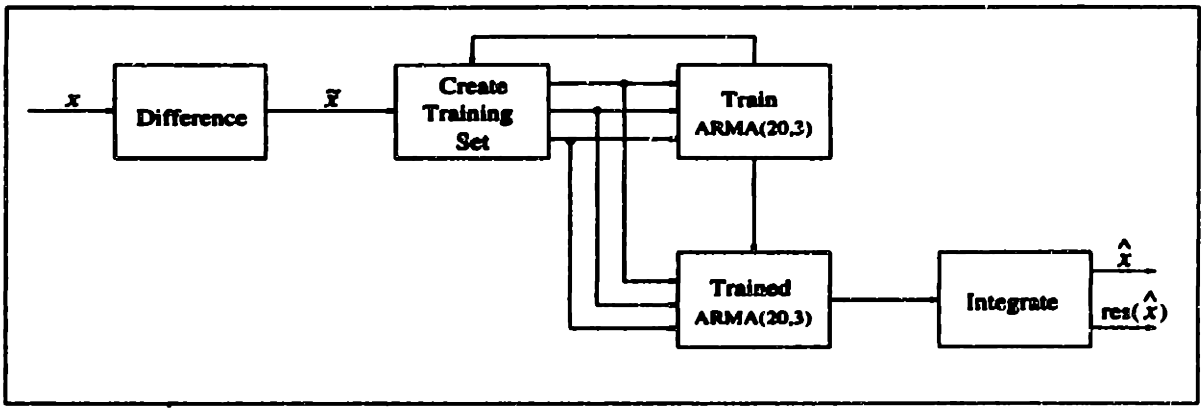


Figure 4.4: ARMA Prediction Flow Chart

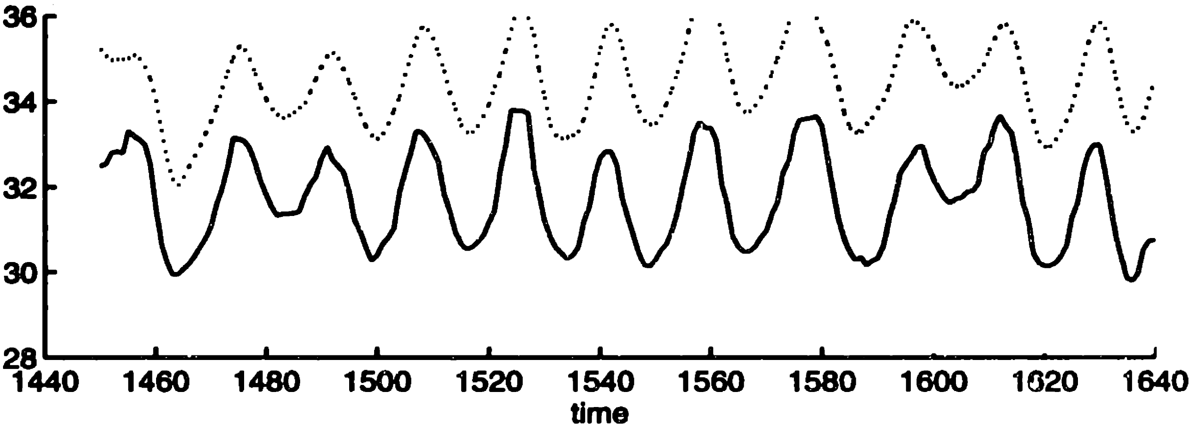


Figure 4.5: Prediction With Differencing Only and ARMA(20,3)

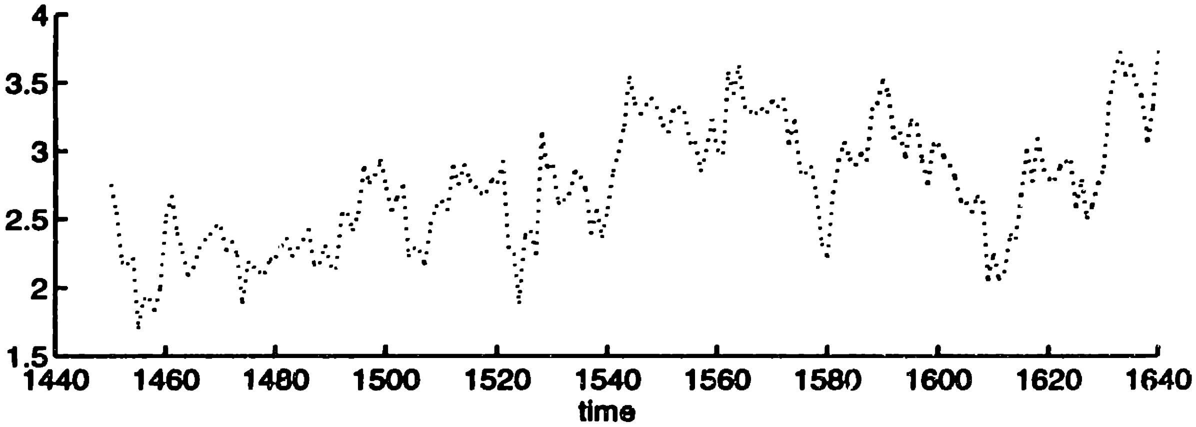


Figure 4.6: Residuals From Above Plot

The periodicity was removed by using a bandpass filter to filter out a center frequency of .0588 corresponding to about 17 samples. This signal was then subtracted from the original signal. An ARMA(20,3) was trained on this result. A neural network was trained on the periodic band-filtered signal. Then the two were combined to give the result shown in Figure 4.8. The improvement is dramatic.

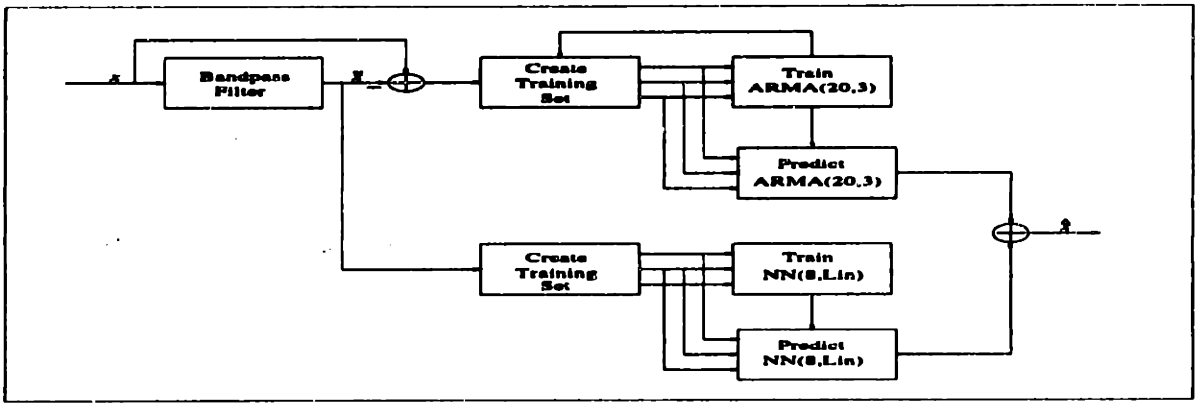


Figure 4.7: Flow Chart Showing Hybridization of ARMA with Neural Network

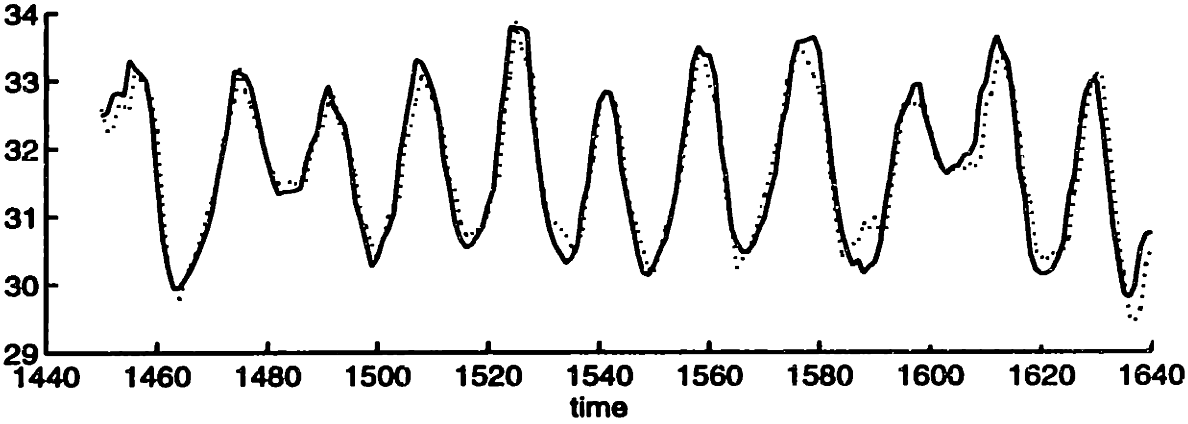


Figure 4.8: Predictions Separating Out Periodic Components

As one final test, the parameter was predicted using purely a neural network. The results indicate that a combination of neural network and an ARMA model is superior.

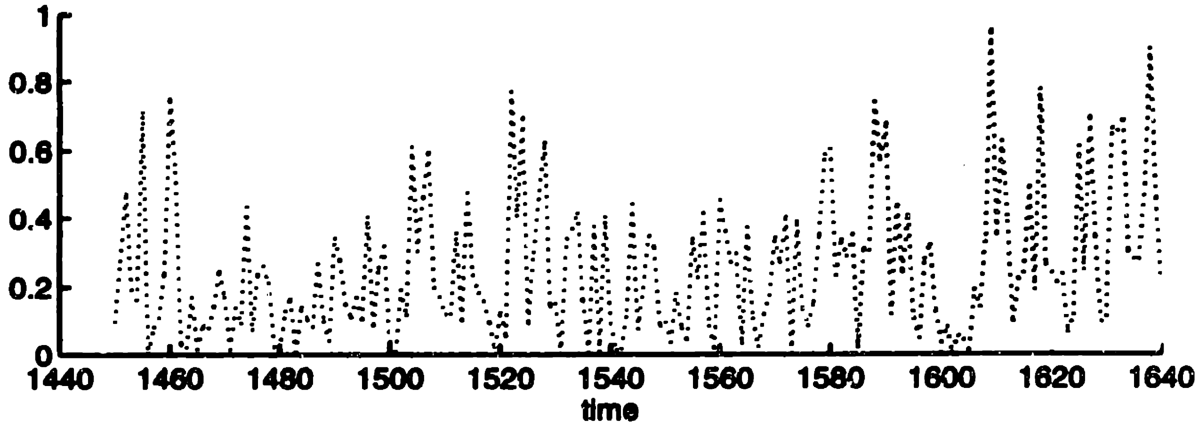


Figure 4.9: Residuals From Above Plot

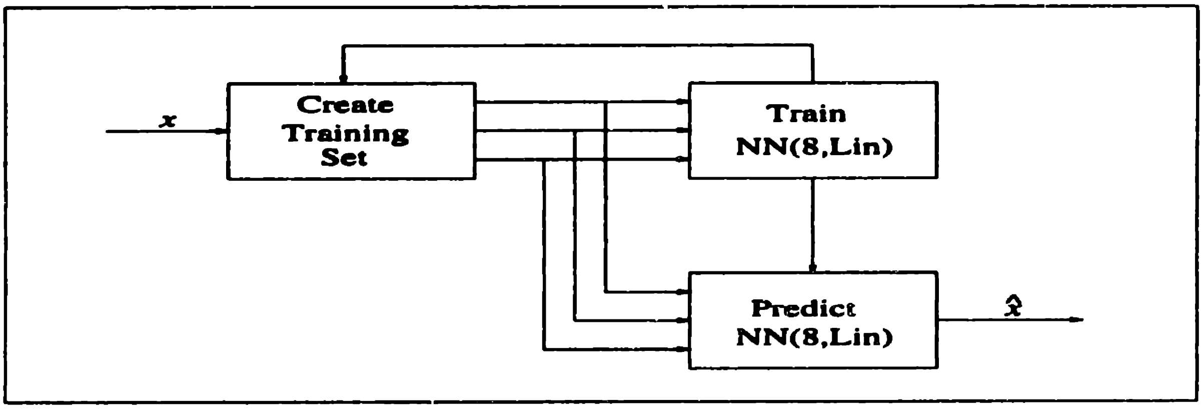


Figure 4.10: Flow Chart Using Only Neural Network

4.3 Defect Analysis

The output measurements made on the Web process are multivariate measurements indicating the size and location of a defect. The subsequent analysis will treat the defect variables as binary. At a given location or time there either is or is not a defect.

