NAVAL
POSTGRADUATE
SCHOOL

MONTEREY, CALIFORNIA

THESIS

ESTABLISHING A VIBRATION THRESHOLD VALUE, WHICH ENSURES A NEGLIGIBLE FALSE ALARM RATE FOR EACH GEAR IN CH-53 AIRCRAFT USING THE OPERATIONAL DATA

by

Mehmet Elyurek

December 2003

Thesis Advisor: Samuel E.Buttrey
Second Reader: Robert Koyak

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY	**2. REPORT DATE** December 2003	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE: Establishing a vibration threshold level, which ensures a negligible false alarm rate for each gear in CH-53 aircraft using the operational data	**5. FUNDING NUMBERS**
6. AUTHOR(S) Elyurek, Mehmet	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) Goodrich Corporation Fuel & Utility Systems, Vergennes, VT 05491	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.	**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

Rotating machinery such as gears plays an important role in control of an aircraft. The health of this machinery is a key ingredient to both platform safety and mission success, especially in military operations. The purpose of the thesis research is to establish a vibration threshold level for each particular gear in CH-53 aircraft such that, while minimizing in-flight risk, a negligible false alarm rate obtained.

This study uses Box-Jenkins time series modeling (ARMA) with regression; Mahalanobis distance metrics, goodness-of-fit tests and the Bonferroni correction to explore the structure of the historical acquisition datasets for particular gear type and aircraft, to set vibration threshold values for "Warning" and "Alarm" situations. Although 28 datasets could not be modeled because of small sample sizes, the other 224 data sets were successfully modeled using ARMA with regression modeling technique. The Mahalanobis distance metric was then used to set a threshold value of "Warning" and "Alarm" for each gear type. These threshold values were then checked with new data; 200 outliers for "Warning" and 69 outliers for "Alarm" were detected. These outliers might be evaluated as false alarms.

14. SUBJECT TERMS Box-Jenkins Time Series Modeling (ARMA), Mahalanobis Metric, Goodness of Fit Tests, Bonferroni Correction	**15. NUMBER OF PAGES** 97
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

ESTABLISHING A VIBRATION THRESHOLD LEVEL, WHICH ENSURES A NEGLIGIBLE FALSE ALARM RATE FOR EACH GEAR IN CH-53 AIRCRAFT USING THE OPERATIONAL DATA

Mehmet Elyurek
Lieutenant Junior Grade, Turkish Navy
B.S., Turkish Naval Academy, 1998

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 2003

Author: Mehmet Elyurek

Approved by: Samuel E. Buttrey
Thesis Advisor

Robert Koyak
Second Reader

James N. Eagle
Chairman
Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Rotating machinery such as gears plays an important role in control of an aircraft. The condition of this machinery is a key ingredient to both platform safety and mission success, especially in military operations. The purpose of the thesis research is to establish a vibration threshold level for each particular gear in CH-53 aircraft such that, while minimizing in-flight risk, a negligible false alarm rate is obtained.

This study uses Box-Jenkins time series modeling (ARMA) with regression, Mahalanobis distance metrics, goodness-of-fit tests and the Bonferroni correction to explore the structure of the historical acquisition datasets for particular gear type and aircraft, to set vibration threshold values for “Warning” and “Alarm” situations. Although 28 datasets could not be modeled because of small sample sizes, the other 224 data sets were successfully modeled using ARMA with regression modeling technique. The Mahalanobis distance metric was then used to set a threshold value of “Warning” and “Alarm” for each gear type. These threshold values were then checked with new data and 200 outliers for “Warning” and 69 outliers for “Alarm” were detected. These outliers might be evaluated as false alarms.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

The purpose of this study was to establish a vibration threshold level for each particular gear in CH-53 aircraft such that, while minimizing in-flight risk, a negligible false alarm rate is obtained. Aircraft safety is a very important issue to the military. Every precaution should be taken to minimize risk to the aircraft crew. The basic concept for threshold setting is to pick a threshold value high enough such that the worst aircraft, while still healthy, would not give a false alarm.

The data used in this study was supplied by Goodrich Corporation Fuel & Utility Systems. The data consist of 23,187 acquisitions and 20 attributes for 63 gear types and four different tail-numbered CH-53E aircraft. The data includes seven condition indicators[1] (CI) (See Table 1 in Chapter II) for each gear type. To calculate a threshold value, first, 252 individual data sets were created from the entire data for each particular gear and tail number. Each of the seven CIs were considered as a univariate time series.

Box-Jenkins Autoregressive Moving Average Models (ARMA) were used to model each of these univariate time series. Examining the time plots for each CI, it was observed that almost all of them were plausibly stationary. The autocorrelation and partial autocorrelation plots were then examined to determine the order of Autoregressive (AR)

[1] "Condition indicator (CI) is nothing more than an algorithm. For example, residual kurtosis measured the kurtosis of the time domain signal after the major gear and shaft rates have been removed" (Goodrich Corporation Fuel & Utility Systems, 2003).

and Moving Average (MA) components. Based on these plots, ARMA(1,1) models were suggested. Then we added the torque effect as a regression variable to our models because it was believed that torque affected the CIs. The standardized residuals of each CI model were used to set threshold values of “Warning” and “Alarm”.

Our analysis was based on detecting any unusual level in CI values. For this purpose, we used the Mahalanobis distance, which is a multivariate distance metric. This analysis provided insight about the expected range of the distance metric for a specific healthy gear type.

Next, we needed to find the distribution, which would fit to each Mahalanobis distance data set. Most of the histogram plots for the Mahalanobis distance data sets for a particular gear type and tail number looked as if they came from exponential distributions.

However, we applied Chi-Square and Kolmogorov-Smirnov goodness-of-fit to verify if the Mahalanobis distance data sets came from exponential distributions. Since more than one goodness-of-fit test was performed, in order to control Type I error, we applied the Bonferroni multiple comparison correction which assured an overall Type I error no greater than 0.05. Using the Bonferroni adjusted goodness-of-fit tests, 84% of the data sets using Chi-square and 87.5% of the data sets using Kolmogorov-Smirnov produced non-significant results with respect to the null hypothesis specifying the exponential distribution. Therefore, we set threshold values for “Warning” and “Alarm” using the critical values of the exponential distributions of those data sets. The basic concept for threshold setting was to pick a threshold high enough that the worst aircraft, while still healthy, would not give a false alarm. For this reason, as a rule of thumb, we used a 0.999 quantile level for “Warning”, and a 0.999999 quantile level for “Alarm” threshold levels.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

The United States Navy, in association with Goodrich Corporation Fuel & Utility Systems, is continuously seeking ways to decrease the false alarm rates for “Warning” and “Alarm” in different types of aircraft using the vibration data collected during the operational flights.

Rotating machinery such as pumps, gears and transmissions are used in vehicles, ships and aircraft. These components support critical functions that aid in power, stability, propulsion and control of these platforms. The health of this machinery is a key ingredient to both platform safety and mission success, especially in military operations.

> Components subject to cyclic fatigue conditions develop cracks in critical high-stress locations as a result of pre-existing machining or manufacturing-induced defects, poor operating conditions (loss of lubrication, etc. leading to fretting damage), foreign object damage, environmental factors (corrosive environments and resulting pitting damage) or excessive loading. Such interactions, either between new components, new and worn components, and healthy or fatigued/damaged components, coupled with the difficulty in determining exact crack initiation sites makes it difficult to predict remaining component life. Practical real-time optical or strain measurement using conventional sensor technologies has not proven reliable for production purposes (Goodrich Corporation Fuel & Utility Systems, 2003).

Techniques designed to assess the health of this machinery use component-level state-awareness indicators obtained from analyzing the vibration signal. These indicators are categorized as either normal, warning or alarm. There are some reliable indicators that already are used to ascertain the health of each component and the corresponding assembly (group of components) at a specific instance in time. Despite the improvement in probability of detection and false alarm rate, current health assessments do not relate previous and remaining component life.

There are two mechanical diagnostic tests that can be performed. The first is a usage-based test. The second is measurement-based. The usage-based test calculates the worst-case damage that a new part could accumulate before failure. The real-time damage is recorded and reflects actual flight conditions such as airspeed and maneuvers. The proportion of real-time damage to worst-case damage is considered the usage of the aircraft component. This method does not account for manufacturing defects, corrosion or faulty maintenance. On the other hand, the measurement-based test uses an accelerometer close to the component that measures the vibration felt by that component. This test is used to infer the current health of the component. Our analysis relies on measurement-based data. The following describes the process by which component health is measured.

> An acquisition takes configuration data which consists of gear, bearing and shaft information, and calculates a health index (HI) based on a number of CIs. The gear information consists of the number of teeth, the RPM, the shaft on which it is mounted and sensor. The health of the component is calculated (currently) by taking consensuses of CIs used for that part. For example, in the case of gears, 7 CI's (See Table 1 in Chapter II) are used. If three CIs are

> greater than three standard deviations above the normal mean level, the component is considered in “Warning” and if there are 3 CIs greater than 6 standard deviations, the component is in alarm (Goodrich Corporation Fuel & Utility Systems, 2003).

A false alarm occurs when the health index (HI) is in warning or alarm when it should be in normal. One of the most important issues is to minimize the number of false alarms during operations. But on the other hand, undetected faults can result in catastrophic failures. There must be a balance between these objectives.

In this thesis, we will deal with the gear data and our goal is to determine a threshold value of “Warning” and “Alarm” for each particular gear of CH-53E type helicopters, and to obtain reasonable false alarm rates.

B. OBJECTIVE

The purpose of the thesis research is to establish a vibration threshold level for each particular gear in the CH-53 aircraft such that, while minimizing in-flight risk, a negligible false alarm rate is obtained.

This thesis will benefit the military by ensuring a lower false alarm rate on its helicopters. This will help to decrease ownership costs, which include the replacement and/or maintenance of any helicopter component as a result of a false alarm.

The vibration data for CH-53 helicopters was provided by Goodrich Corporation Fuel & Utility Systems. The data was collected collected between July 1, 2001 and September 1, 2003 during operational tests and it includes different CIs related to accelerometers and gears for each specific tail number. The entire data consists of 23,187 observations on 20 attributes.

C. SCOPE

Helicopter safety is a very important issue to the military. The lives of the crew on the helicopter are precious and every precaution to minimize risk while flying should be taken. A naïve model would set a very low threshold level to ensure that no failure occurs in flight. This model is impractical due to cost constraints. A low threshold would require frequent replacement of the components of the helicopter, at a high cost. The thresholds must be set high enough such that a false alarm is a rare event. Therefore, the goal of this thesis is to determine if a threshold level exists for each particular gear in an aircraft such that, while minimizing in-flight risk, a negligible false alarm rate is obtained.

D. COURSE OF STUDY

This thesis is comprised of four chapters. Chapter II reviews the previous work by the Goodrich Corporation Fuel&Utility Systems (Bechhoefer, 2003) and describes the dataset used for the analysis. It also explains the statistical models and techniques used for the study. Chapter III describes univariate Box-Jenkins (ARMA) modeling with regression analysis, Mahalanobis metric analysis, goodness-of-fit test analysis and the Bonferroni correction procedure. Chapter IV summarizes the conclusions of the analysis and presents recommendations for further study.

II. DATA AND METHODOLOGY

A. PREVIOUS STUDY AND DATASET

1. Previous Study

Data acquisitions are made by the Integrated Mechanical Diagnostic-Health and Usage Management Systems (IMD HUMS) installed on CH-53 aircraft. An accelerometer mounted closest to the component sends a signal that is used to measure the vibrations of the component. The acquired vibration data is then processed in the vibration processing unit (VPU). The VPU is used to calculate a HI based on CIs. The VPU can process up to eight channels at a time. Each channel process four seconds of acquired data in about one minute (Goodrich Corporation Fuel & Utility Systems, 2003).

A desired vibration threshold setting for each particular gear is high enough so that even a healthy aircraft with the most aged gears does not indicate false alarms. One method for setting the threshold values for warning and alarm is to model the variance between aircraft and to add a correction for different predefined ranges of torque (torque bands). Initially, the least squares method is applied to the CI values which are assumed to be randomly sampled from a seven-dimensional normal distribution. This method uses the data coded into an information matrix format organized by aircraft type and torque bands. After the least squares fit method is applied, the estimated condition indicators $(\hat{CI})$ and the sample variance for each CI are calculated. An adjustment is made for additional components of variance arising from selection of the sample’s aircraft from the population. These calculations use assumptions of normality, independence and homoscedasticity. A CI is considered to be in a “Warning” state when its value is three standard deviations above the mean. The computation of the standard deviation includes an adjustment for variability between aircraft and between torque bands: the value of three is chosen from Normal theory. Similarly, a CI is considered to be in an “Alarm” state if the value is six standard deviations above the mean (Bechhoefer, personal communication, October 01, 2003).

The HI of a component is calculated by taking a consensus of a particular part’s CIs. As in the case of gears, there are seven CIs to take into account. These seven CIs for each particular gear are given in Table 1.

Condition Indicator Name	**Variable Name**
Residual Kurtosis	Residual_kurtosis
Residual Root-Mean-Square (RMS)	Residual_rms
Gear Distributed Fault	GearDisFault
Frequency Module Peak-to-Peak	fmP2P
Sideband Modulation 1	sm_1
Sideband Modulation 2	sm_2
Signal Average Ratio RMS	sigAvg_rms

Table 1. List of Seven CIs for Each Particular Gear

If three of the seven CIs exceed the normal mean level by three standard deviations or more, the component is in a

“Warning” state. Similarly, if three of the seven CIs exceed the normal mean level by more than six standard deviations, the component is in an “Alarm” state. The study shows that visibly damaged parts typically have CI values 6 to 8 standard deviations larger than the normal mean level. Severely damaged parts have CI values which are at least 12 standard deviations above the normal mean level (Goodrich Corporation Fuel & Utility Systems, 2003).

The current approach, however, makes assumptions about the data that are untenable. The assumption that each CI follows the normal distribution (conditional on aircraft and torque bands) has not been tested. The creation of torque bands discards some information; presumably, by considering torque to be continuous, we can better exploit that data. Most seriously, the current approach’s computations assume that the data are like independent random samples, whereas in reality there is a strong element of time-dependence within each set of data (See Chapter III).

2. Data Used in the Analysis

The data set consists of 23,187 acquisitions and 20 variables. These variables are:

- Tail Number
- Accelerometer Name
- Torque
- Gear Name
- Gear Index
- Accelerometer Signal to Noise Ratio (SNR)
- Accelerometer Root-Mean-Square (RMS)
- Accelerometer Clipping

- Accelerometer Low Frequency Intercept
- Accelerometer Low Frequency Slope
- Accelerometer Analog-to-Digital Converter (ADC) Bits Used
- Accelerometer Dynamic Range
- Residual Kurtosis
- Residual RMS
- Gear Distributed Fault
- Frequency module peak to peak
- Side Modulation 1-2
- Signal Average RMS

Tail: This variable consists of the tail number of each aircraft for each acquisition. Table 2 provides a list of the sample sizes for each tail number.

Tail Number	Total Acquisitions
162494	5934
163075	2437
163086	3461
164539	11335

Table 2. Number of Acquisitions for Each Tail Number

Accelerometer Name/Part: The dataset includes acquisitions from 21 different accelerometer names and part names, which are represented in Table 3.

Accelerometer Name	Accelerometer Part	Number of Acquisitions
AGBAft	DTA30	1532
AGBFwd	DTA29	2681
IGBInput	DTA32	52
IGBOutput	DTA07	1392
MGBRear	DTA23	1062
No2Input	DTA12	3186
OilCooler	DTA22	354
OilCoolerTakeOff	DTA28	354
PortInputHanger	DTA13	609
PortMain	DTA18	850
PortNGBInput	DTA08	609
PortNGBOilCooler	DTA24	1915
PortNGBOutput	DTA10	1218
PortRing	DTA16	1360
TbdMain	DTA19	170
TbdNGBInput	DTA11	609
StbdNGBOilCooler	DTA25	1149
StbdNGBOutput	DTA09	609
TGBInput	DTA31	208
TGBOutput	DTA05	696
TailTakeOff	DTA06	2572

Table 3. List of Accelerometer Names and Parts

Torque: Torque is a force or system of forces that tend to cause rotation. The data includes the different torque levels applied by each helicopter during the operational test flights.

Gear Name/Index: These two variables include 63 type of gears and the associated index numbers of those gears. Table 4 provides the list of gear names and the total number of acquisitions for each of those gears.

Gear Name	**Size**	**Gear Name**	**Size**	**Gear Name**	**Size**
#2EngFCDrvShftSpur	354	AuxLbVnPmpShftBlades	170	PortNGBTachShftSpur	383
#2EngFrWhShftCamGear	354	AuxLbVnPmpShftGear	170	RrCovIdlerShftIdler	354
#2EngFrWhShftDrvSpur	354	GrndStg1Ring	170	SmpRotPmpShftBlades	170
#2EngFrWhShftSpur	354	GrndStg2Ring	170	SmpRotPmpShftGear	170
#2EngInpShftSpur	354	IGBInpShftPin	52	StbdAftInpDrvShftPin	170
#2EngTachShftSpur	354	IGBOutShftGear	696	StbdNGBEngInpShftPin	609
#2GenShftSpur	354	IGBOutShftPumpBlades	696	StbdNGBFCDrvShftGear	383
#2InpShftAftIdler	354	MainRtrShftOPSpur	170	StbdNGBOPDrvShftSpur	383
#2InpShftIdler	354	MainRtrTachShftSpur	354	StbdNGBOutShftPin	609
#2InpShftPin	354	OilCoolShftSpur	354	StbdNGBTachShftSpur	383
AGBActShftIdler	383	OuterShaftMainBev	170	Stg1HydPmpShftSpur	354
AGBActShftSpur	383	OuterShaftSunGear	170	Stg1PlntShftGear	170
AGBDrvShftGear	383	PortAftInpDrvShftACCPi	609	Stg2PlntShftGear	170
AGBDrvShftSpur	383	PortAftInpDrvShftPin	170	Stg2SunShftGear	170
AGBEngStrtShftSpur	383	PortNGBEngInpShftPin	609	TRTakeoffShftSpur	1286
AGBGen#1ShftSpur	383	PortNGBFCDrvShftGear	383	TGBInpShftGear	52
AGBGen#3ShftSpur	383	PortNGBFCDrvShftLHZerl	383	TGBInpShftPin	52
AGBOPShftSpur	383	PortNGBFCDvnShftLHZerl	383	TGBOilPmpShftBlades	52
AGBStg2SrvPmpShftSpur	383	PortNGBOPDrvShftSpur	383	TGBOilPmpShftGear	52
AGBUtPmpShftSpur	383	PortNGBOutShftACCSpur	609	TGBOutShftGear	696
AGBWchPmpShftSpur	383	PortNGBOutShftGear	609	TTOIdlerShaftIdlerSpur	1286

Table 4. Gear Names and Number of Acquisitions

For the remaining variables the text from Goodrich Corporation Fuel & Utility Systems (2003) is attached.

Signal to Noise Ratio (SNR): Each data channel has a specified observed SNR associated with it. Before the vibration data is calibrated, a power spectral density is calculated from the data set. Each component in the data channel has known frequencies associated with it. SNR measures the excess strength of a known tone (corrected for operational speed differences) above the minimum baseline levels in a user-defined bandwidth.

Root Mean Square (RMS): The overall energy level of the vibration is represented by the RMS value of the raw vibration amplitude. Major overall changes in the vibration level can be detected by the RMS value.

Clipping: For a specific gain value, the raw ADC bit values cannot exceed a specific calculated value. There is no clipping in the data used in this analysis.

Frequency Slope and Low Frequency Intercept: These CIs were installed in the algorithm per Navy request. Using the first 10 points of the power spectral density estimated from the raw data, a simple linear regression is performed to obtain the intercept and slope in the frequency-amplitude domain.

ADC Bit Use: ADC Bit Use measures the number of ADC bits used in the current acquisition. The ADC board is typically a 16 bit processor. The log base 2 value of the maximum raw data bit acquired is rounded up to the next highest integer. Channels with inadequate dynamic range typically use less than 6 bits to represent the entire dynamic range.