The ultimate goal, from a manufacturing point of view, is to relate inlying process variables to defects. However, two aspects of this data set make finding this relationship, if

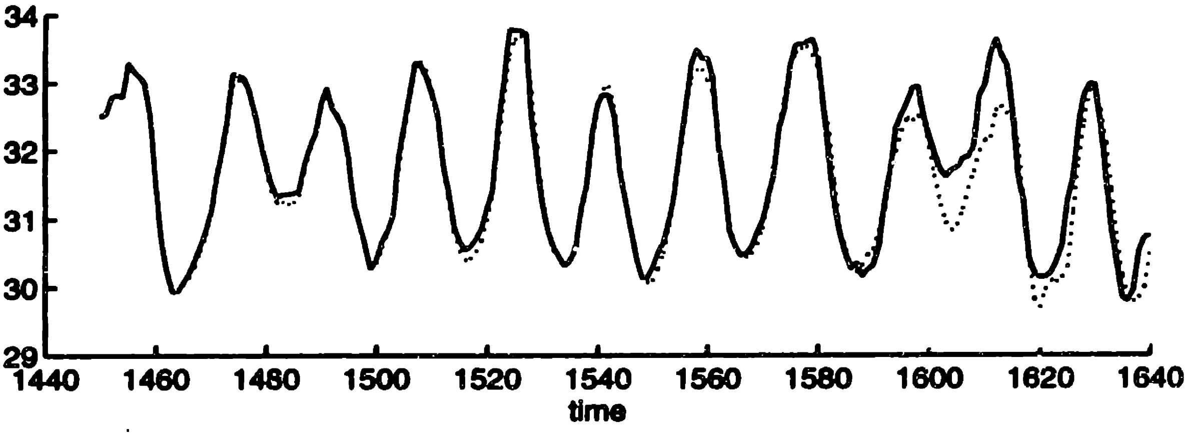


Figure 4.11: Predictions Using Neural Network

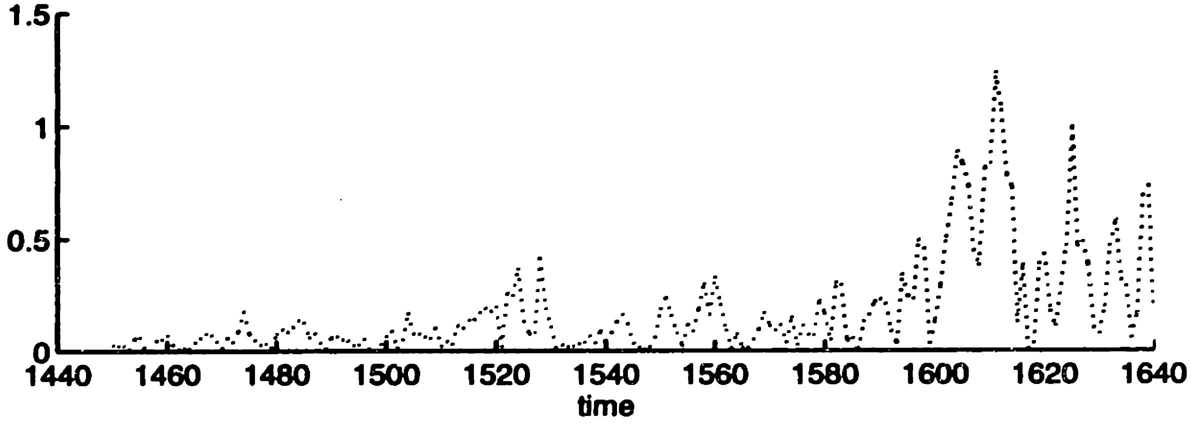


Figure 4.12: Residuals From Predictions Using Neural Network

it exists, difficult. The first problem lies in the discrete nature of the defects. Most input/output modelling techniques do not working effectively on analog input signals and discrete binary output signals. A first order solution to this problem is to consider correlations among outliers in the process data and defects. One might expect that inlying variables that deviate substantially from the mean to be responsible for defects in the output. With this particluar data set there were no significant outliers so this type of analysis is not possible. To generalize a little bit here, a good strategy is to try and find threshholds in the process variables which, when exceeded, result in defects in the output. These thresholds should be searched for in both a univariate and multivariate sense.

The second problem is more serious and potentially much more difficult to overcome.

The issue lies in the time-series nature of the data and manifests itself in two ways. First, there is not currently any way to precisely match up measurements at one particular point in the Web line with other points. So, for instance, a temperature measurement on the Web at one precise location and time cannot be reconciled with an end of line thickness measurement at the same point on the Web. While this is a real problem that shows up in many manufacturing process it is one that can be solved by combining appropriate data with some data crunching. Second, and more problematic, is the transient times between a variable change and its effect. Some parameters in the Web process may change but take many minutes or even hours before changes in the Web take place. A process that has many different variables with widely varying transients can become extremely challenging to model.

Given these two problems and given the fact that no first order relationships were found between the input variables and the defects the subsequent analysis focuses on the defects exclusively. Figure 4.13 shows the distribution of defects. This plot was created by laying 74 Web sheets on top of one another. The vertical axis is the longitudinal position along a sheet and the horizontal axis is the width across the sheet. The figure shows that a great many of the defects occur in vertical streaks. In particular a high density of defects occur in streaks along the left edge of the sheets. A clear streak also stands out around horizontal position 20 and two other streaks at around 27 and 28. This relatively simple plot indicates a great deal about the nature of the defects. It provides important information about where to look to reduce the number of defects in the process. The remaining defects appear to be randomly distributed.

In order to study the streaks in more depth, individual streaks are studied for patterns. The fourier transform is used to provide a frequency domain analysis of the defect signals. Proper analysis requires that the defects be placed on top of a well-structured support. This means that the time between defects needs to be divided up into uniform intervals. In its original form times are only recorded when a defect occurred. The signals constructed for analysis are binary and consists of a value of 1 at each time slot that a defect occurred and a value of 0 if nc defect occurred. These signals are on the order of 2 million points. To reiterate, vertical strips from Figure 4.13 are extracted and studied for patterns. Figure 4.14 to figure 4.21 show the defect signals as well as a frequency domain analysis of the signal. A 4096 point fast fourier transform was used in the calculation of the power spectral density.

The frequency domain analysis of the signals show that, except for Figure 4.16, each of the streaks shown has a periodicity at about 2.4023 Hz. Its harmonic also shows up at around 4.8 Hz. This value of 2.4023 Hz corresponds to a time of .4163 secouds. If the velocity of the web is known then this value can be used to estimate the distance between defects and aid in locating the source of the defect.

Defects that occur periodically are likely to be the result of something other than the process variables. If this is the case, as it almost surely is here, then the periodic defects should be removed before performing analysis for determining the relationship between process parameters and the defects.