ADC Sensor Range: ADC Sensor Range is the maximum range of the raw acquired data. This range cannot exceed the operational range of the ADC board, and the threshold value of 32500 is just below the maximum permissible value of +32767 or -32768 when the absolute value is taken.

Dynamic Range: Dynamic Range is similar in spirit to the ADC Sensor Range, except the indicator reports dynamic channel range as a percent rather than a fixed bit number.

Kurtosis: The fourth moment (Kurtosis) of the distribution has the ability to enhance the sensitivity of tail changes. It has a value of 3 (Gaussian distribution) when the machinery is healthy. Kurtosis values, larger than 3.5, are usually an indication of localized defects. However, distributed defects such as wear tend to smooth the distribution and thus decrease the Kurtosis values.

Gear Distributed Fault (GDF): GDF is thought to be an effective detector for distributed gear faults such as wear and multiple tooth cracks. GDF is calculated from the formula below

$$GDF = \frac{StdDev(RS)}{StdDev(A1)}$$

RS = residual data

A1 = signal average

Peak-To-Peak (P2P): The Peak-To-Peak value of the raw vibrating amplitude represents the difference between the two vibration extreme. When failures occur, the vibration amplitude tends to increase in both upward and downward directions and thus the Peak-To-Peak value increases.

Sideband Modulation (SM): SM analysis is designed to reveal any sideband activities that may be the results of certain gear faults such as eccentricity, misalignment, or looseness (Goodrich Corporation Fuel & Utility Systems, 2003).

B. METHODOLOGY

The goal of this analysis is to compute a threshold value for each particular gear type and tail number, so that a single numerical value can be used to track the wear on each gear. In order to calculate this threshold value, a new data set of a single gear and tail number was created from the whole data set. This was done using the make.datanew function in S-PLUS. The code for this function is presented in Appendix A. This function created 252 different data sets from the 63 gear types and four unique tail numbers. Each of the seven CIs (See Table 1) for each data set was considered to behave as a univariate time series.

1. Univariate Time Series

Since the data was obtained continuously over a time interval, each of the CIs was assumed to be equally spaced in time and to exhibit univariate time series behavior.

A "univariate time series" consists of scalar observations recorded sequentially with equal time intervals between observations. For ease of analysis, univariate time series data sets are usually displayed in column form. In a univariate time series, time is an implicit variable. Properties of a time series data set, such as stationarity, seasonality and trend, must be considered before starting the analysis (NIST SEMATECH, 2003).

a. Stationarity

Stationarity is often assumed for data that follows a time series pattern. Under the stationarity assumption, the mean, variance and autocorrelation structure remain constant over time. Graphically, stationary series exhibit no apparent trends. Time plots are very useful because nonstationarity can often be detected from a study of the plot (NIST SEMATECH, 2003).

For our study, time plots were used to examine if each CI data for a particular gear and tail number is stationary or not. To draw the time plot for each gear CI, the “timeplot” function in Appendix A was used.

b. Seasonality

Seasonality refers to the periodic fluctuations in a data set. We tested for seasonality since we are dealing with a time series. If the presence of seasonality is observed it must be considered in our time series model.

There are several graphical methods with which to detect the presence of seasonality. These include time plots, seasonal sub-series plots and multiple box plots. The analyst must know the seasonal period to be able to use sub-series plots or multiple box plots. For our data, the seasonal period is unknown; therefore the preferred method was to use time plots. An alternate course of action would be to use the autocorrelation plot to detect seasonality. If there are seasonality spikes (sudden increases) in the plot, they can be observed at lags equal to the period (NIST SEMATECH, 2003).

c. Trend

A trend in a data is the movement in a direction over a long-term period of time. It is defined by the added influence of many factors that will affect the time series in a consistent and gradual way over a long period of time (Ragsdale, 2001, p. 509). We used time plots to detect the presence of trends in our data sets.

d. Time Plots

Once the background information was gathered and the objectives are carefully defined, the next and most important step was to plot the data versus time. Time plots graphically summarize a univariate data set in a way that makes it easy to analyze and understand characteristics of the data set. Characteristics that can be detected from time plots include trend, seasonality, outliers, and discontinuities. The time plot is also a very useful tool for the analyst, because it assists in describing the data and formulating a plausible model (Chatfield, 1996, p.11). Time plots are formed by using the time variable on the horizontal axis, and a response variable on the vertical axis.

For our study, we plotted every gear CI of a data set for a particular gear type and tail number using the "timeplot" function in Appendix A to detect seasonality or trends.

2. Autocorrelation

In a time series model, there is often correlation between observations at different time lags. These correlations are estimated by sample autocorrelation coefficients, which can be used to provide insights into the probability model from which the data may have been drawn. Given N pairs of observations on two variables x and y, the correlation coefficient is

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\left[\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2\right]}} \qquad (1)$$

This same idea can be applied to time series models to check for correlation between successive CI observations (Chatfield, 1996, p.19).

If we have *N* pairs of CI observations such as $(X_1, X_2),\ (X_2, X_3), \ldots, (X_{N-1}, X_N)$, the first order correlation coefficient between X_t and X_{t+1} is given by

$$r_1 = \frac{\sum_{t=1}^{N-1}\left(x_t - \overline{x}_{(1)}\right)\left(x_{t+1} - \overline{x}_{(2)}\right)}{\sqrt{\left[\sum_{t=1}^{N-1}\left(x_t - \overline{x}_{(1)}\right)^2 \sum_{t=1}^{N-1}\left(x_{t+1} - \overline{x}_{(2)}\right)^2\right]}} \tag{2}$$

where the mean of the first and last *N-1* CI observations are

$$\overline{x}_{(1)} = \sum_{t=1}^{N-1} x_t / \left(N-1\right) \tag{3}$$

$$\overline{x}_{(2)} = \sum_{t=2}^{N} x_t / \left(N-1\right) \tag{4}$$

respectively. The correlation between successive CI observations is called an autocorrelation coefficient (Chatfield, 1996, p.19).

Since $\overline{x}_{(1)} \cong \overline{x}_{(2)}$ and *N/(N-1)* gets close to one for large sample sizes, a simpler formula can be given by

$$r_1 = \frac{\sum_{t=1}^{N-1}\left(x_t - \overline{x}\right)\left(x_{t+1} - \overline{x}\right)}{\sum_{t=1}^{N}\left(x_t - \overline{x}\right)^2} \tag{5}$$

Similarly the correlation between CI observations a distance *k* apart is given by

$$r_k = \frac{\sum_{t=1}^{N-k}\left(x_t - \overline{x}\right)\left(x_{t+k} - \overline{x}\right)}{\sum_{t=1}^{N}\left(x_t - \overline{x}\right)^2} \tag{6}$$

This is called the autocorrelation coefficient at lag *k* (Chatfield, 1996, pp.19-20).

In our study, we used autocorrelation to identify an appropriate time series model. To accomplish this we plotted autocorrelation functions for each CI varying the number of lags. There are two types of graphical methods that show autocorrelations.

a. Autocorrelation Plots

In this study, we used autocorrelation plots to identify the order of a moving average model (MA) (See Section B.3). To draw the autocorrelation plots of each CI, the “draw.acf.plots” function in Appendix A was used.

Autocorrelation plots lead us to discover where the function approaches a zero value and ultimately the order of the Moving Average (MA) model, which is denoted as q (NIST SEMATECH, 2003).

b. Partial Autocorrelation Plots

The partial autocorrelation at lag *k* is the autocorrelation between X_t and X_{t-k} not conveyed through the intervening values. The autoregressive (AR) (See Section B.3) order of a Box-Jenkins (ARMA) model is commonly identified through the use of partial autocorrelation plots (NIST SEMATECH, 2003). Detailed information about the partial autocorrelation function can be found in Brockwell and Davis (1996).

In our study “draw.acf.plots” function in Appendix A was used to draw partial autocorrelation plots of each CI. We determined the order of an AR model by examining the lag where the function approached zero. Details about calculating the order of AR model can be found in Section B.3 (NIST SEMATECH, 2003).

3. Time Series Models

a. Autoregressive (AR) Models

A commonly used approach for modeling univariate time series is through applying an AR model to the series. The general form of the AR model applied is AR(*p*):

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + . \; . \; . + \phi_p X_{t-p} + \varepsilon_t \qquad (7)$$

where X_t is the time series and ε_t is a white noise series. An autoregressive model can simply be thought of as a linear regression relationship between the current value and one or more prior values of the series. The order of the AR model is known as *p* (NIST SEMATECH, 2003).

Examining the partial autocorrelation plots leads us to discover where the function approaches a zero value. Since the AR(*p*) process becomes zero at lag *p+1* and greater, we can now deduce the value of *p*. If an AR model is shown to be appropriate from the analysis of a sample autocorrelation plot, then we can use the analysis of the sample partial autocorrelation plot to help identify the order of the AR model. For this study the range of values within the 95% confidence intervals are accepted as zero values (NIST SEMATECH, 2003).

After examining the partial autocorrelation plots of each CI for each data set of a particular gear and tail number, the order of the AR model was determined to often be *p=1*. The AR(1) is given by

$$X_t = \phi \, X_{t-1} + \varepsilon_t \qquad (8)$$

In an AR(1) model, *x* depends on the value it previously held. This characteristic should prevent large jump sizes from X_{t-1} to X_t. The value of ϕ should be between -1 and 1 (Chatfield, 1996, p.35). For this study we chose ϕ = 0.8 as a starting value because it usually led to convergence in the software.

b. Moving Average (MA) Models

The MA(q) model is described as follows:

$$X_t = \theta_0 \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - . . . - \theta_q \varepsilon_{t-q} \qquad (9)$$

where X_t is the time series, ε_{t-q} are white noise, and θ_1, . . . , θ_q are the parameters of the model (Chatfield, 1996, p.33).

Examining the autocorrelation plots leads us to discover where the function approaches a zero value. Since the MA(q) process becomes zero at lag *q+1* and greater, we can now calculate the value of *q*. For this study the range of values within the 95% confidence intervals are accepted as zero values (NIST SEMATECH, 2003).

After examining the autocorrelation plots of each CI for each data set of a particular gear, and tail number, the order of the MA model was determined to often be one. The moving average model of order one, which is MA(1), is given by:

$$X_t = \varepsilon_t - \theta_1 \varepsilon_{t-1} \qquad (10)$$

In an MA(1) model, X_t depends on the value of the immediate past error, which is known at time *t*. This characteristic should prevent large jump sizes from X_{t-1} to X_t (Chatfield, 1996, p.34). The value of θ should be between -1 and 1. For this study we used the value of θ where the optimizer converged for most of the CI models, which was θ = 0.2.

Through the analysis of the autocorrelation function (ACF) plots and the partial autocorrelation function (PACF) plots, we observed that a combination of these two models (AR and MA) would best fit to each CI data of particular gear type and tail number. Therefore we applied Box-Jenkins (ARMA) Models.

c. Box-Jenkins Models (ARMA)

The Box-Jenkins ARMA (Autoregressive Integrated Moving Average) model is a combination of the AR and MA models previously discussed. The first main assumption of the Box-Jenkins ARMA models is that the time series is stationary. So we must first ensure that there is stationarity in all of the univariate CI time series. If non-stationarity is observed in the time series data, Box and Jenkins recommend a process called differencing that can be applied one or more times to achieve stationarity (NIST SEMATECH, 2003).

For our study, each CI of a data set for a particular gear and tail number was plotted against time. It was observed from these plots that each particular data set was stationary or almost so. Therefore, we did not need to apply the differencing process.

The general form of an ARIMA model is ARIMA(p,d,q). Since the data exhibited no apparent deviations from the stationarity, we fit an ARMA model setting the differencing value to zero. The general ARMA(p,q) model is given as:

$$X_t - \phi_1 X_{t-1} - \phi_2 X_{t-2} - \ldots - \phi_p X_{t-p} = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \ldots - \theta_q \varepsilon_{t-q} \qquad (11)$$

Now that we have a model without differencing, we need to identify the orders (i.e., the *p* and *q*) of the autoregressive and moving average terms. After examining ACF and PACF plots for every CI of a particular gear type and tail number, both p and q values were estimated as one. Therefore to model each gear CI data, we used an ARMA (1,1) model given by

$$X_t = \phi X_{t-1} + \varepsilon_t - \theta \varepsilon_{t-1} \qquad (12)$$

To model the univariate time series model for each CI, we used the "arima.mle" function that is built in to S-PLUS®.

Providing the starting values of the ARMA model parameters (ϕ and θ) is necessary for the optimizer. Poor starting values can lead to slow convergence to a local maximum (S-PLUS 2000 Guide to Statistics, Volume 2, pp.177, 1999).

d. ARMA Models with Regression Variable

At this point, we added the torque effect to our model. It is believed that there is a relation between the torque and the CI levels. Therefore, we added torque as a regressor variable to each univariate time series model for each CI. To accomplish this, the S-PLUS function "arima.mle" was used. This function allows us to add torque as an additive regressor variable to our models via the "xreg" optional argument. After adding the torque effect, our general model is given by

$$X_t = \phi X_{t-1} + \varepsilon_t - \theta \varepsilon_{t-1} + \beta_1 + \beta_2 T_t \qquad (13)$$

where β_1 is the intercept, β_2 is the slope and T_t is the torque value at time t for the regression model. Detailed information about ARMA models with regression variables can be found in S-PLUS 2000 Guide to Statistics, Vol 2, pp.173, 1999).

e. Standardized Residuals

A residual is defined as the difference between the observed value and the fitted value. A standardized residual equals the residual divided by its estimated standard error.

So far, we have modeled each CI for every data set of a particular gear type and unique tail number. Since the variances of every modeled CI vary considerably from one CI to another, it is rather difficult to know whether a fitted residual should be considered large or small. Therefore, we use standardized residuals, which are independent of the units of measurement of the variables. In particular, standardized residuals provide a statistical metric for determining the size of a residual for each CI. Because of this fact, we decided to use the standardized residuals of each modeled CI as our new CIs for the rest of the analysis. Therefore, we created new CI matrices for each particular gear and tail number data set using the function “make.newci” in Appendix A (SSI Scientific Software, 2003).

After modeling each of the seven CIs (See Table 1 in Chapter II) given in Table 1 for each data set, the standardized residuals of each CI model were saved. The seven standardized residual vectors were then used to create new CI matrices for each data set corresponding to a particular gear type and tail number.

4. Mahalanobis Metric

The general form of the Mahalanobis metric is given by

$$r^2 = \left(X - M_x\right)' C_x^{-1} \left(X - M_x\right) \tag{14}$$

where X is the new CI matrix consisting of seven CI (See Table 1 in Chapter II) vectors,

$$X = \begin{bmatrix} CI_1(1), & . \; . \; . \; , CI_7(1) \\ . & . \\ . & . \\ CI_1(n), & . \; . \; . \; , CI_7(n) \end{bmatrix} \tag{15}$$

n : number of the acquisitions for each CI

CI_1 : Residual kurtosis

CI_2 : Residual rms

CI_3 : Gear Distributed Fault

CI_4 : Fm Peak-to-Peak

CI_5 : Side modulation 1

CI_6 : Side modulation 2 and

CI_7 : Signal Average rms

The mean of the CI_j is represented by

$$\hat{\mu}_j = \frac{\sum_{i=1}^{n} CI_j(i)}{n} \tag{16}$$

The mean vector M_X is

$$M_X = \begin{bmatrix} \mu_1 & \mu_2 & \mu_3 & \mu_4 & \mu_5 & \mu_6 & \mu_7 \end{bmatrix}' \tag{17}$$

and the covariance matrix C is given by

$$Cov(X) = C_X = \begin{bmatrix} Cov(1,1) & . & . & . & Cov(1,7) \\ Cov(2,1) & . & . & . & Cov(2,7) \\ . & & . & & . \\ . & & & . & . \\ . & & & & . \\ Cov(7,1) & . & . & . & Cov(7,7) \end{bmatrix}. \qquad (18)$$

The Mahalanobis metric is commonly used to detect outliers in a multivariate data set which includes two or more variables of interest (dependent variable). The Mahalanobis metric does not treat all CI values equally when calculating the distance from the mean vector; instead it weights the differences by the range of variability and by the vectors' covariances. The Mahalanobis measurement is also useful for discrimination since the distances are calculated in units of standard deviations from the mean vector (Thermo Galactic, 2003).

Our analysis is based on detecting any unusual level in CI values relating to pre-existing machining or manufacturing-induced defects, poor operating conditions (loss of lubrication), foreign object damage, environmental factors (corrosive environments and resulting pitting damage) or excessive loading. We know that no failure occurred during the collection of each data set for each particular gear type and tail number. Therefore, by calculating the Mahalanobis distances for each of these data sets, we gain insight about the expected range of the Mahalanobis distances. If the Mahalanobis distance of any acquisition is bigger than a given threshold value, we can conclude that there might be a defect in that particular gear.

After calculating the Mahalanobis distances for each data set, we must then find a specific distribution which best fits the set of Mahalanobis distance vectors of all data sets. By accomplishing this we can set a Mahalanobis threshold value for each particular gear type and tail number. Then this threshold value will be used to detect any defective gears.

5. Goodness-of-Fit Tests

a. Chi-Square Goodness-of-Fit Test

To set a threshold value for each gear type, we need to fit the Mahalanobis distances to a specific distribution. In this study, the chi-square goodness-of-fit test was used to test if the calculated Mahalanobis distances of each data set for a particular gear type and tail number fit to an exponential distribution.

The chi-square goodness-of-fit test can be applied to any univariate distribution for which the cumulative distribution function can be calculated. Chi-square goodness-of-fit test is applied to binned data (NIST SEMATECH, 2003).

The chi-square test null hypothesis and alternative hypothesis are

Ho: The data follows a specific distribution

Ha: The data does not follow a specific distribution.

The chi-square goodness of fit computation uses the following test statistic:

$$\chi^2 = \sum_{t=1}^{k} \left(O_i - E_i\right)^2 / E_i \qquad (19)$$

where *k* is the number of bins, O_i is observed frequency for bin *i* and E_i is the expected frequency for bin *i*. The expected frequency is calculated by using

$$E_i = N(F(Y_u) - (F(Y_l))\qquad(20)$$

where the cumulative distribution function for the distribution being tested is F, the upper limit for class *i* is Y_u, the lower limit for class *i* is Y_l, and the sample size is *N*.

The null hypothesis was accepted if

$$\chi^2 < \chi^2_{(\alpha,k-c-1)}$$

where *c* is the number of estimated parameters and $\chi^2_{(\alpha,k-c-1)}$ is the critical value from the chi-square distribution with *k*-*c* degrees of freedom and a significance level of α. (NIST SEMATECH, 2003).

b. Kolmogorov-Smirnov Goodness-of-Fit Test

The Kolmogorov-Smirnov (K-S) test is an alternative goodness-of-fit test that is used to decide if a sample comes from a population with a specific distribution. For our study, the K-S goodness-of-fit test was also used as an alternative to test if the calculated Mahalanobis distances of each data set for a particular gear and tail number fit an exponential distribution.

The K-S test is based on the empirical distribution function. Given *N* ordered data points $Y_1, Y_2, Y_3, \ldots, Y_N$, the empirical distribution function is defined as

$$E_N = n(i)/N\qquad(21)$$

where $n(i)$ is the number of points less than Y_i and the Y_i are ordered from smallest the largest value. This is a step function that increases by *1/N* at the value of each ordered data point. An attractive feature of this test is that the distribution of the K-S test statistic itself below does not depend on the underlying cumulative distribution function being tested. Despite this advantage, the K-S test has several important limitations:

- It only applies to continuous distributions.
- It tends to be more sensitive near the center of the distribution than at the tails (NIST SEMATECH, 2003).

The K-S test null hypothesis and alternative hypothesis are

Ho: The data follows a specific distribution.

Ha: The data does not follow a specific distribution.