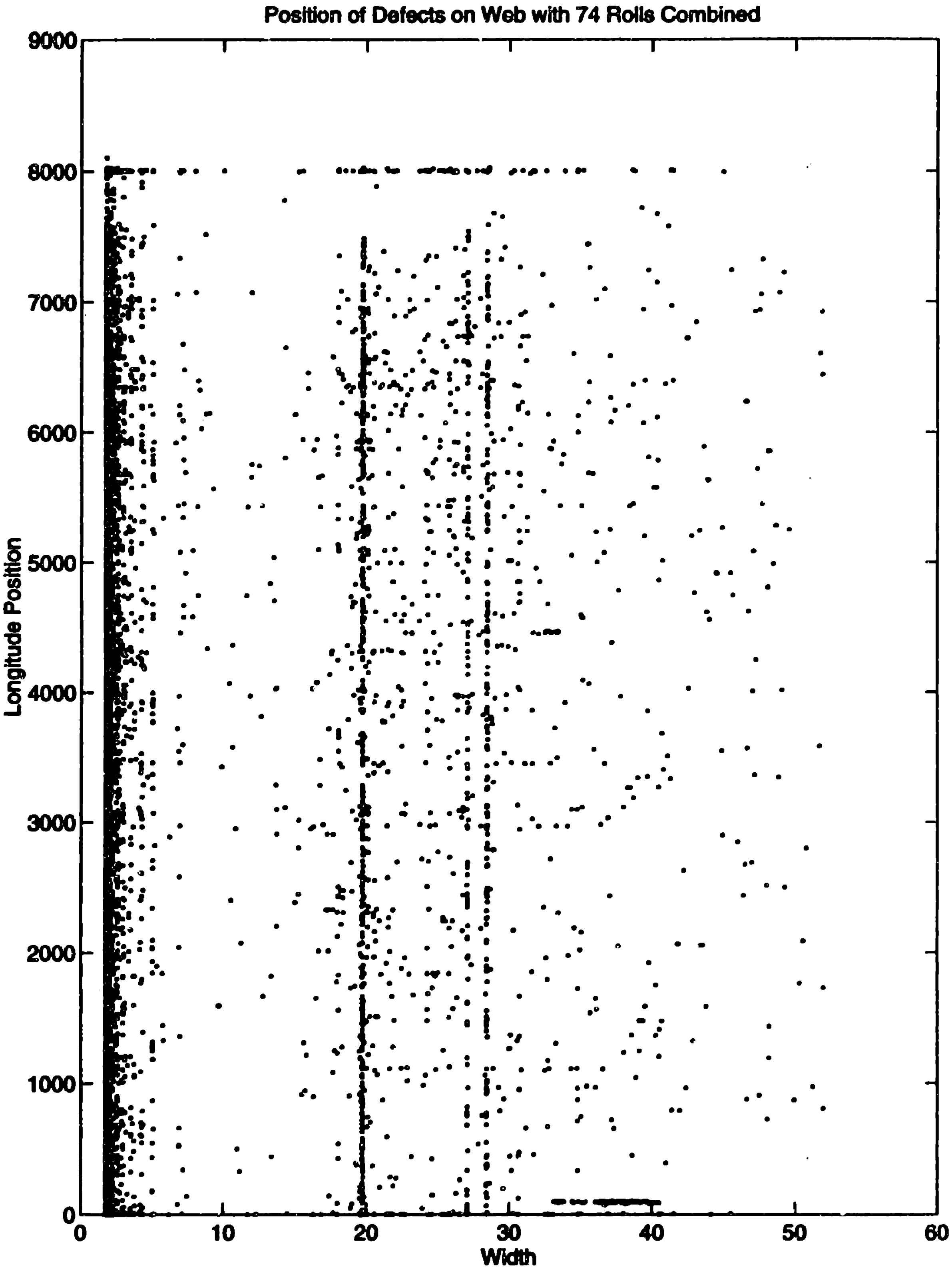


Figure 4.13: Location of Defects

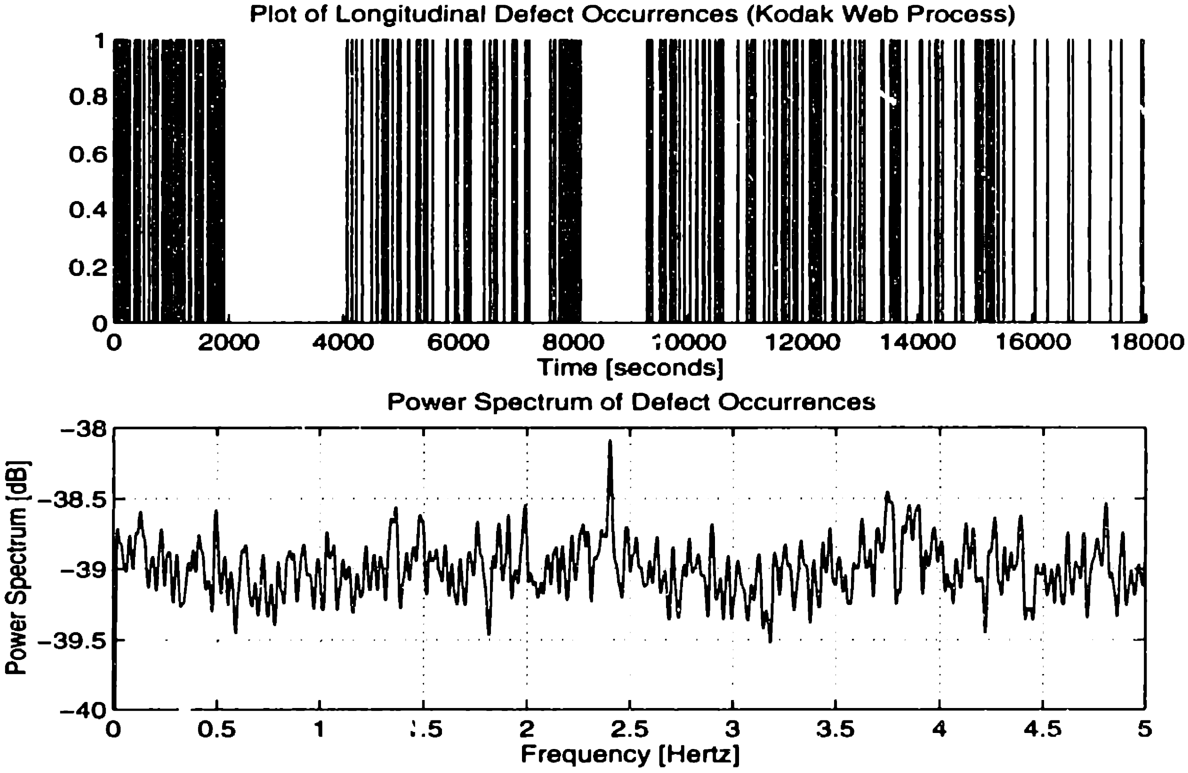


Figure 4.14: Defect Occurrences at x=19.7150 and Frequency Analysis

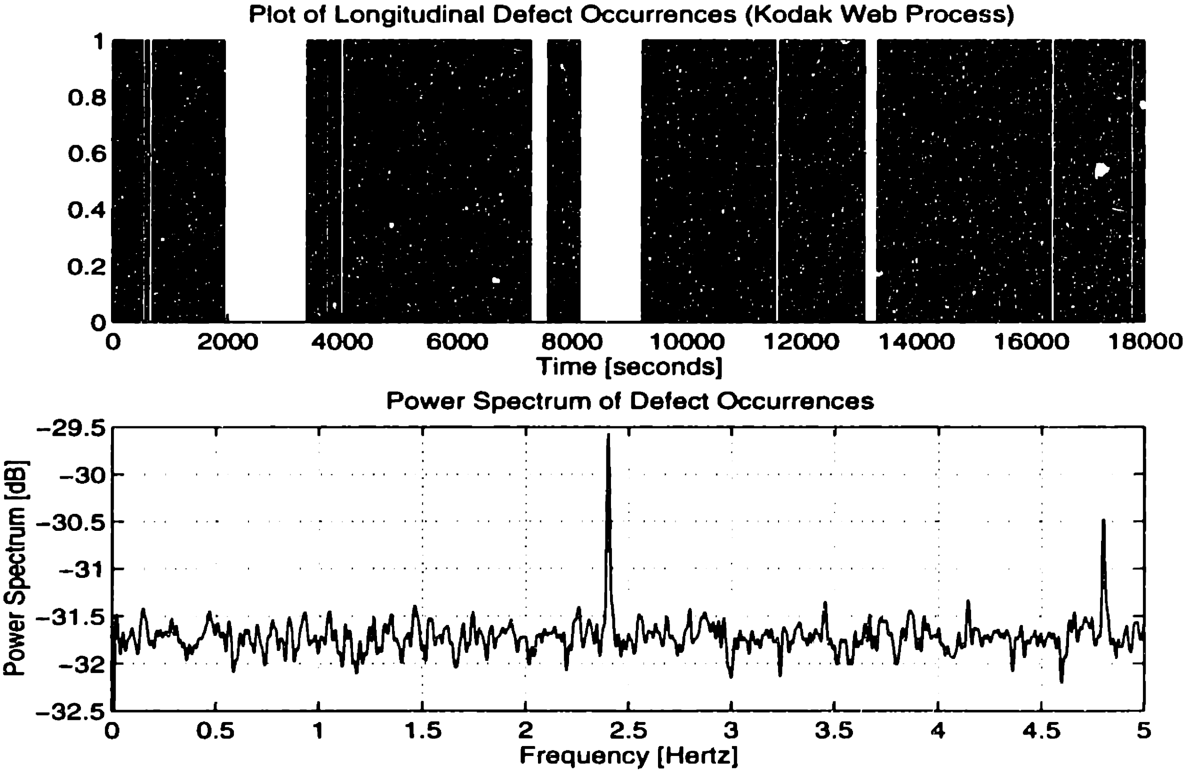


Figure 4.15: Defect Occurrences at x=1.77502 and Frequency Analysis

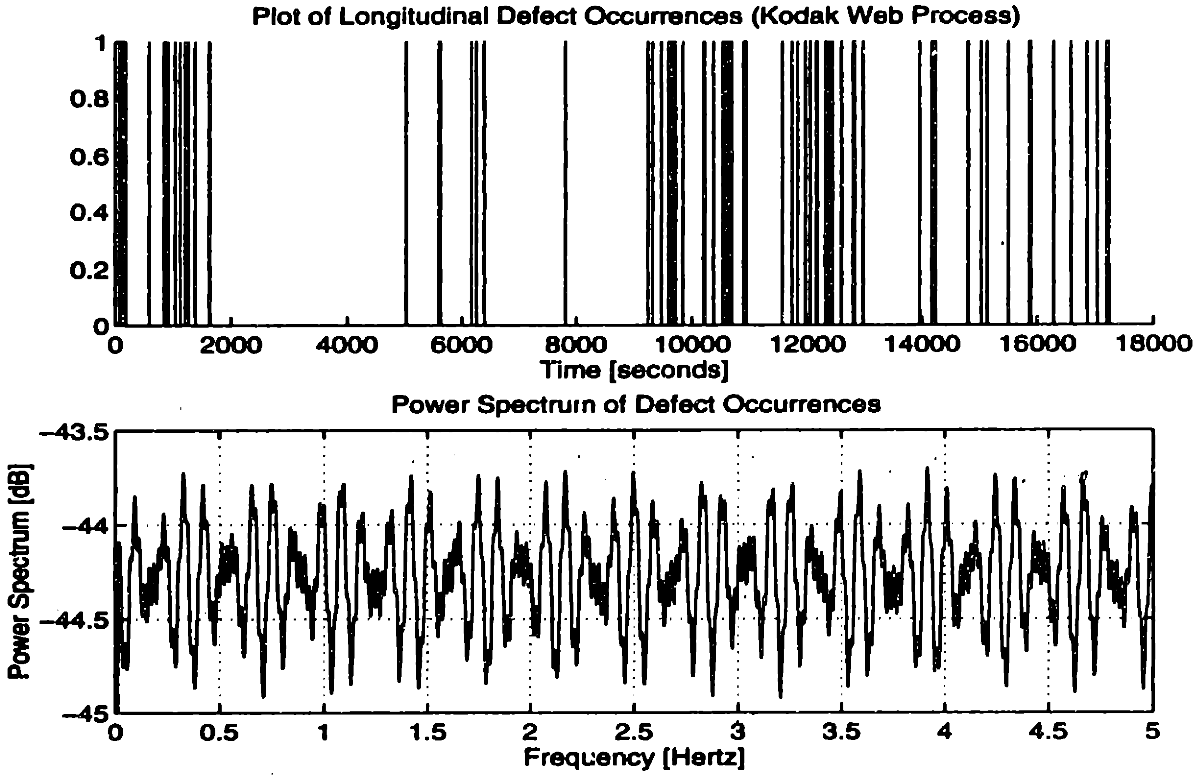


Figure 4.16: Defect Occurrences at x=28.4681 and Frequency Analysis

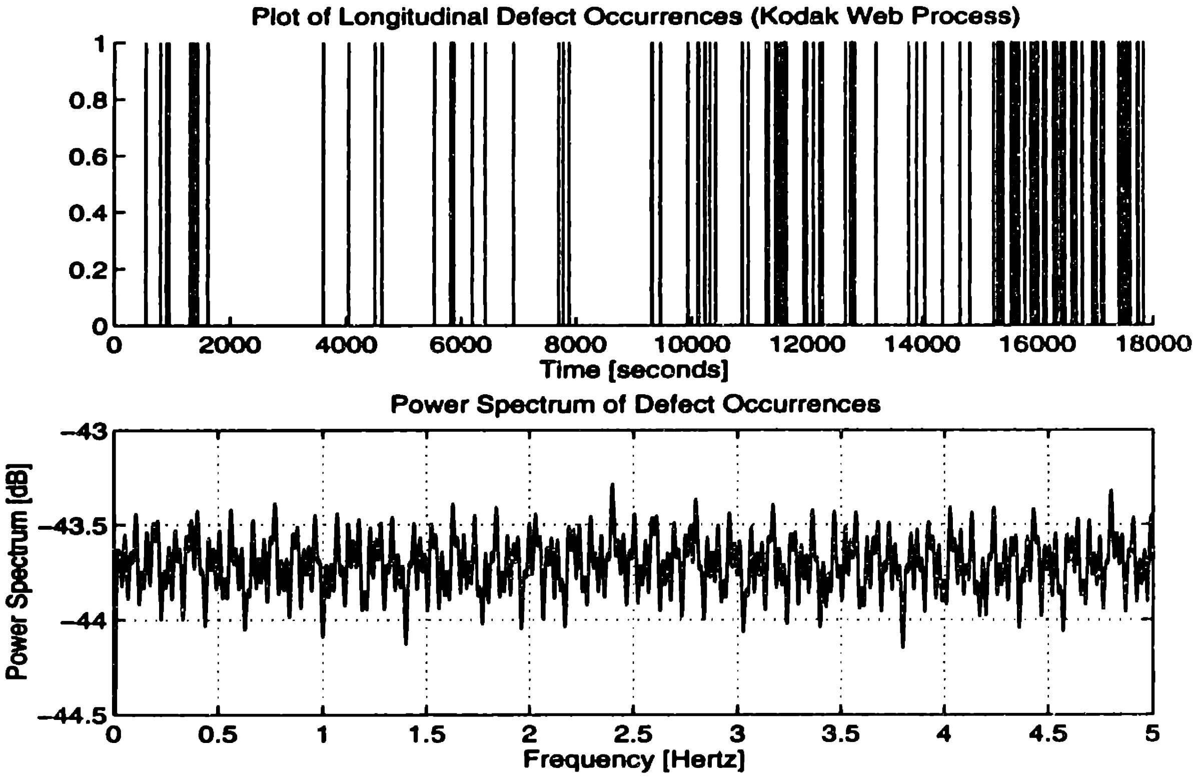


Figure 4.17: Defect Occurrences at x=28.4139 and Frequency Analysis

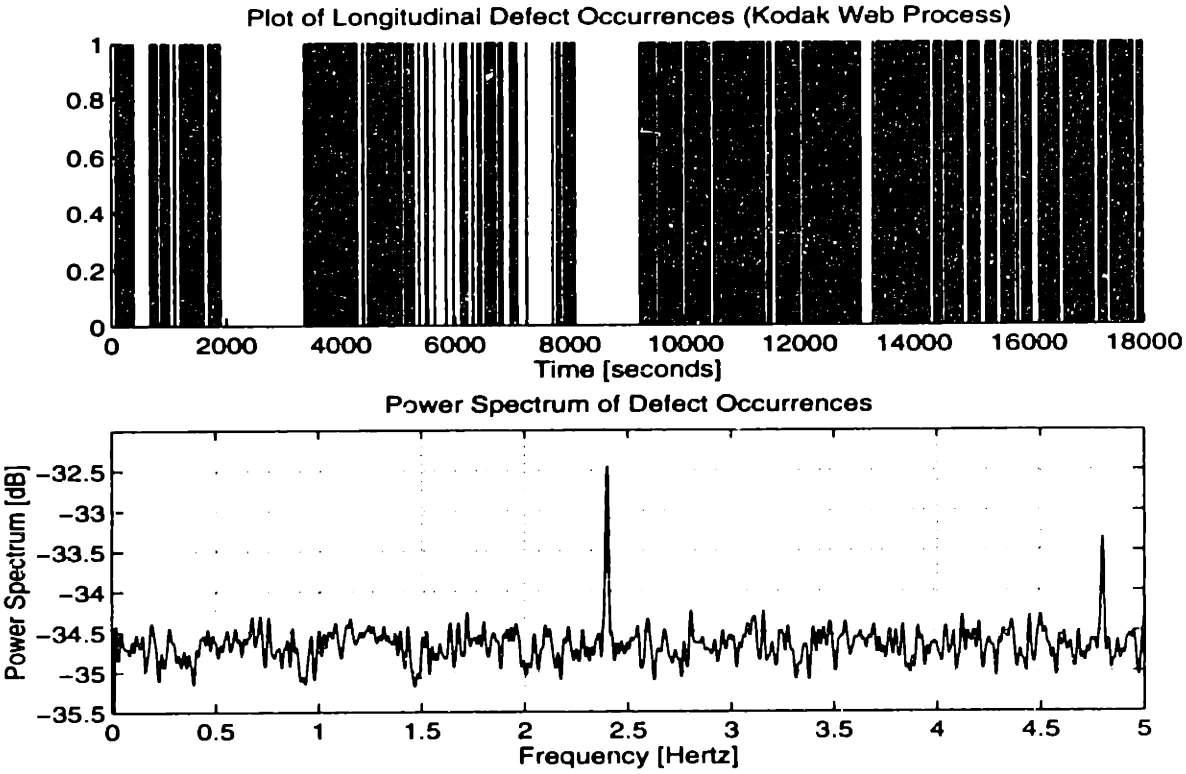


Figure 4.18: Defect Occurrences at x=1.74792 and Frequency Analysis

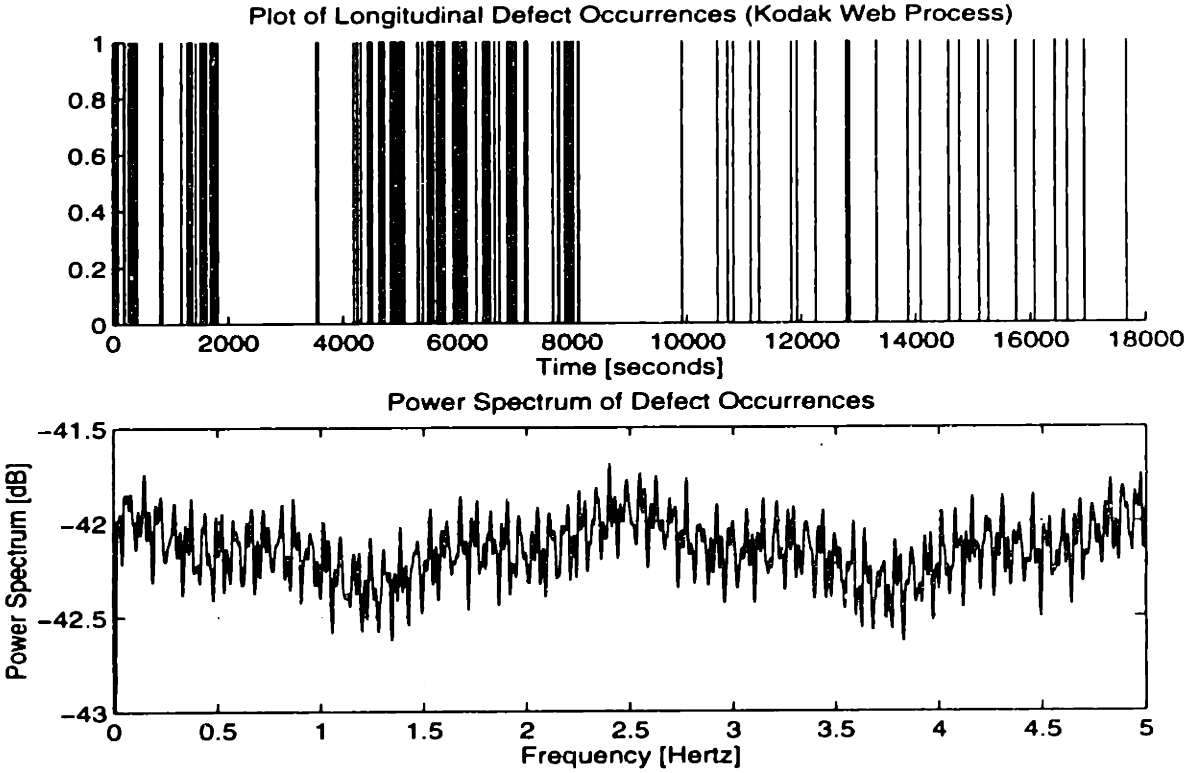


Figure 4.19: Defect Occurrences at x=1.80212 and Frequency Analysis

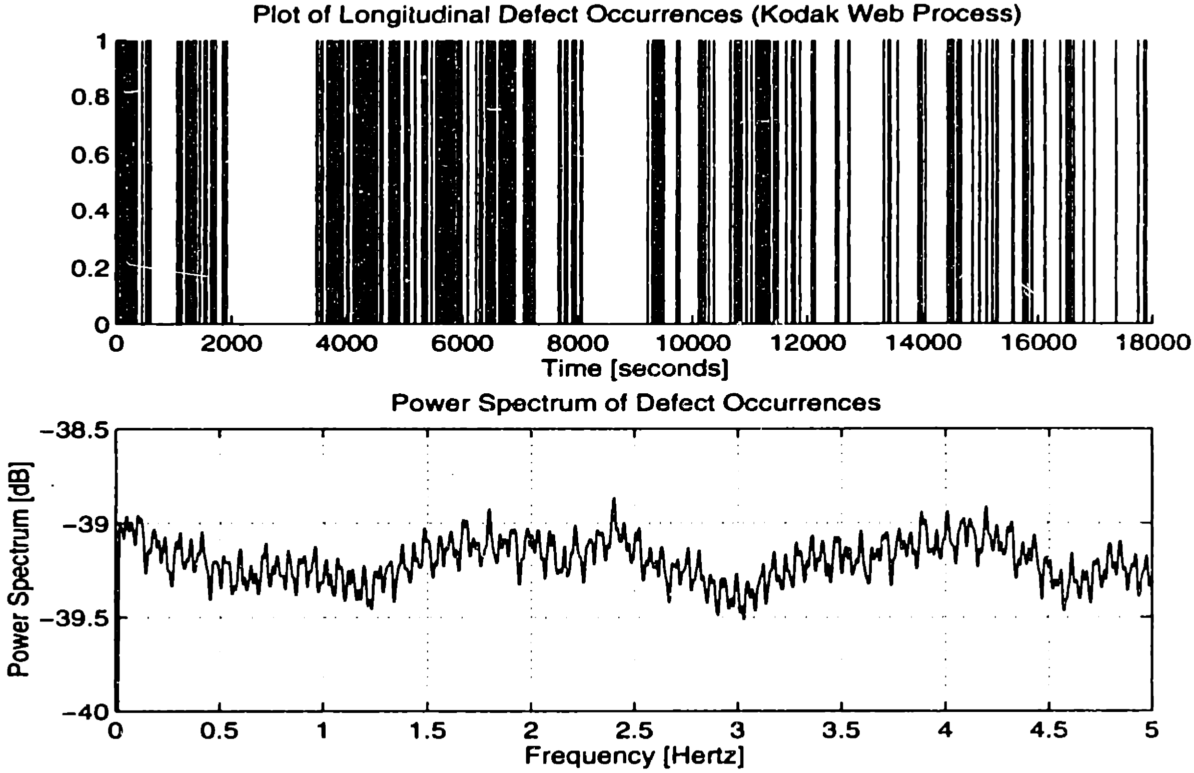


Figure 4.20: Defect Occurrences at x=2.01892 and Frequency Analysis

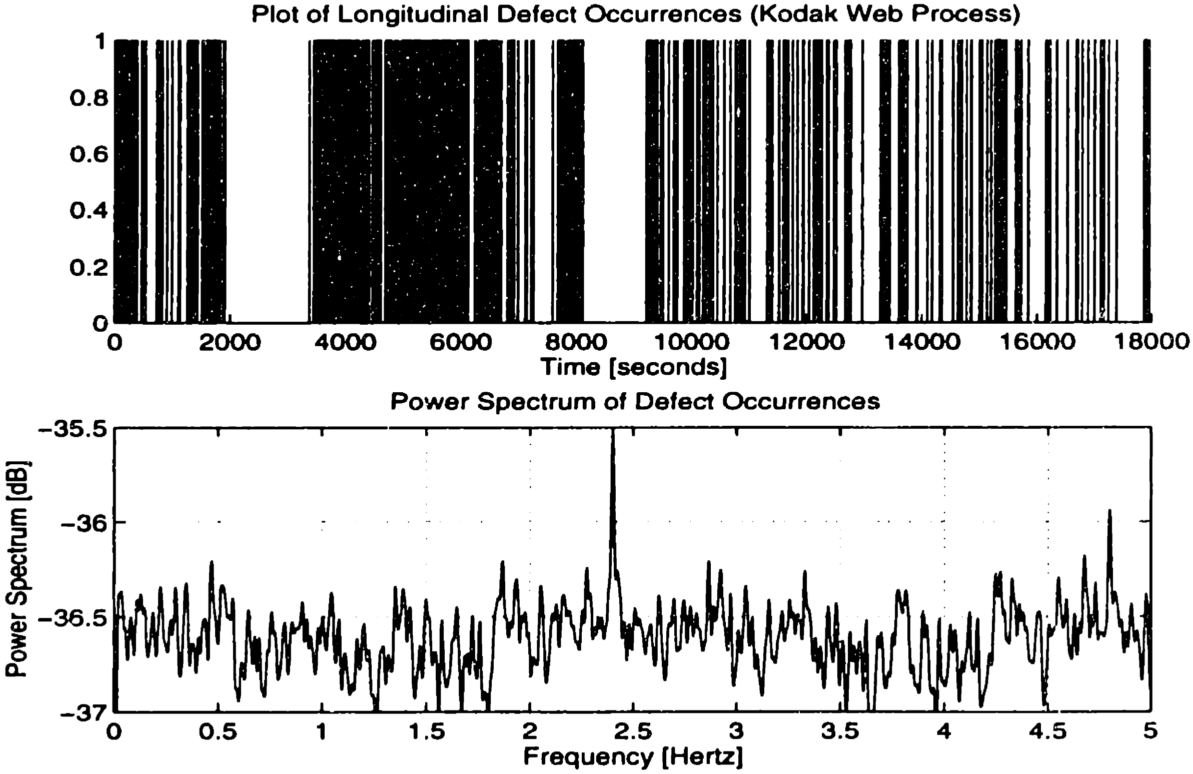


Figure 4.21: Defect Occurrences at x=1.99182 and Frequency Analysis

4.4 Multivariate Analysis of Inlying Process Parameters

For completeness, the process parameters are analyzed here. A subset of the full data set consisting of 854 variables is considered. Variables that had zero variation or variables that were perfectly correlated with one another were removed from the data to produce this subset.

4.4.1 Time-Series Plots of Principal Components