The K-S test statistic is defined as

$$D = \max_{i \le N} \left| F(Y_i) - \frac{i}{N} \right| \qquad (22)$$

where *F* is the theoretical cumulative distribution of the continuous distribution being tested. The hypothesis regarding the distributional form is rejected if the test statistic, *D*, is greater than the critical value obtained from a table. There are several variations of these tables in the literature that use somewhat different scaling for the K-S test statistic and critical regions. These alternative formulations should be equivalent, but it is necessary to ensure that the test statistic is calculated in a way that is consistent with how the critical values were tabulated (NIST SEMATECH, 2003).

6. Bonferroni Correction/Adjustment Procedure

In our study, we performed 224 goodness-of-fit tests. Conventionally, the α level is set at 0.05 for each Chi square and K-S goodness-of-fit test.

If we perform more than one statistical test, the probability of observing at least one test statistically significant due to chance fluctuation, and to incorrectly declare a difference to be true (Type I error), increases (Simple Interactive Statistical Analysis (SISA), 2003).

Since we performed a total of 224 hypothesis tests, the probability of making Type I error increases from the conventional value of .05. Our purpose is to control the Type I error, the decision to reject the null hypothesis (Ho: The Mahalanobis data set follows a specific distribution) when it is, in fact, true.

The Bonferroni is used when more than one statistical test in a particular study are being performed simultaneously. The Bonferroni correction procedure adjusts the α level of each individual test downwards to ensure that the overall risk for a number of tests remains 0.05. To accomplish this, instead of using the α significance level for each test in an entire set of n comparisons, the Bonferroni correction sets the α value for each test to α/n (Weistein, 2003).

In our study, the Bonferroni adjusted level of significance was calculated as $0.05/224 \approx 0.0002193$. The null hypothesis was rejected for any test that resulted in a probability of less than *0.0002193* which was statistically significant. The null hypothesis was accepted for the tests with a probability value greater than *0.0002193.* See Chapter III, Section C, for results of this analysis.

THIS PAGE INTENTIONALLY LEFT BLANK

III. ANALYSIS

A. UNIVARIATE BOX-JENKINS (ARMA) MODELLING ANALYSIS

1. Model Identification Analysis

a. Stationarity, Seasonality and Trend Analysis

The first step in developing a time series model is to determine if the series is stationary and if there is any significant seasonality or trend that needs to be modeled.

Using the “timeplot” function in Appendix A, each of the CIs for a particular gear type and tail number was plotted against time. After examining each of these plots, it was observed that almost all of them were plausibly stationary. Since non-stationarity was not observed in our univariate time series data sets, we did not need to use differencing. Figures 1 through 3 provide a few examples of these plots indicating stationarity.

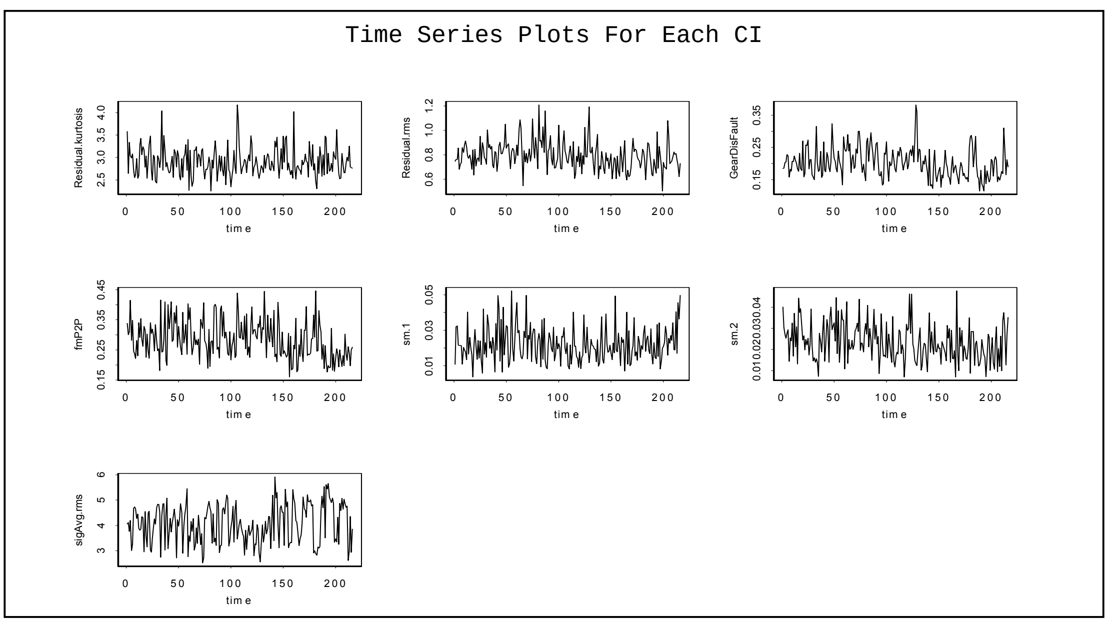

Figure 1. Time Plots of The CIs For Gear Type “AGB Wch Pmp Shft Spur” and Tail Number “164539”

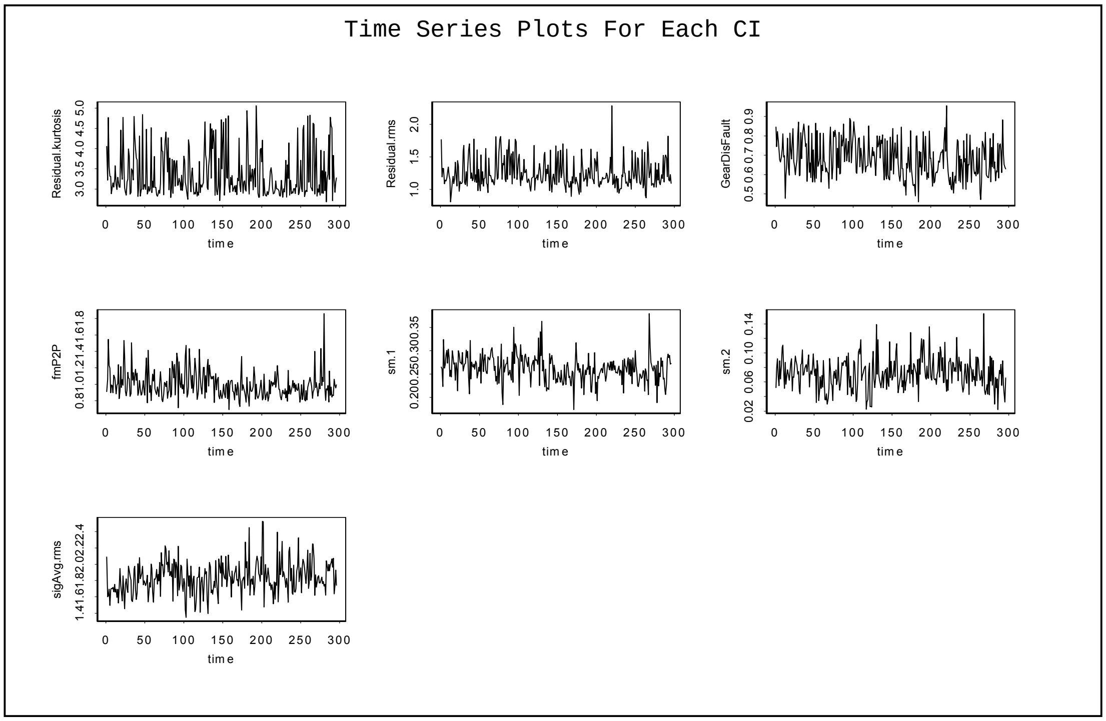

Figure 2. Time Plots of The CIs For Gear Type "TGB Out Shft Gear" and Tail Number “164539”

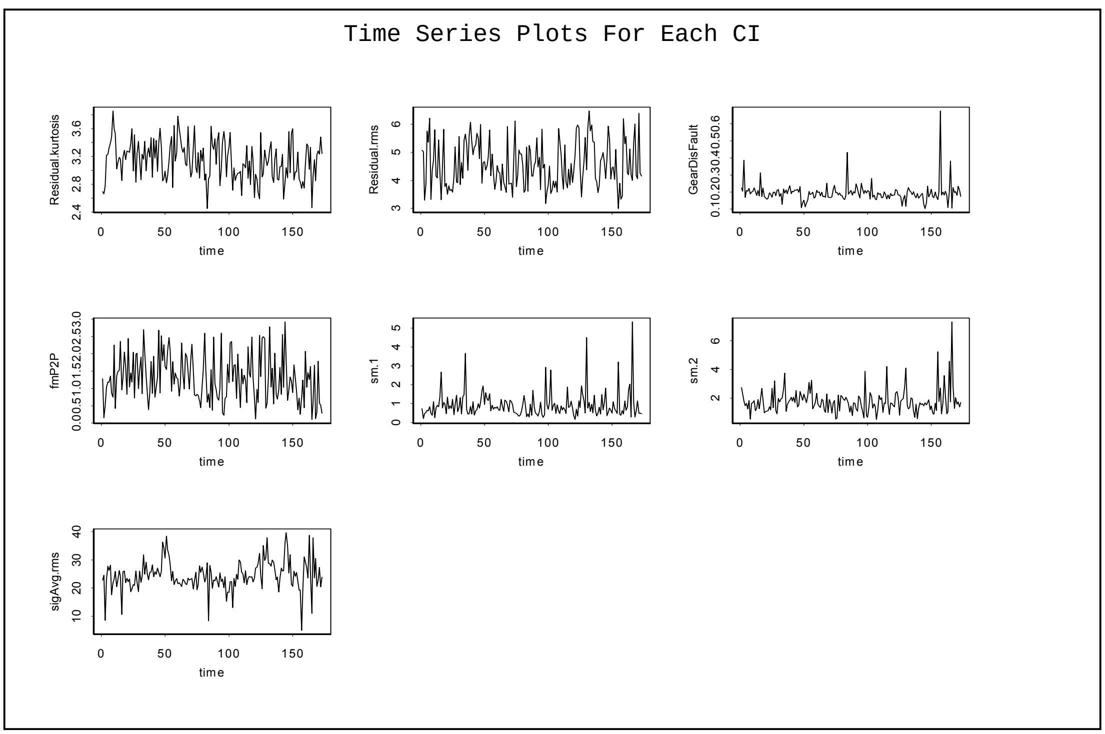

Figure 3. Time Plots of The CIs For Gear Type "IGB Out Shft Pump Blades" and Tail Number “162494”

b. Autoregression (AR) and Moving Average (MA) Order Analysis

Since the above time series plots of each CI and the others for particular gear type and tail number did not exhibit any significant non-stationarity or seasonality, we generated the autocorrelation and partial autocorrelation plots of the raw data to decide about the orders ARMA(p,q) models. For this purpose, we used the “plot.acf.plots” function in Appendix A. Figures 4 through 6 provide a few examples of these plots.