Figures 4.22 and 4.23 show the first 8 principal components plotted as a function of time. The first two principal components, indicative of a real physical process, drift slowly over time. The first 8 principal components are shown to highlight an interesting feature of this data as well as a point out a limitation of principal components. Figure 4.24 show zoomed in versions of PC3 and PC4. Notice the very obvious periodic nature of the data. While PC3 and PC4 appear to be similar recall that principal components are uncorrelated with one another. All of the first 8 principal components have this periodic nature.

Figure 4.24 also shows a plot of the power spectral density principal components 3 through 5. The power spectra are all very similar as the plot indicates. The strong similarity between the different principal components brings into question whether they are all truly representative of different physical phenemona.

If one information carrying signal were modulated onto a carrier signal that drifted in frequency the result would be multiple signals that are uncorrelated with one another.

Let

$$x_i(t) = \text{signal} * sin(2\pi f_i t) \tag{4.2}$$

where $x_i(t)$ is a column vector.

If f_i and f_j are integers and $f_i \neq f_j$ then:

$$x_i(t)' * x_j(t) = 0, \tag{4.3}$$

meaning that $x_i(t)$ and $x_j(t)$ are uncorrelation with one another.

The implication is that the first 8 principal components in the web process parameters could in reality be represented by only two pieces of information: (1) the underlying signal and (2) the modulating signal.

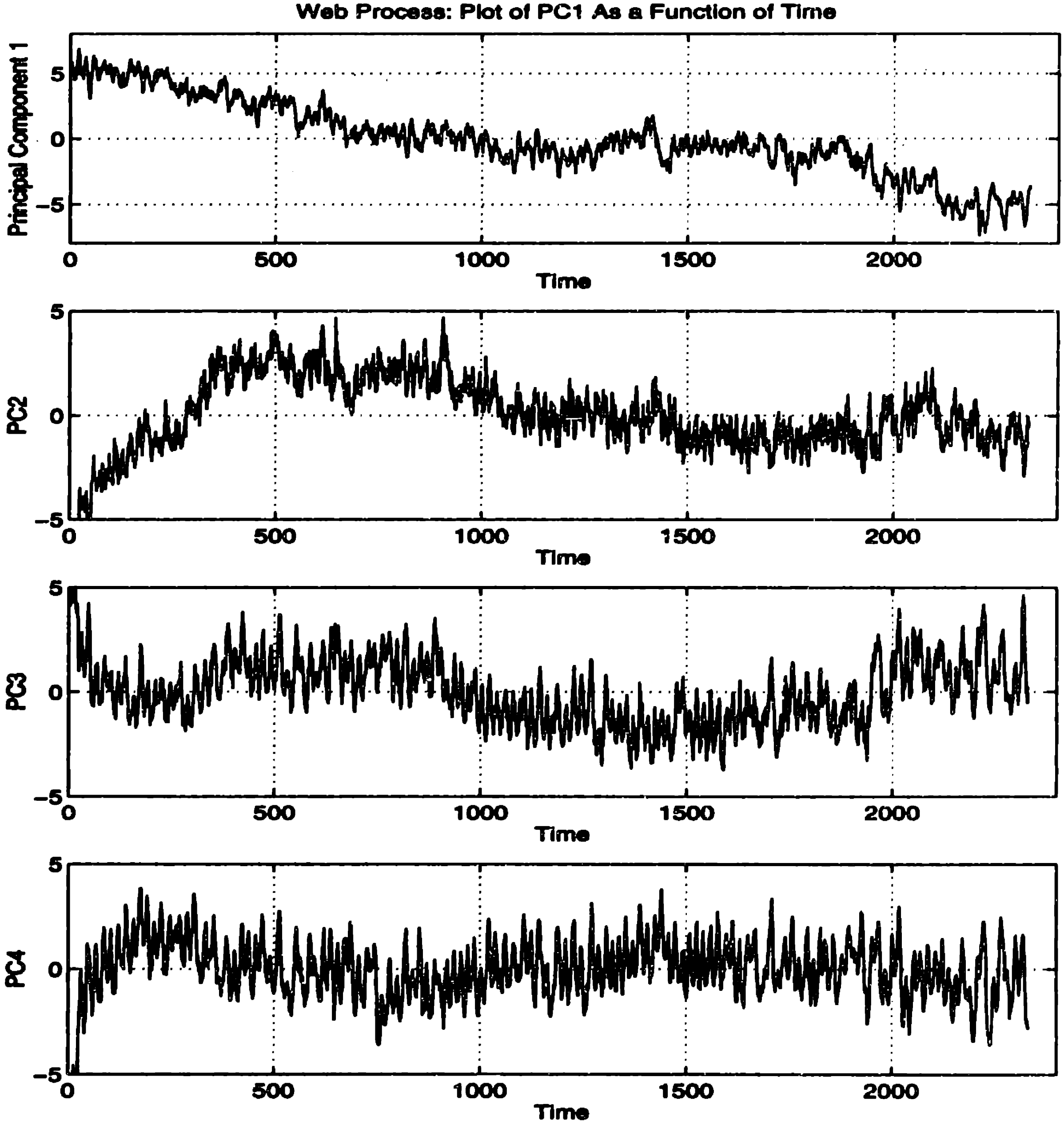


Figure 4.22: First Four Principal Components of Web Process Variables

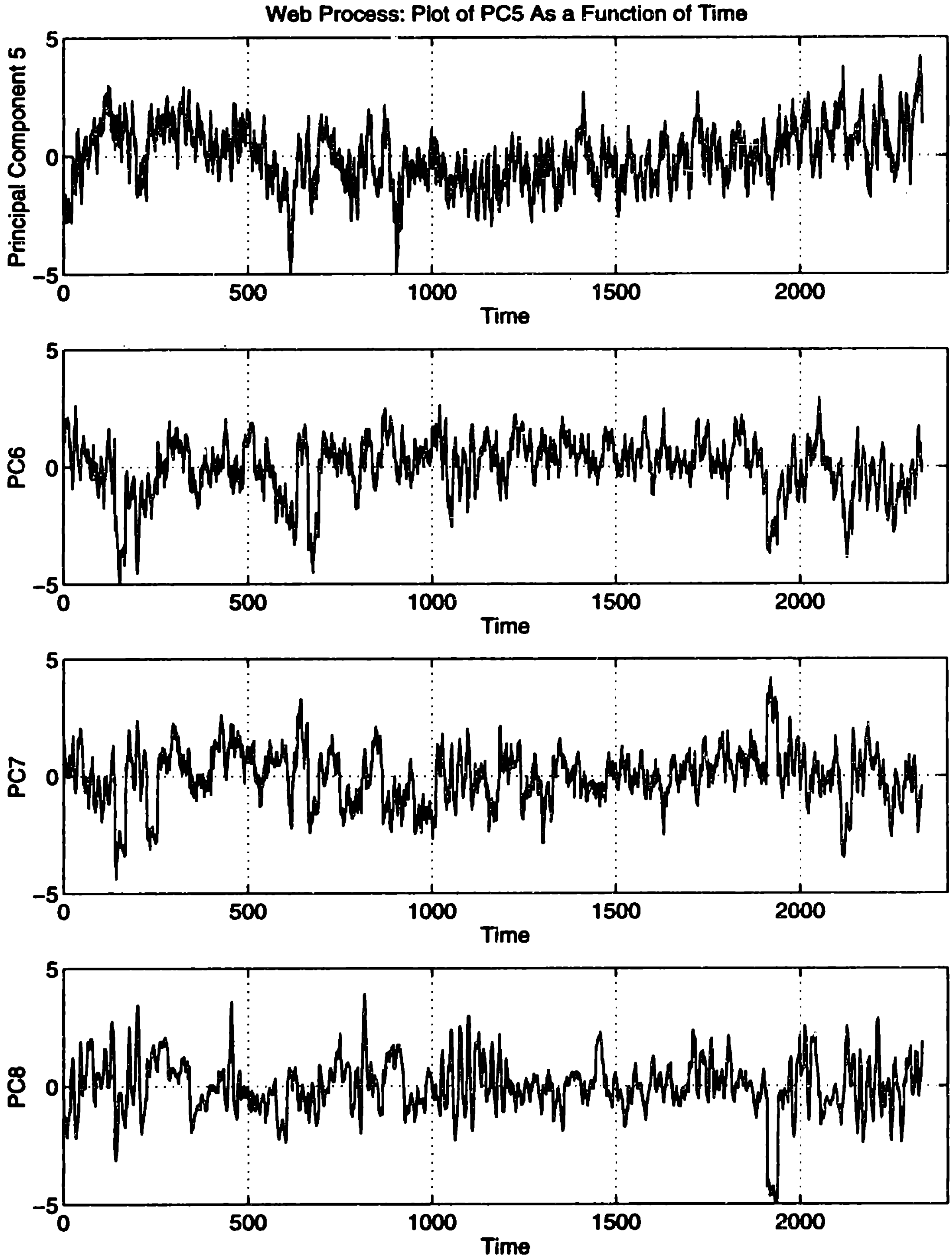


Figure 4.23: PC5, PC6, PC7 and PC8 of Web Process Variables

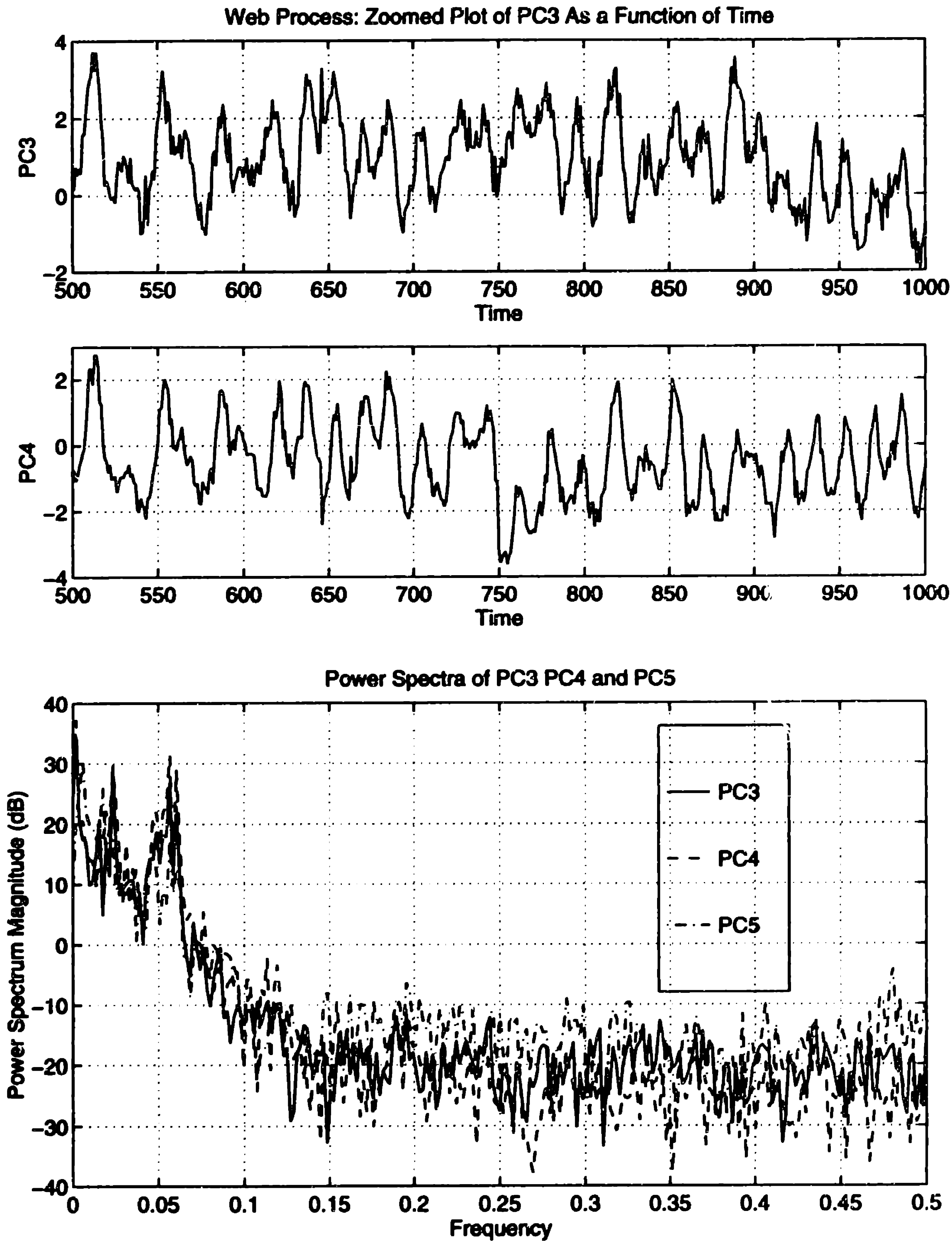


Figure 4.24: Zoomed PC3 and PC4 highlighting similar periodic nature. Bottom plot shows power spectrum of PC3-PC5

4.4.2 Cross Plots of Principal Components

Figures 4.25 and 4.26 shows the same data in principal component space. These are simply two dimensional views of the multidimensional data space that contain the most variation. Figure 4.26 shows the principal components plotted with different symbols representing 4 contiguous blocks of time. The process drifts gradually from region to region as time progresses.

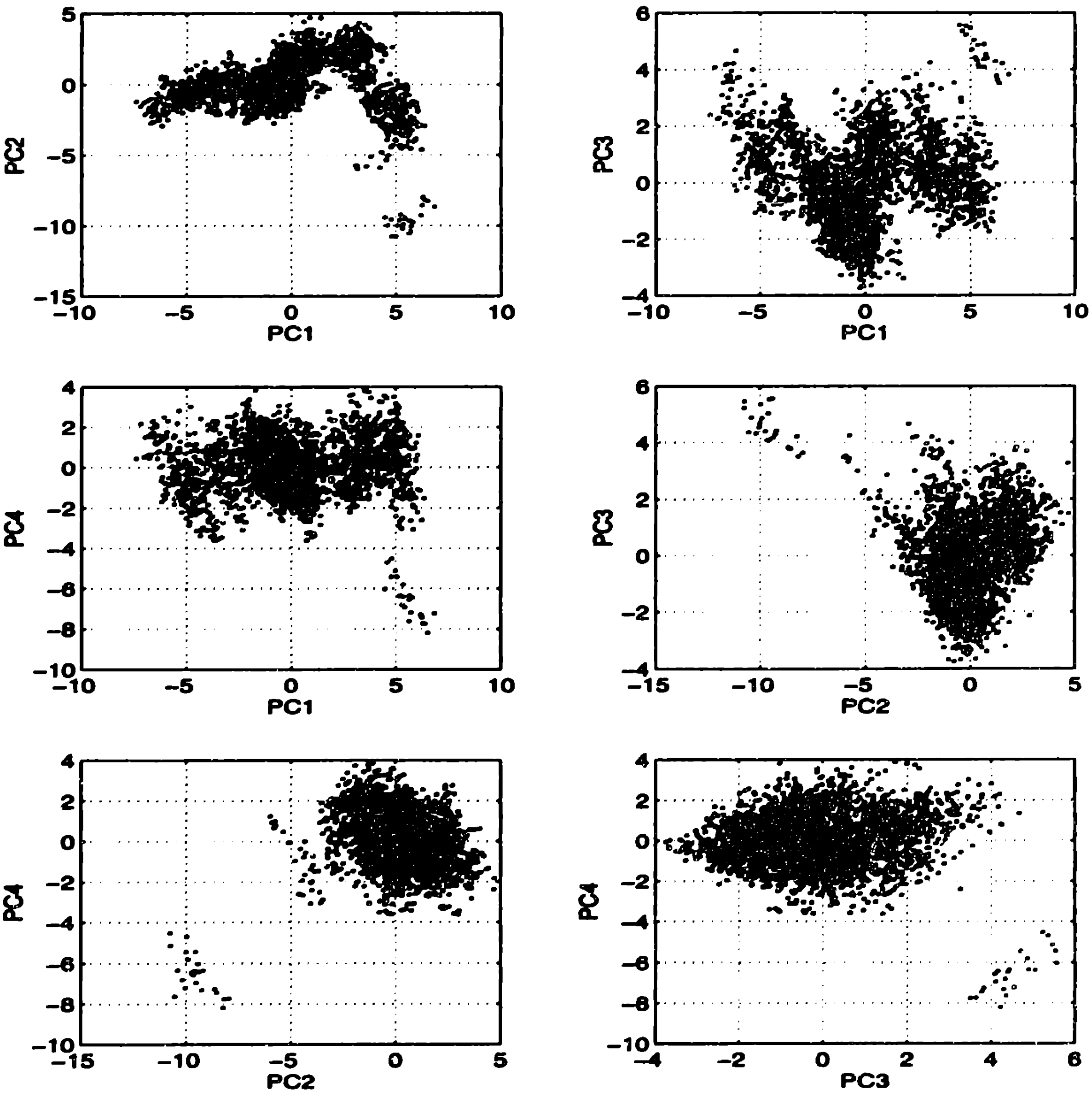


Figure 4.25: Cross Plots of the First 4 Principal Components.

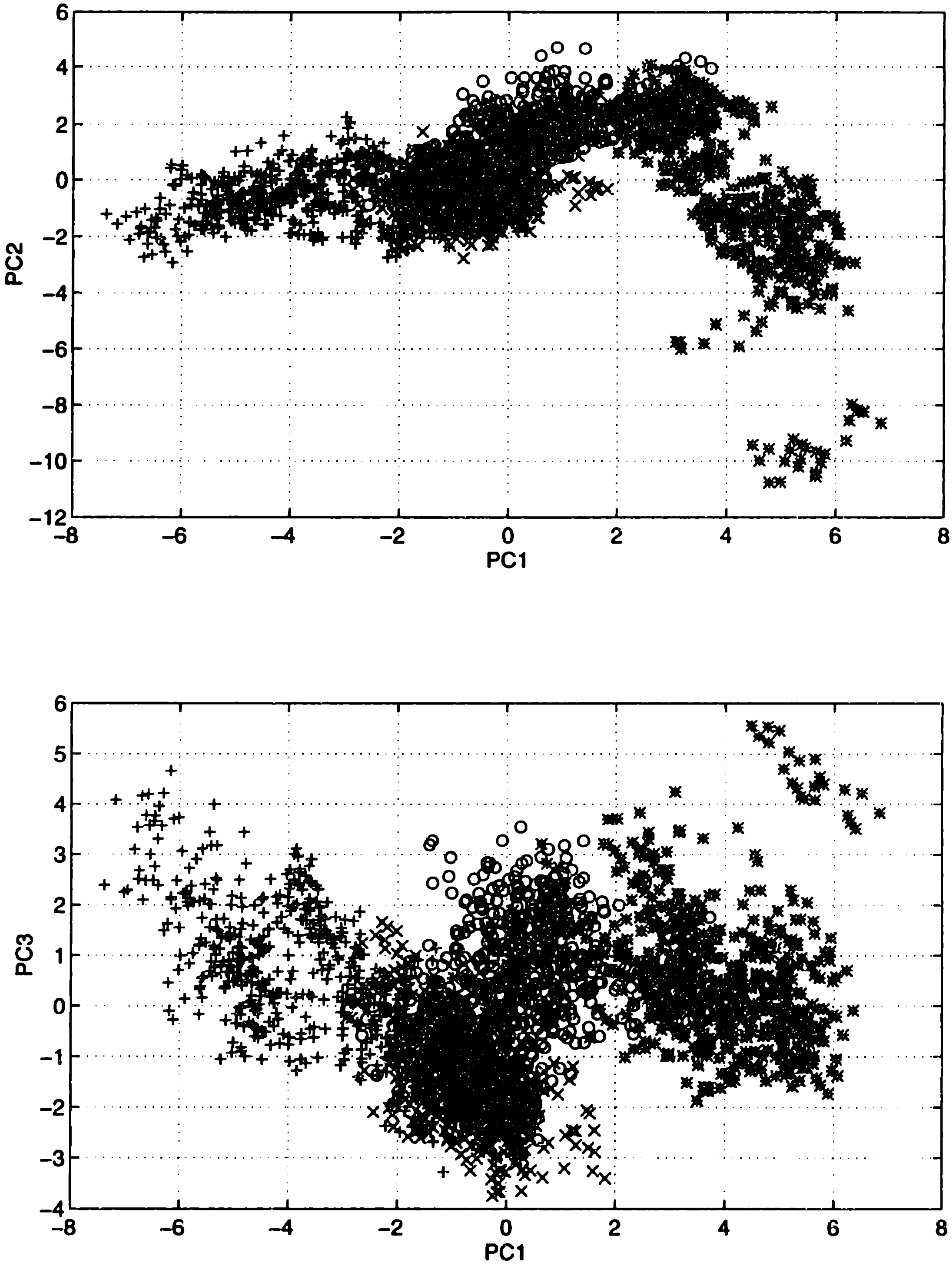


Figure 4.26: Cross Plots of Principal Components Showing Time Evolution of the Data

4.5 Discussion

This chapter started off by considering improved methods of prediction. The results suggest that a combination of linear ARMA modelling techniques hybridized with neural networks result in lower RMS error over just ARMA modelling. The second section on the Web Process consider an analysis of the defects. Frequency analysis indicates that there are periodically occurring patterns in the defects. This is helpful for locating the source of the defects in the process. In addition, removing these periodically occurring defects is a logical precursor to further correlation analysis between the process variables and the defects. Finally, the final section showed a useful way for viewing the time evolution of a process in multidimensional space. This space can be used to identify the best operating regions for a process.

Chapter 5

Medical Data

5.1 Introdution

The analysis done in this section is on a set of medical data. In all 8 variables are recorded on each of 11 patients simultaneously over a period of months. The methods of analysis focused on will be data reduction and dealing with unevenly sampled data. This medical data is a true test of the tools developed so far because it represents a worst case scenario:

- Most of the powerful tools for time-series analysis depend on evenly sampled data. However, there are widely varying sampling intervals in the data ranging from several seconds to days.