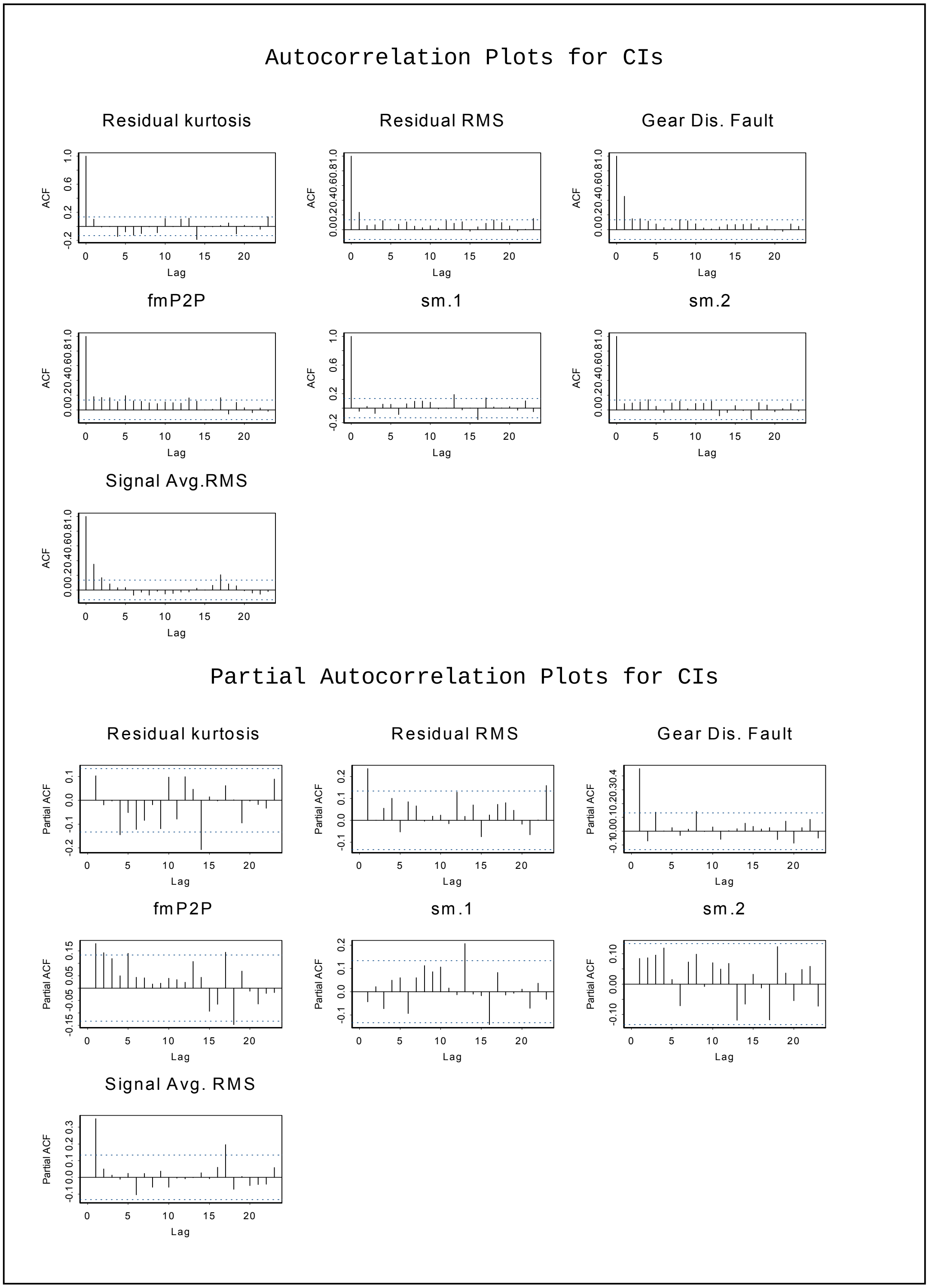

Figure 4. Autocorrelation and Partial Autocorrelation Plots for CIs of Gear Type “AGB Wch Pmp Shft Spur” and Tail Number “164539”

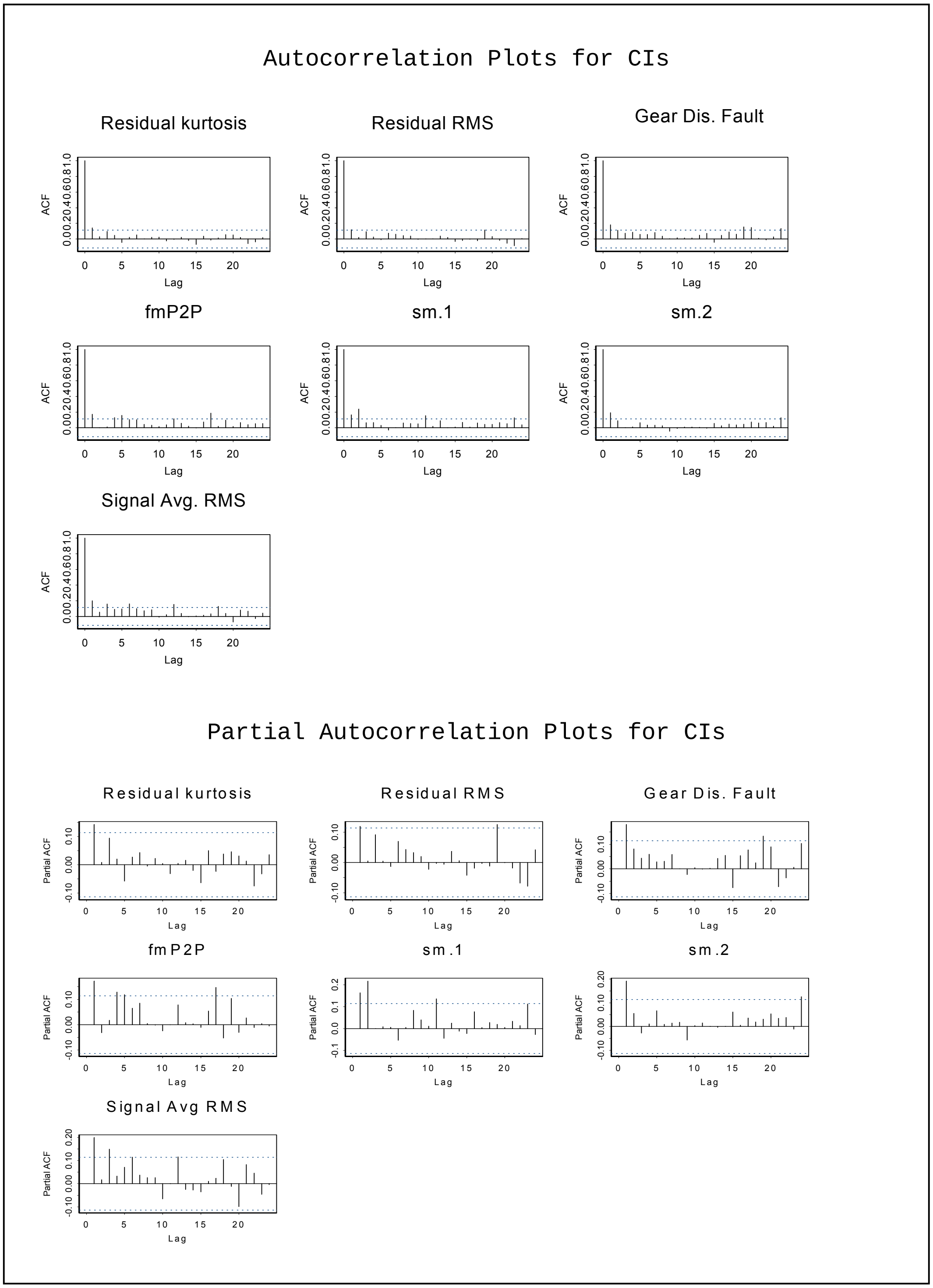

Figure 5. Autocorrelation and Partial Autocorrelation Plots for CIs of Gear Type "TGB Out Shft Gear" and Tail Number “164539”