- The data sets are relatively small with typically only about 60 measurements.

- The data itself shows signs of being quite noisy.

By using Principal Components and by modifying the way autocorrelations are calculated the deleterious effects of the above limitations can be mitigated.

5.2 Dimensionality Reduction

One of the first things that should always be consideied about a multivariate data set is how many important degrees of freedom it contains. Principal Component Analysis shows that each of the patients data space can be reduced to 1 or 2 transformed variables that captures most of the variance. Principal Components take advantage of correlations among variables resulting in transformed components which are likely to contain significant information.

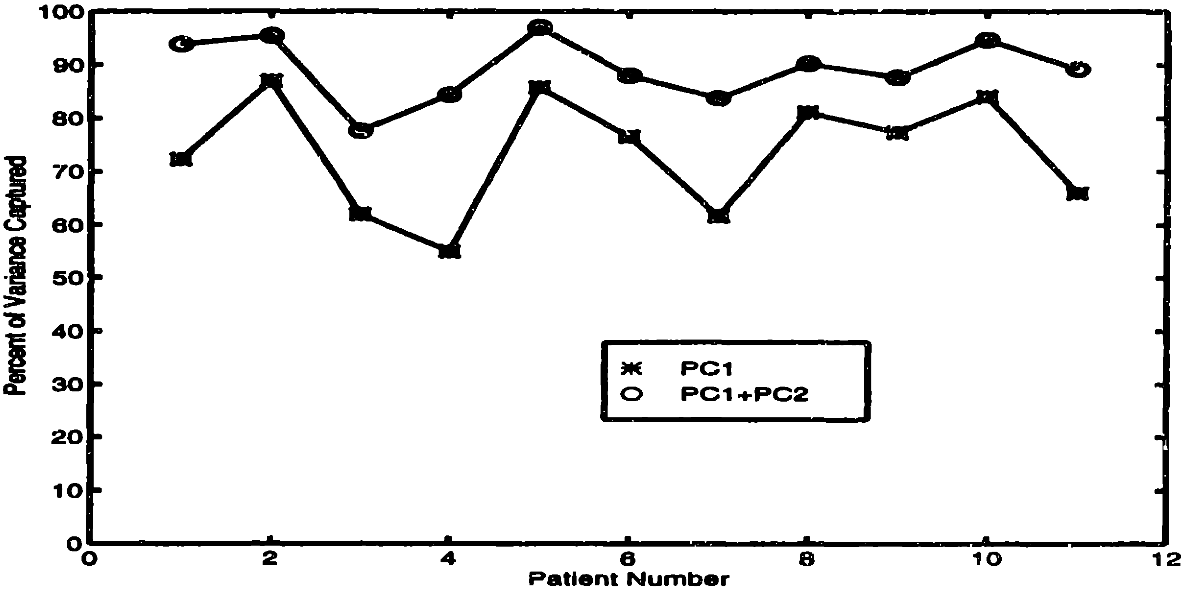
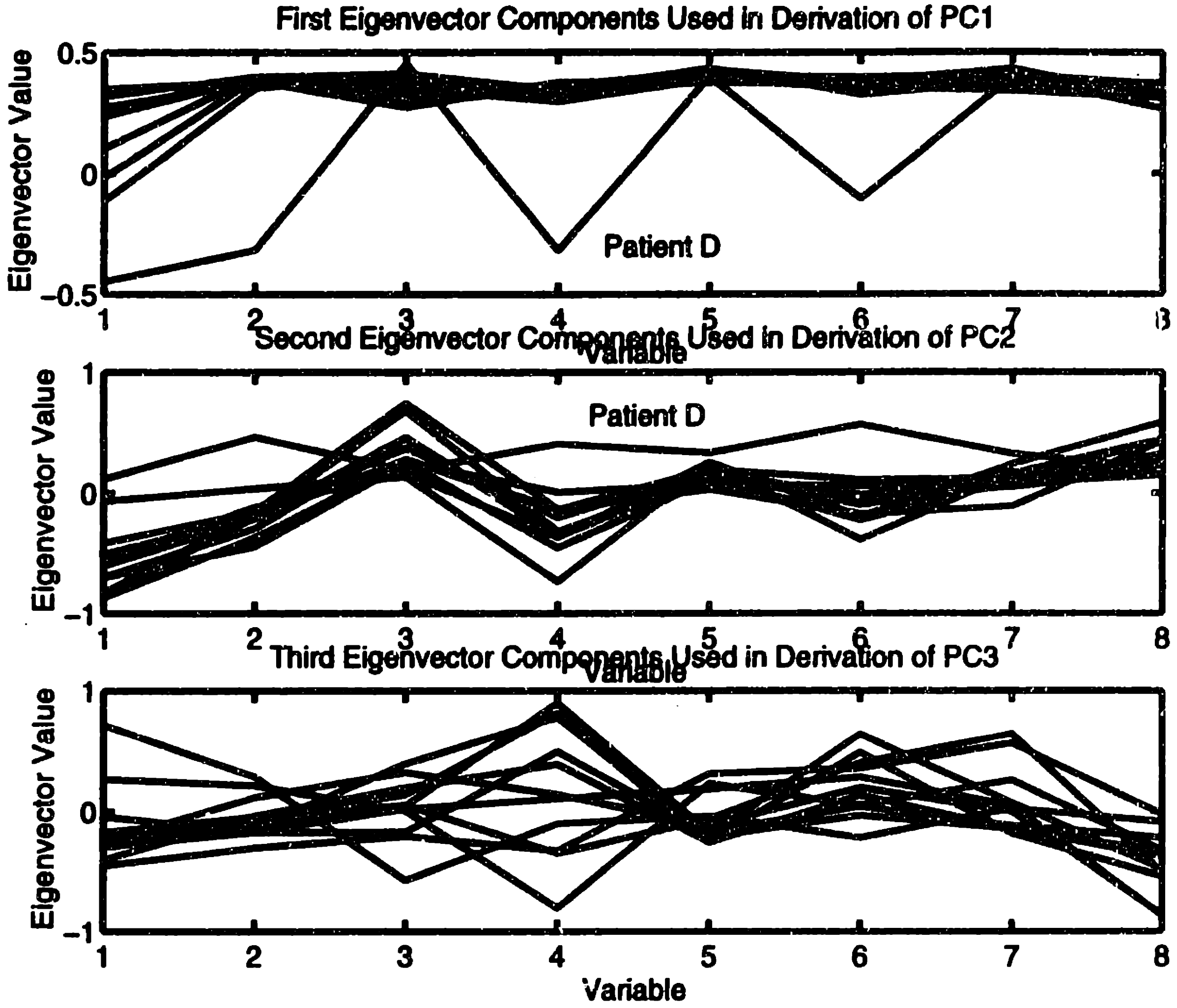


Figure 5.1: Variance Captured by Principal Components

Figure 5.1 shows the percentage of variance capture by the first and second principal components. Figure 5.2 shows that the two dominant eigenfunctions are essentially the same for all patients. In the top graph of Figure 5.2 the dominant eigenfunction for most of the patients is simply an average of all the zero-mean variance-normalized parameters which were measured. The middle figure shows the second eigenvector, for which most patients respond negatively to variable 1, positively to variable 3, and to a reduced degree for the other variables. Note that patient D appears to have the first and second eigenfunctions reversed – that is, the second principal component is roughly an average of all the measurements and the dominant one is roughly the second principal component for the other patients. Thus, there appears to be some medical significance to these two independent


First Eigenvector Components Used in Derivation of PC1

Second Eigenvector Components Used in Derivation of PC2

Third Eigenvector Components Used in Derivation of PC3

Figure 5.2: Plot of Dominant Eigenvector for Each Patient

aspects of the patient data.

The analysis shown here is on patient "C". Figure 5.4 for patient C shows several plots of the principal components. The first plot shows that most of the variance lies in the mean-value of all the measurements, and that there is no obvious correlation between any one principal component and any other; this is typical of all linear jointly Gaussian random processes.

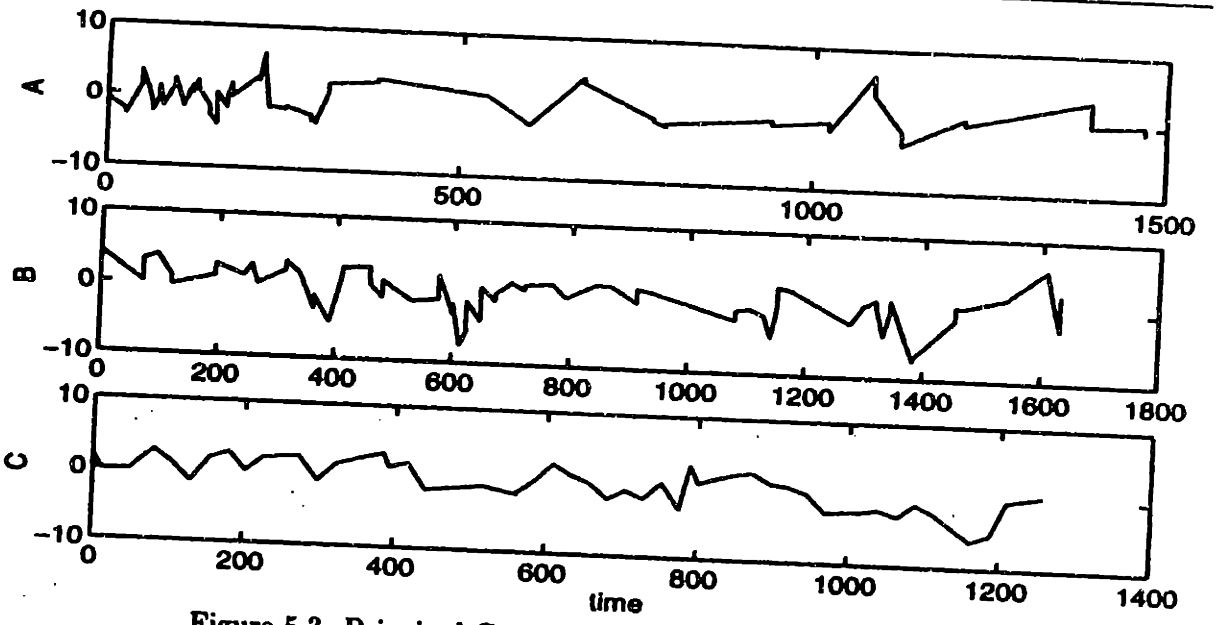


Figure 5.3: Principal Component 1 for Patients A, B, and C

5.3 Autocorrelation and Predictability

One of the most important questions that can be asked about this dataset is how predictable it is over time. Where in the dimensionality reduction analysis a critical factor was correlations *among variables* now correlations *over time* are considered. The function which considers correlations over time is the autocorrelation function. This function provides a measure of how correlated a signal is with itself as a function of time lag. So typically a signal will be correlated with itself over short time lags and less correlated with itself as the time lag increases. Autocorrelation analysis will answer some critical question: 1) Is there predictability in the data? and 2) If it is predictable what is the half life of predictability? (ie. How far into the future can predictions be reliably made?)

Before looking at the results it is important to emphasize the unique and challenging aspect of this set of data. Samples are taken at highly irregular time intervals ranging form 45 second intervals to times 2 or 3 days apart. Ideally the samples are evenly spaced and taken at times shorter than the phenomena of interest. Clearly if a patient's condition is known to be predictable 15 hours in advance but measurements are only taken every 3 days

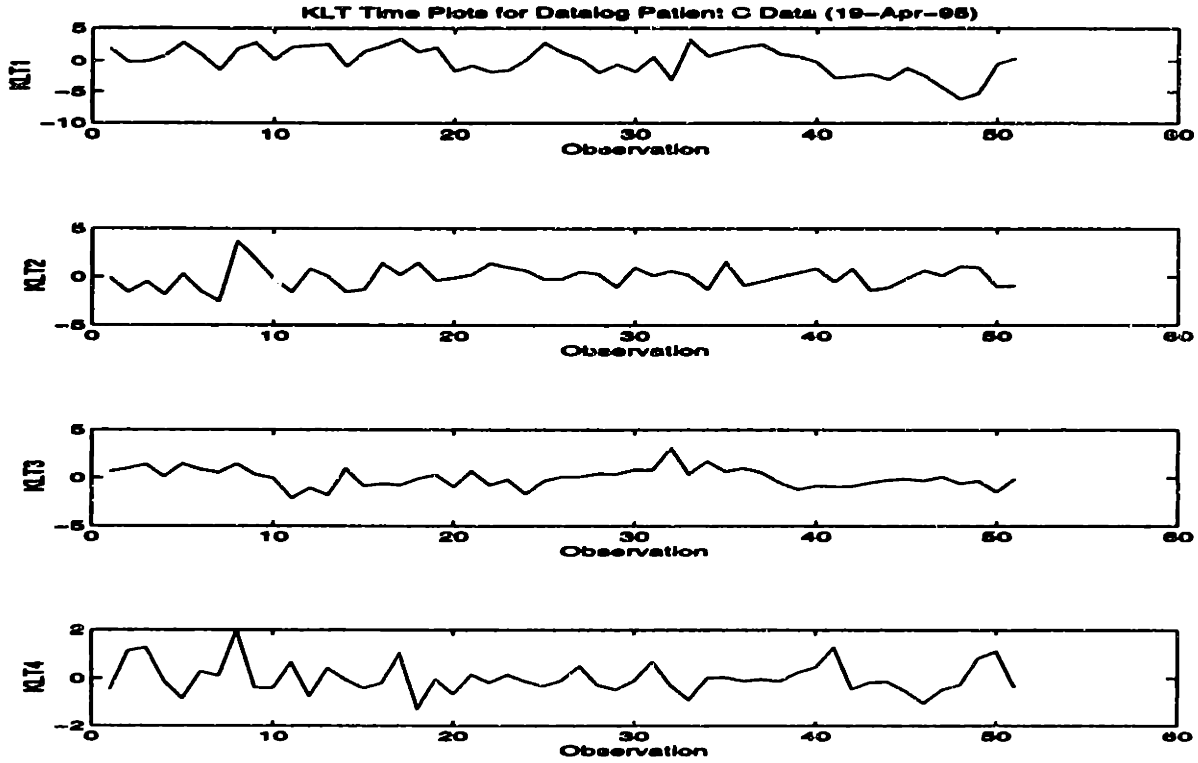


Figure 5.4: Principal Components for Patient C

then the ability to formulate a function which accurately predicts a patient's condition will be severely limited.

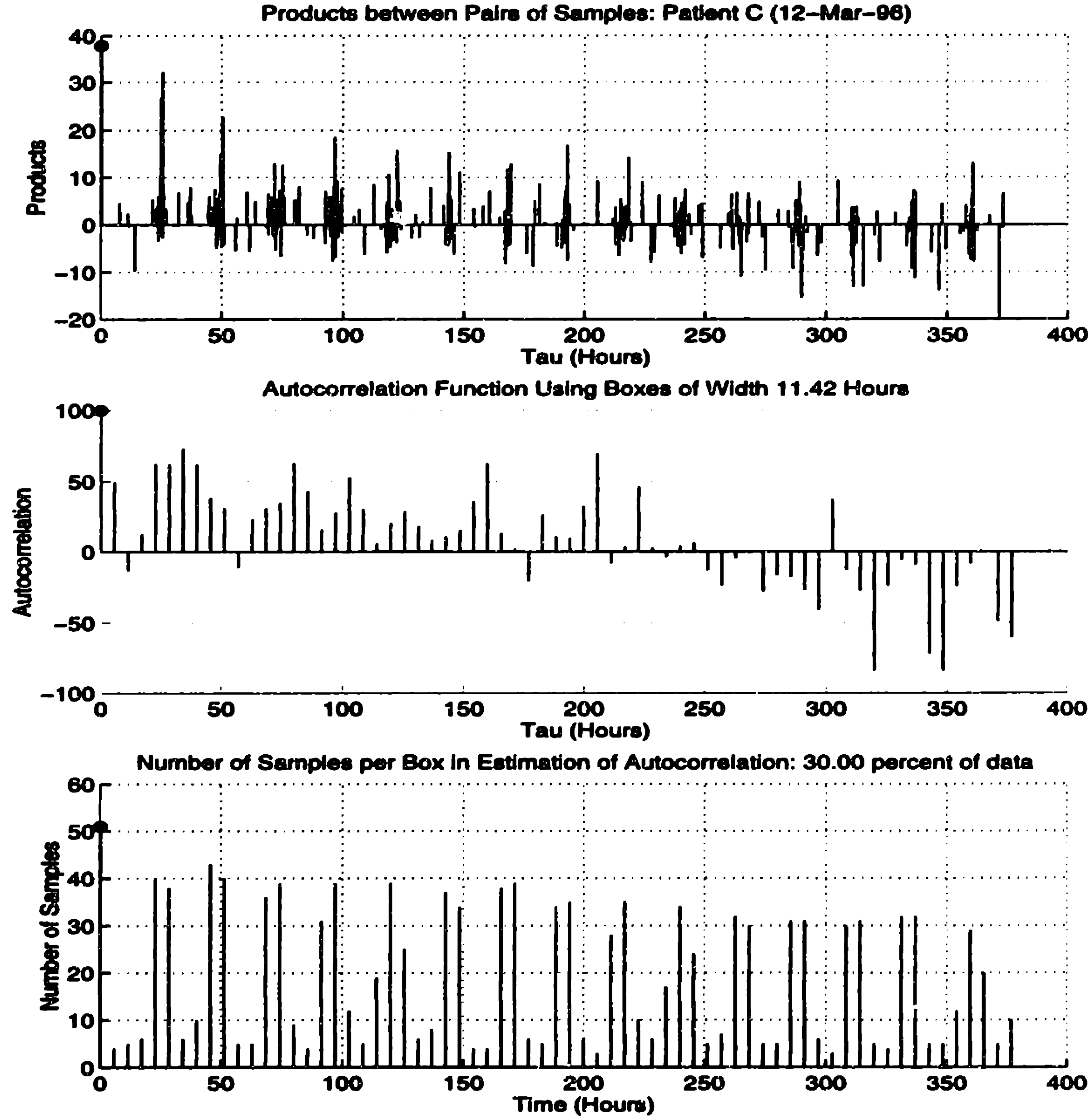
5.3.1 The Generalized Autocorrelator

This section will delve slightly into the technical aspects of the computation of the generalized autocorrelation function because it highlights the trade off that must be made between accuracy and resolution of results over time. The term *generalized autocorrelator* is used because it can be used to estimate autocorrlations for discrete data sampled at arbitrary times. The first step in the computation of the autocorrelation function is to find all pairs of products, $x(t)x(t-\tau)$. So for 50 observations there are 1275 unique pairs of points. Now to find the autocorrelation, $R_{xx}(\tau) = E[x(t)x(t-\tau)]$ these products are averaged over a box of a certain width in time. So to find $R_{xx}(\tau = 10)$ a decision must be made as to the width over which to average, T. The pairs of products which fall between $10 - T/2$ and $10 + T/2$ are determined and summed. This is then divided by the number of pairs of products, $x(t)x(t-\tau)$, in this time interval.

A larger time interval over which to estimate $R_{xx}(\tau)$ will result in a tighter confidence interval for the result, but it also results in less resolution of $R_{xx}(\tau)$ over time. So the time-interval should be big enough to produce small errors but small enough to resolve the autocorrelation estimate over time. Typically, there should be at least 10 samples per box. When looking at the autocorrelation plots at the end of this document it is important to look at the number of samples used in the calculation of $R_{xx}(\tau)$ for each τ. A small number of samples indicates a large possible error in the autocorrelation function. A good example of this is patient C. The "Number of Samples per Box" plot in Figure 5.5 (bottom plot) shows small values, followed periodically by relatively large values. At these very small values we want to ignore the autocorrelation function because the error is potentially large.

Figure 5.4 shows the time evolution of the top three principal components, and it suggests only that there is a drift towards negative values over the last five or ten observations; other than that, the data is difficult to distinguish from white noise. Figure 5.5 for patient C shows sample values of the autocorrelation function for various values of time offsets. The mean-square value for zero offset is indicated by the large black dot at the left side of the graph. If a boxcar filter with a width of 11.42 hours is used to filter the data in the top graph, the result shows that the correlation between measurements drops almost immediately to 50% and then decays slowly with a time constant of approximately 70 hours. Again, it is important to know that only certain boxes in the time delay domain have enough samples to be statistically significant; these are indicated by the bottom figure on the page (for example, the samples at around 24 hours). The implications here are that the health of patient C as indicated by the average measure of the eight normalized sense parameters is predictable only two or three days in advance. The power of predictability is poor because two or three days correspond to a reduction in correlation coefficient to values below .3.

Predictability for most of these patients is typically on the order of 12 hours. In some cases data is taken at short time intervals, so it is possible to be more accurate in terms of predictability at these short time intervals. In other cases the data is recorded only daily, and then it is much more difficult to deduce how rapidly predictability is falling.


Products between Pairs of Samples: Patient C (12-Mar-96)

Autocorrelation Function Using Boxes of Width 11.42 Hours

Number of Samples per Box in Estimation of Autocorrelation: 30.00 percent of data

Figure 5.5: The Generalized Autocorrelation Function for Patient C

5.4 Summary

Bullet Summary

- Individual patient can be reduced to usually 1 sometimes 2 significant variables which capture most of the variation. Each patient's data set originally contains 8 variables.
- Some of the patients indicate a potential predictability around 12 hours.
- The remaining patients are predictable to less than 12 hours.
- Many of the patients have autocorrelation values that drop almost immediately to 50% implying high patient variability or noise.

Discussion

All of the patient's data sets could be reduced to at most 2 significant components in the KLT domain. However individual patients demonstrated significant differences in predictability. Some of the patients mentioned above have some degree of predictability while others appear to have very little predictability. It should be noted that for most of the patients we cannot accurately access the autocorrelation function much below 10 hours. The only way to increase the resolution here is to get samples at smaller time intervals then in the present data set.

Various scenarios can be imagined which would adversely affect analysis. If samples are taken only when the phenemona of interest is present then the samples will not represent a complete picture of a patient's condition. Or if a measurement is always taken shortly after treatment then his condition is likely to be dependent on when he took the treatment and not on natural fluctuations in the patient'c condition.

The results suggest that a slightly more disciplined measurement taking procedure could yield greatly improved results. Clearly the more often measurements are taken the better. But for practical purposes, if measurements were taken 2 or 3 times day for a period of 2-3 weeks a solid set of data could be obtained yielding medically valuable information.

Chapter 6

Wafer Data

6.1 Introdution

Monitoring changes from normality is critical in a manufacturing environment. Keeping a process within a specified set of parameters is essential to reduce variation. Consider plotting all the parameters of a process in variable space. Under normal operating conditions these parameters will lie within some multivariate operating region. Often this will show up in Principal Component space as an elliptical scatter of data. In some cases a plot of one principal component against another will reveal a distinct clustering characteristic. Data collected from an assembly line process, for instance, revealed many distinct clusters each corresponding to distinct positions in the assembly line. In a different data set the reduced variable space revealed 4 distinct operating regions each corresponding to an assembly route.

So a plot of process parameters in variable space provides a static view of important operating regions. However, operating characteristics are also strongly dependent on time. Suppose a process separates into several clusters in variable space. In this form there is no way of understanding the dynamic evolution of these variables. For one process a point might jump randomly from cluster to cluster sequentially in time. Another process may jump to a cluster and then stay there for a while before moving on to another cluster. Time-series analysis is needed to unravel the dynamics of a process. In particular, this section addresses methods for both *detecting* important changes in a process and for finding time patterns which are prevalent in physical phenenoma.

The data set used here was purposely chosen because there is no obvious clustering in variable space. However, a dynamic analysis of the elliptical fuzzball in the plot of PC1 against PC2 reveals a nonstationary process that does change over time.