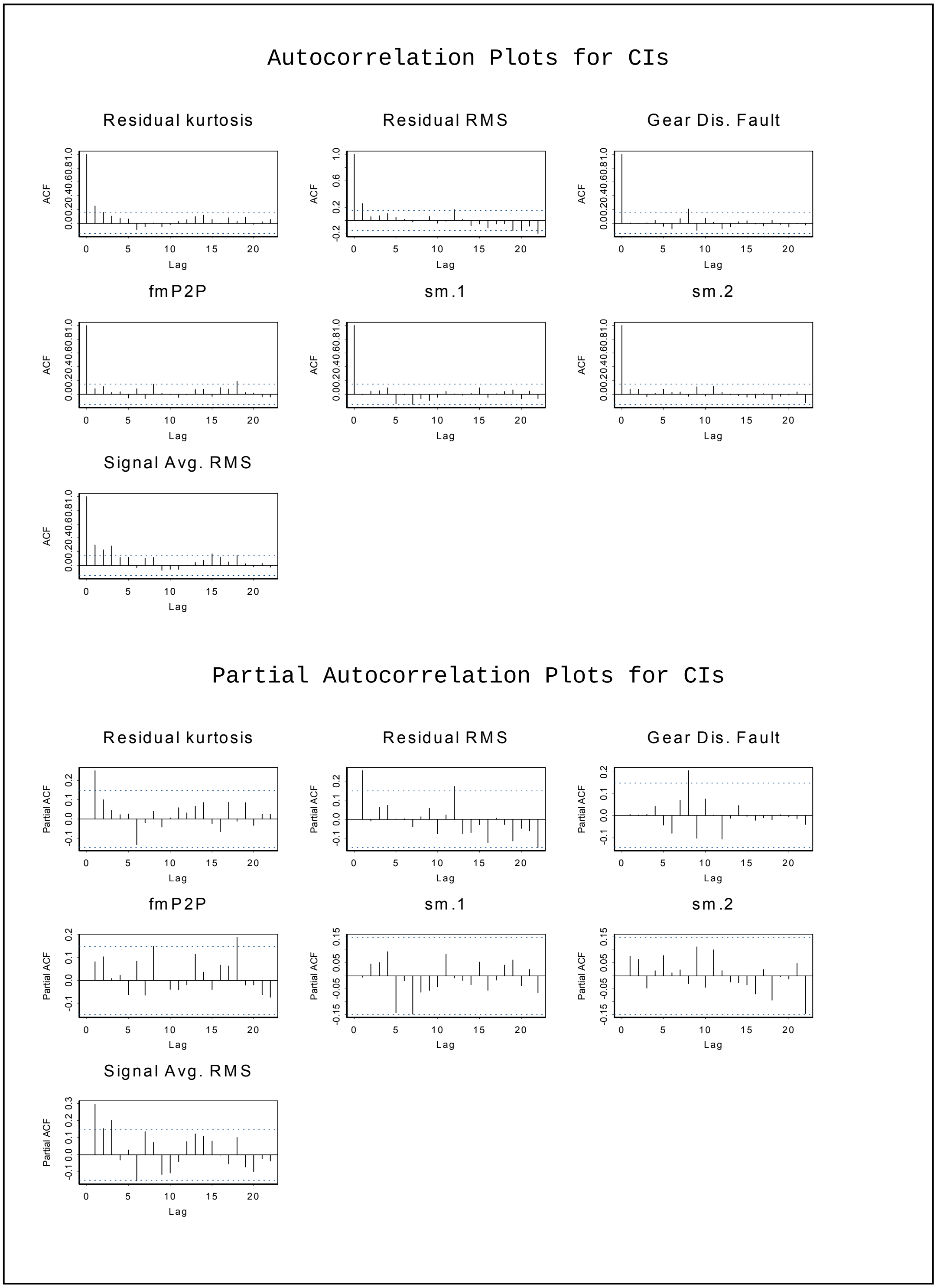

Figure 6. Autocorrelation and Partial Autocorrelation Plots for CIs of Gear Type "IGB Out Shft Pump Blades" and Tail Number "162494"

Autocorrelation function plots display the coefficients starting from lag 0 to lag 25. Dashed lines mark off approximate 95% confidence bands. Most of the autocorrelation and partial autocorrelation plots with a 95% confidence band showed that the autocorrelation at lag 1 was significant. Based on these plots, ARMA(1,1), was suggested. For convenience we used a single model $X_t = \phi X_{t-1} + \varepsilon_t - \theta \varepsilon_{t-1}$ for every CI of gear and tail number.

Since it is believed that torque affects the CI levels, we added torque effect to our single ARMA(1,1) model as a regression variable and the single model changed to $X_t = \phi X_{t-1} + \varepsilon_t - \theta \varepsilon_{t-1} + \beta_1 + \beta_2 T_t$. See Chapter II, Section B.3 for details. Then, we modeled 252 univariate CI time series for 63 different types of gears and 4 different tail numbered aircraft.

2. Model Validation Analysis

Having developed the models, diagnostics were checked to determine if the models were reasonable. Specifically, standardized residual plots were analyzed to determine if ARMA(1,1) with regression variable models were valid models.

A plot of the standardized residuals over time is the single most important diagnostic for time series model validation. By examining the standardized residual plots, we can detect outliers, non-homogeneity of variance, and obvious structure in time. If our model is correct, then standardized residuals should look approximately like a Gaussian white noise (purely random) process with zero mean and unit variance (S-PLUS 2000 Guide to Statistics, Vol 2, pp.179, 1999).

Another method for time series model validation is to examine the autocorrelation function of the residuals. If our models are adequate, then the autocorrelations of the residuals should be uncorrelated and approximately Gaussian random variables with mean zero and variance n^{-1}. Therefore, observing large autocorrelations indicates that our models may be inadequate (S-PLUS 2000 Guide to Statistics, Vol 2, pp.179, 1999). Figures 7 through 13 provide an example of ARMA(1,1) with regression variable model diagnostic graphs for each of the CIs for the gear type "AGB Wch Pmp Shft Spur" and Tail Number "164539" (description of the individual parts of the graphs follow).

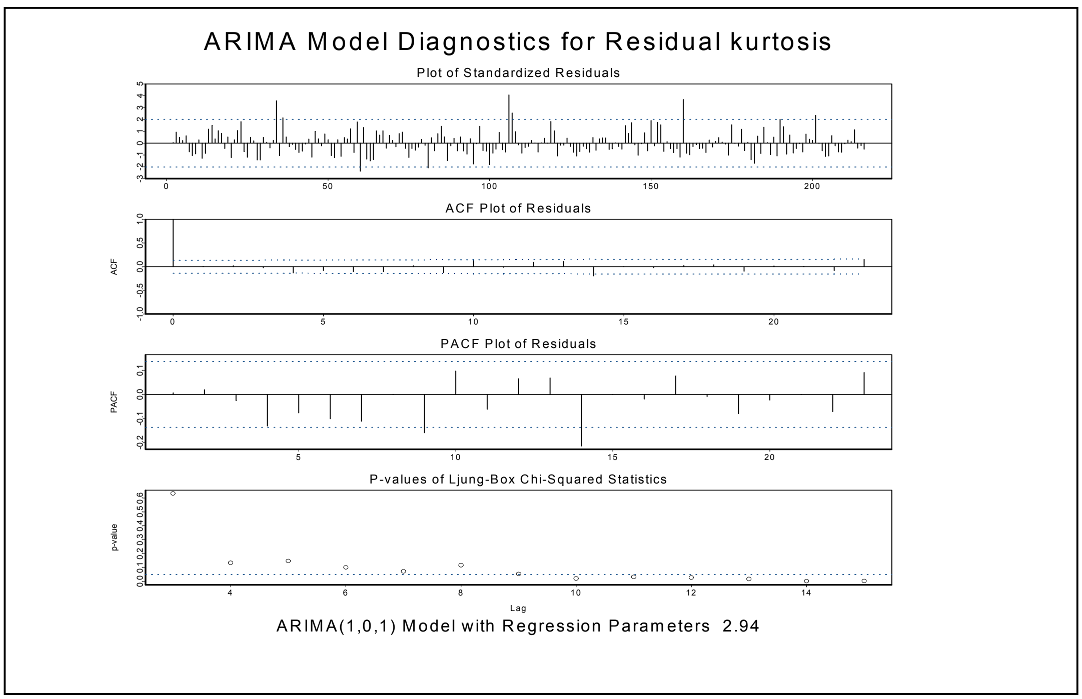

Figure 7. ARMA Model Diagnostics for CI "Residual kurtosis" of Gear Type "AGB Wch Pmp Shft Spur" and Tail Number "164539"

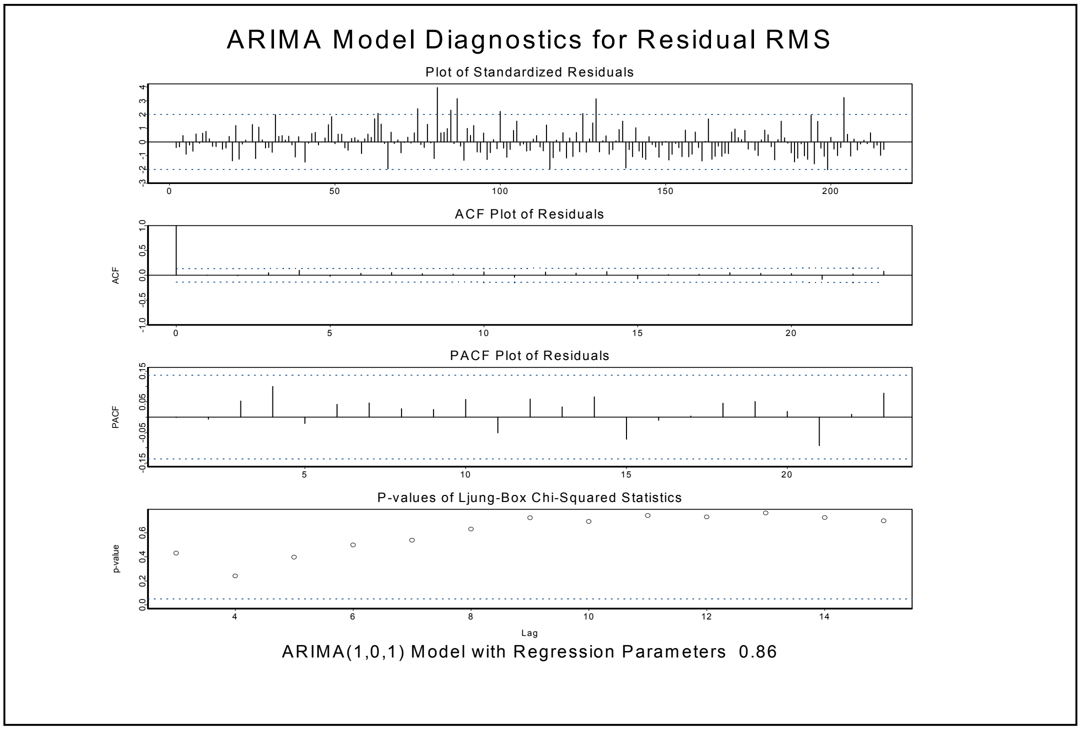

Figure 8. ARMA Model Diagnostics for CI “Residual RMS” of Gear Type “AGB Wch Pmp Shft Spur” and Tail Number “164539”