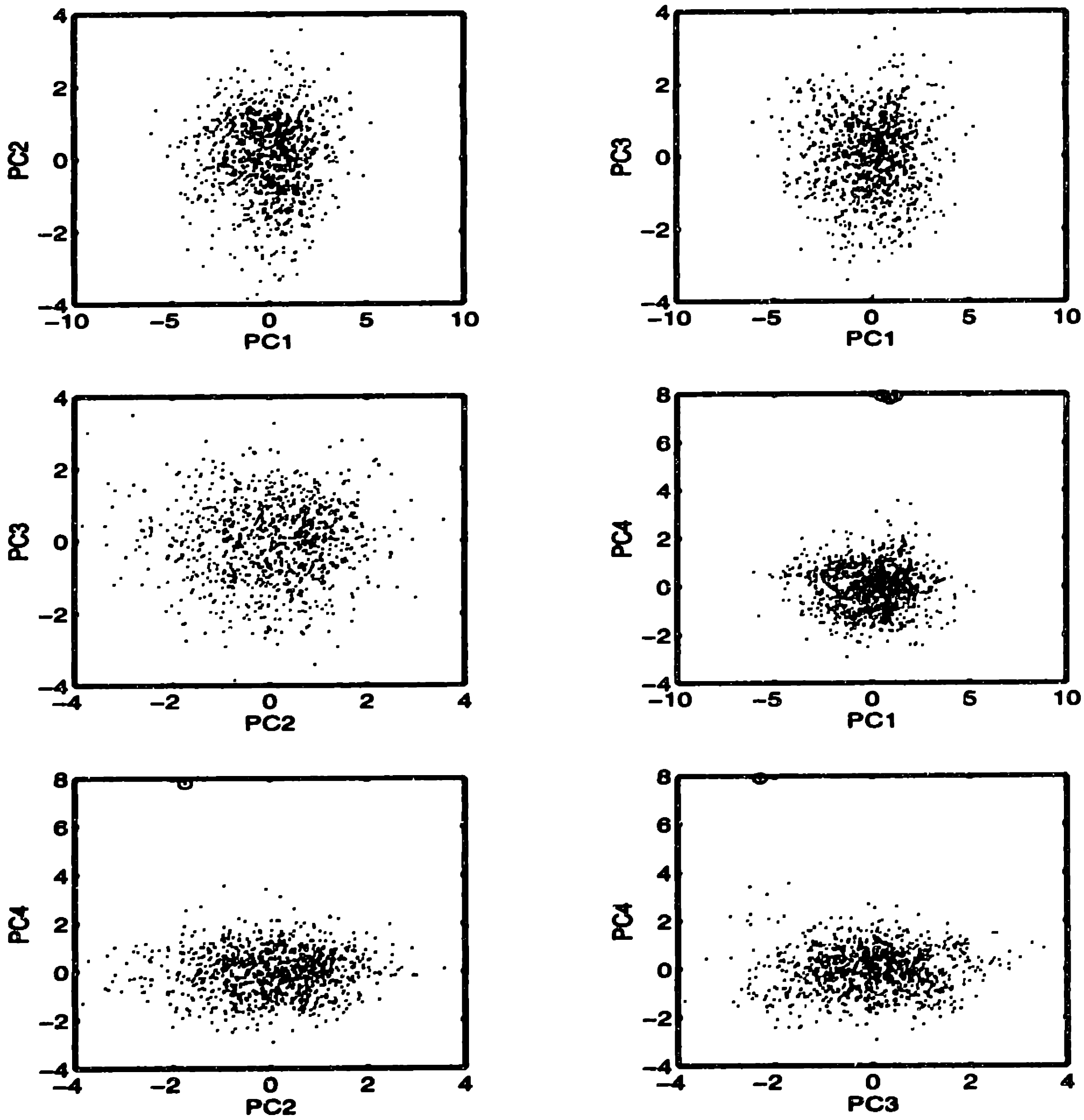


Figure 6.1: Plot of the Principal Components

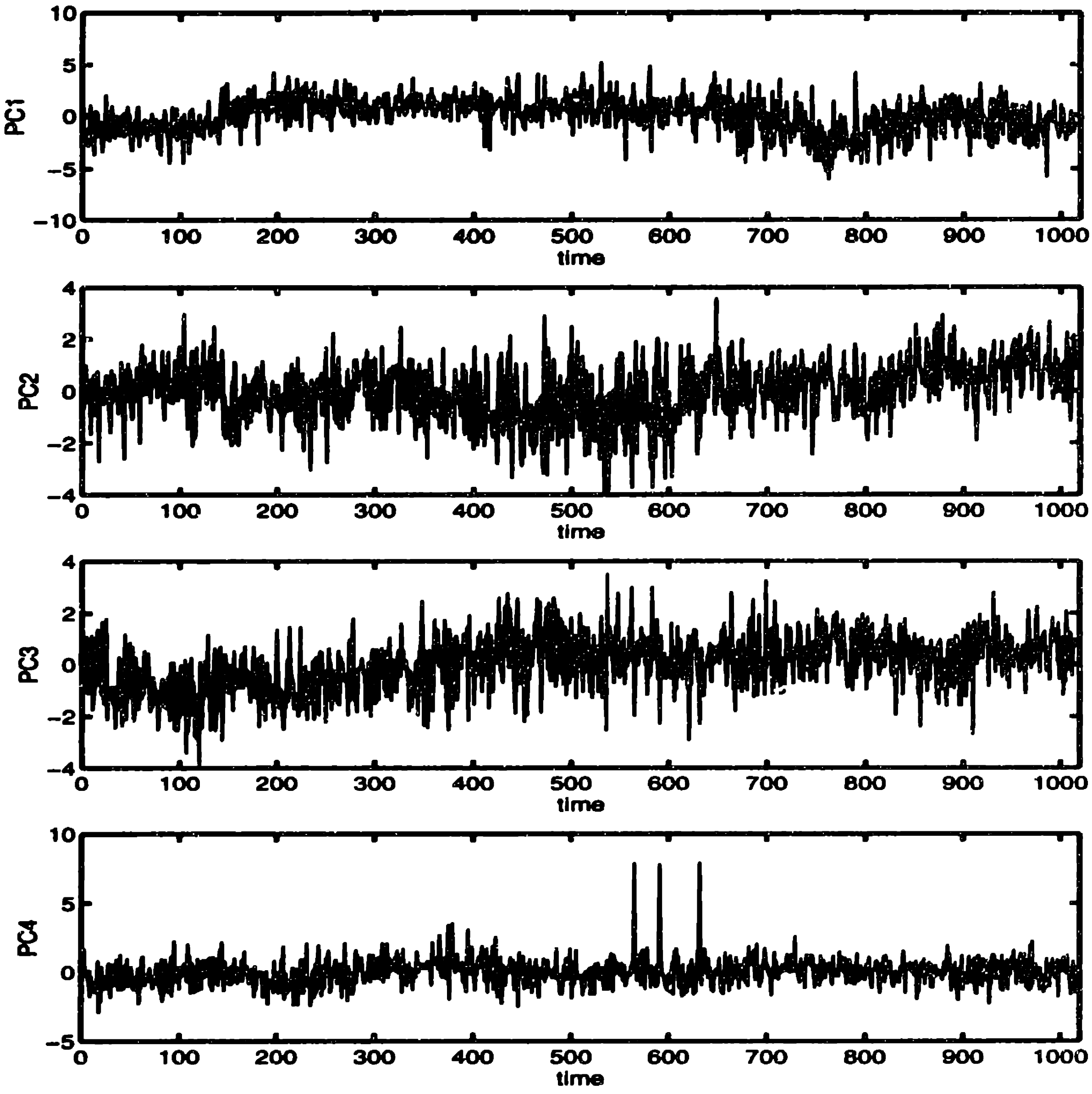


Figure 6.2: The Principal Components as a Function of Time

6.2 Surprise Detection

Figure 6.2 shows plots of the first 4 principal components as a function of time. A quick glance suggests that the process does indeed change over time. PC1 has a fairly obvious changing mean. A little less obvious, but also relevant, is a change in the variance of the process over time. The last 200 points vary over a wider range than over the range from about 200 to 400.

There are many different ways to try and describe changes in a set of data. A special type of change, called a *surprise*, will be defined in such a way so as to have a foundation for analysis. A *surprise* is behavior that is not predictable from prior data in a multidimensional ordered set. This means that characterization as a surprise is very dependent on the information before it. Behavior found to be unpredictable with a set of data may possess behavior that is in fact predicatable given a larger set of data. A simple example is a periodic square wave; the first jump isn't predictable but after a few periods have been seen the jumps are predictable.

6.2.1 In-Quadrature Filtering

In this the in-quadrature filter is used to locate points in a time series where changes are taking place. The output of the in-quadrature filter has the interpretation of showing the rate of change of the input to the filter. So filtered signal peaks are places where a signal is changing most rapidly over a particular bandwidth. Points of maximal change are often good indicators of a surprise.

Figure 6.3 shows the first Principal Component passed through a lowpass filter. This process alone makes the shifts in the signal much more obvious. Compare this filtered signal with the orignal signal in Figure 6.2. Shown also is the magnitude of the in-quadrature filtered output. The in-quadrature filter center frequency is .03 with a filter width of .03. Not only does the in-quadrature output highlight the changes which are visually obvious, it also accurately locates in time the points of maximal rate of change. So the complex exponential terms which comprise the frequency domain representation of the series at the given center frequency and bandwidth are changing most rapidly at the peaks in the in-quadrature output.

In-quadrature outputs could be checked at higher frequencies as well for surprises. An output with large magnitude followed by several outputs of low magnitude could indicate a

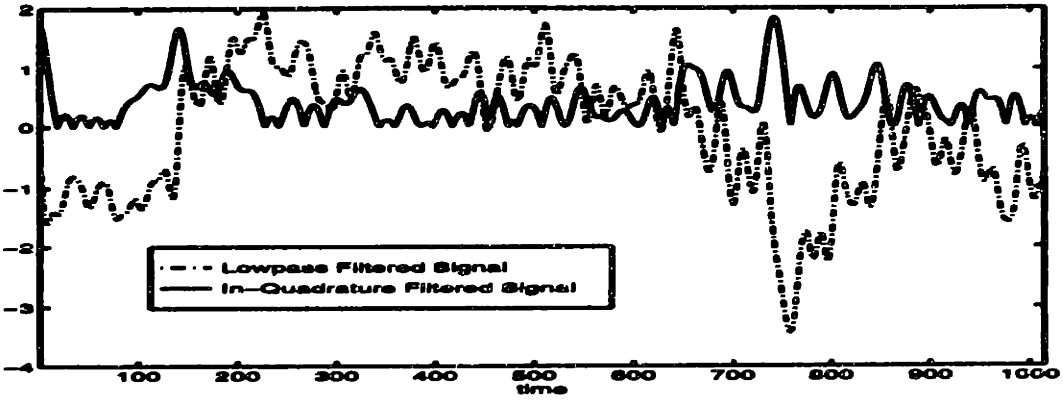


Figure 6.3: In-Quadrature Filtered Output

surprise. However, in general, the performance will gradually deteoriate because as higher frequencies are considered the output is basically just measuring the difference between adjacent points which are subject to random variation.

6.2.2 Detection Using Prediction

Figure 6.3 gives an indication of the degree to which each point in a series is predictable. For reference and clarity a lowpass filtered version of the original signal is included. Superposed on this is the magnitude of a residual series. An ARMA model was trained on the data and a predicted series was generated by going forward in time. Then the same thing was done on a flipped version of the original series. These two predicted series, correctly reordered in time, were then subtracted from one another to produce the residual series plotted. The ARMA model was trained to predict three points into the future. Points in the residual series, or shocks, that are relatively large in magnitude represent place where the ARMA model had difficultly predicting. The plot indicates that there are surprises at around $t = 140$ and $t = 750$. This is consistent with the in-quadrature analysis of Section 6.2.1.

To support these results the exact same analysis as described above was performed on the series with the first 200 points removed. The purpose is to determine if the first surprise around $t = 140$ helps help in the prediction of the surprise around $t = 750$. The results produce very similar residual plots. Therefore it can be concluded that the two prominent

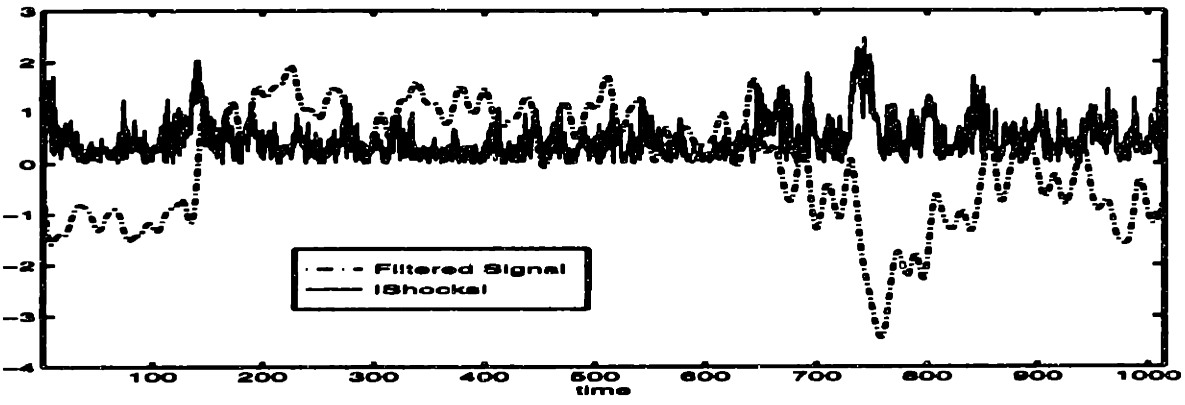


Figure 6.4: Residuals After Prediction

residuals are indeed surprises. The additional information of having the first 200 data points doesn't help the prediction model at around $t = 750$.

6.3 Discussion

This chapter presented two methods for detecting surprises in a process. Flagging points where the process is rapidly changing by using the in-quadrature filter or flagging points which are relatively unpredictable by modelling the data using ARMA model are effective ways of locating a shift in operating characteristics. For detecting surprises which take place over a long time interval the in-quadrature filter is better to use while surprises that take place quickly in time will show up very well by using ARMA prediction techniques.

Chapter 7

Conclusions and Suggestions for Future Work

This thesis presented methods for analyzing time-ordered data sets generated by natural physical process, especially manufacturing. Principal components analysis is very effective in reducing large data sets containing many variables and is almost always used as a first step of analysis. ARMA models combined with neural networks were shown to be very effective in modelling data even when the data was nonstationary. Accruate predictions into the future could be made using these models. Residuals generated from ARMA models were shown to be an effective way to detect surprises in a process. Alternatively, in-quadrature filters highlight where rapid changes in a process are taking place over certain time scales. Fourier techniques proved to be useful in summarizing the important characteristics of a time-series. Even when working with very non-gaussian data, important insights into the data was gained using frequency domain analysis.

Work could be done in the areas of time-series prediction, surprise detection and effective methods for summarizing important and relevant characteristics of a time-series. Further work should be done looking at ways to automate the hybridization of ARMA models with neural networks. How to select the appropriate degrees of freedom to be used in both the ARMA model and the neural network are important. A reasonable criterion is to compare the distribution of the residuals in the training set and the test set; they should be similar. In the area of surprise detection, methods should be developed for dealing with multivariate surprises. This is related to looking for multivariate control limits. Often in a manufacturing process, single univariate parameters are monitored to remain with certain limits. However, multivariable control limits may be the critical factors for a given process. Finally, data summary is essential for communication of critical aspects of time-series data. Manufacturing data sets are usually loaded with periodicities, trends, drifts and patterns. Lucid time domain and frequency domain methods should be developed to highlight these attributes. The fast-foldover fourier transform or the walsh transform, for instance, could be explored to deal with the binary data encountered in Chapter 4.

It has been the opinion of many people working with these data sets that robust linear or quasi-linear methods are the way to go in terms of analysis techniques. Most manufacturing processes are specifically designed to operate in linear operating regions. The relative simplicity of linear methods and the potential for meaningful interpretation of results make robust linear methods most attractive for real data analysis.

Bibliography

[1] Beauchamp, K. G., *Walsh Functions and Their Appliations*, Academic Press, London, ©1975.

[2] Bendat, Julius S. and Piersol, Allan G., *Random Data: Analysis and Measurement Procedures*, Second Ed., John Wiley & Sons, Inc., New York, ©1986.

[3] Brockwell, Peter J. and Davis, Richard A., *Time Series: Theory and Methods*, Second Ed., Springer-Verlag, New York, ©1991.

[4] Haykin, Simon, *Neural Networks: A Comprehensive Foundation*, Macmillian College Publishing Company, Englewood Cliffs, New Jersey, ©1994.

[5] Johnson, Richard A. and Wichern, Dean W., *Applied Multivariate Statistical Analysis*, Third Ed., Prentice Hall, Englewood Cliffs, New Jersey, ©1992.

[6] Krzanowski, W. J., and Marriott, F.H.C. *Multivariate Analysis*, Part I, Edward Arnold, London, ©1994.

[7] Oppenheim, Alan V. and Schafer, Ronald W., *Discrete-Time Signal Processing*, Prentice Hall, Englewood Cliffs, New Jersey, ©1989.

[8] Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T., *Numerical Recipes*, Second Ed., Cambridge University Press, Cambridge, ©1992.

[9] Weigend, Andreas S. and Gershenfeld, Neil A., Editors, *Time Series Prediction: Forecasting the Future and Understanding the Past*, Addison-Wesley Publishing Company, Reading, Massachusetts, ©1994.

[10] Yamashita, Yukihiko and Ogawa, Hidemitsu, Relative Karhunen-Loève Transform, *IEEE Transactions on Signal Processing*, Vol. 44 No. 2, February 1996.