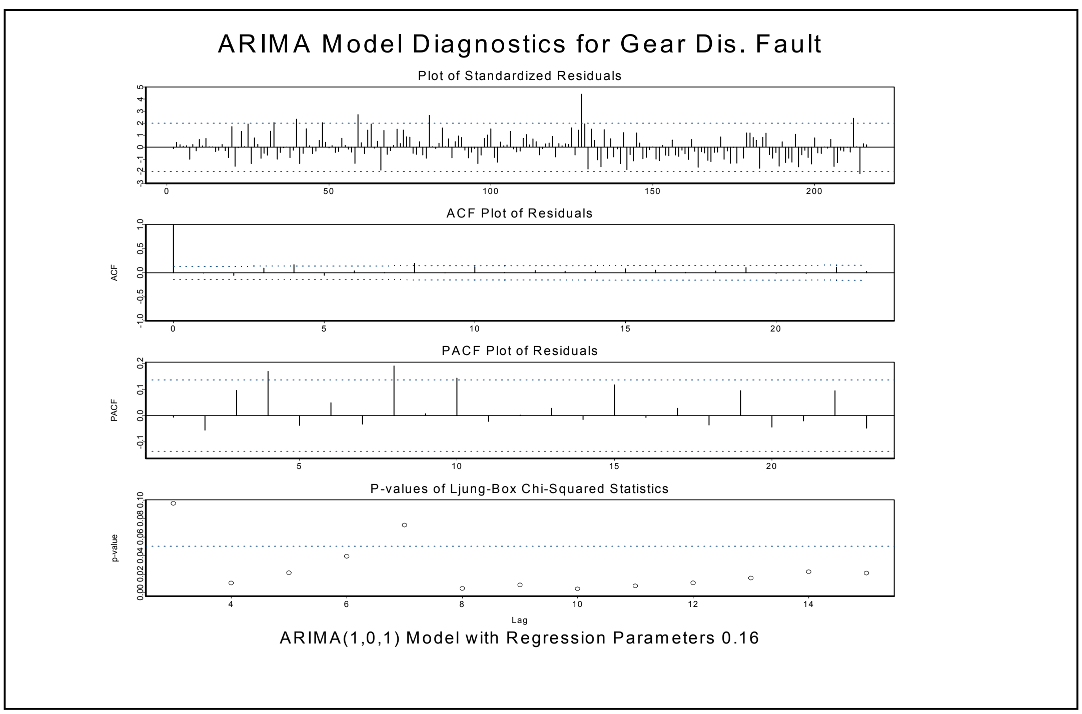

Figure 9. ARMA Model Diagnostics for CI “Gear Dis. Fault” of gear type “AGB Wch Pmp Shft Spur” and Tail Number “164539”

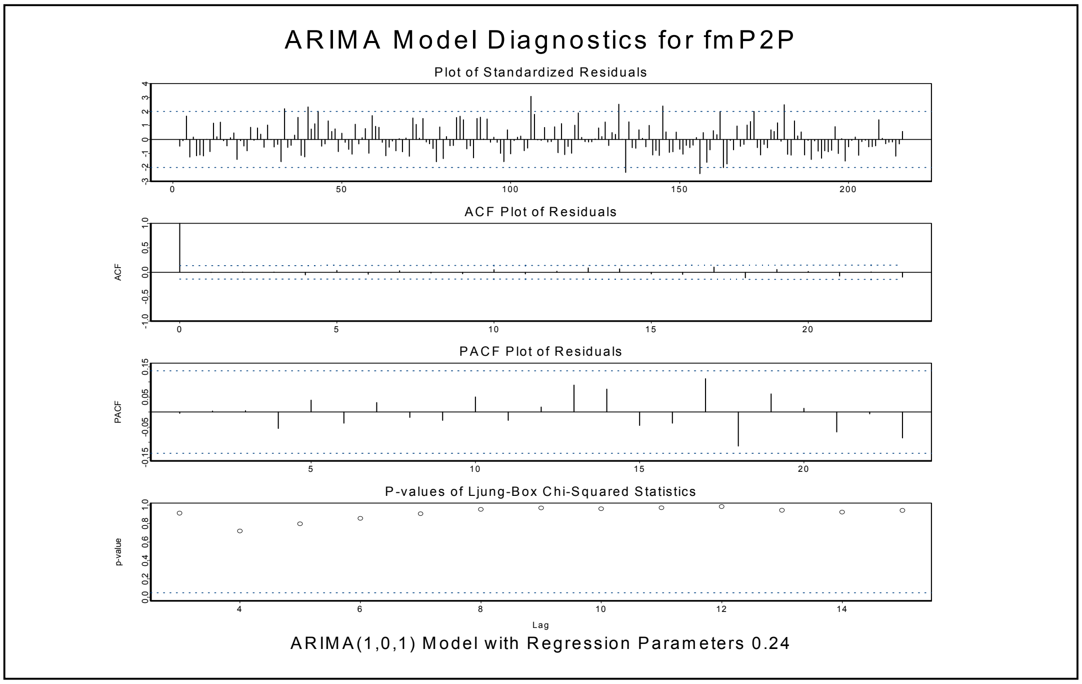

Figure 10. ARMA Model Diagnostics for CI “fmP2P” of Gear Type “AGB Wch Pmp Shft Spur” and Tail Number “164539”

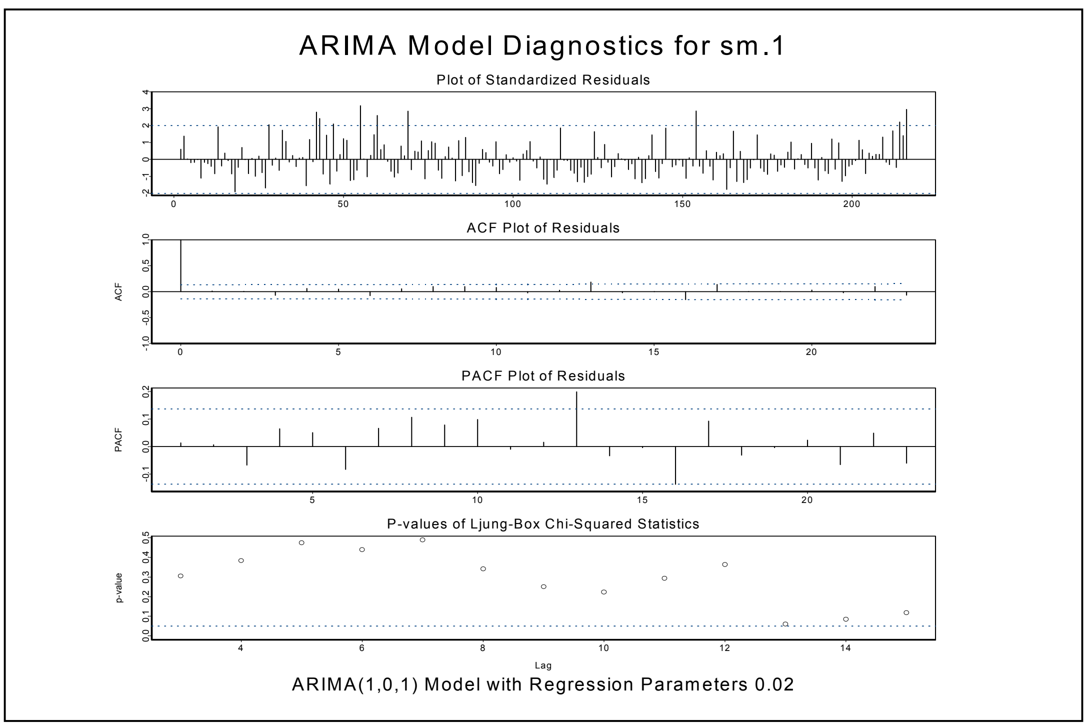

Figure 11. ARMA Model Diagnostics for CI “sm.1” of Gear Type “AGB Wch Pmp Shft Spur” and Tail Number “164539”

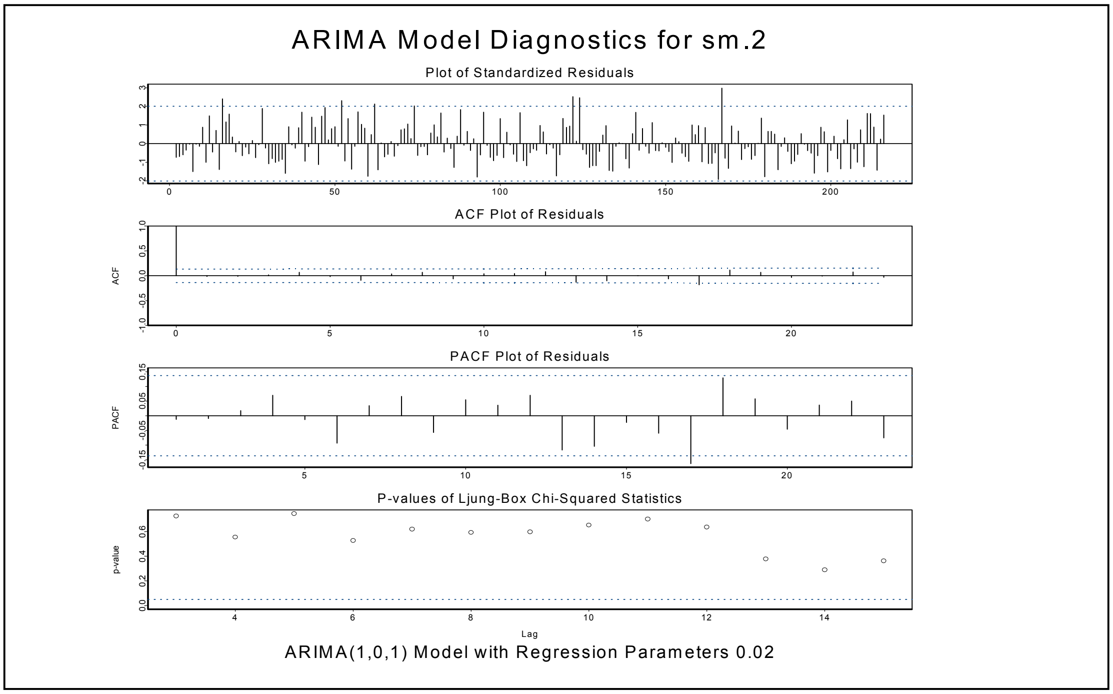

Figure 12. ARMA Model Diagnostics for CI “sm.2” of Gear Type “AGB Wch Pmp Shft Spur” and Tail Number “164539”

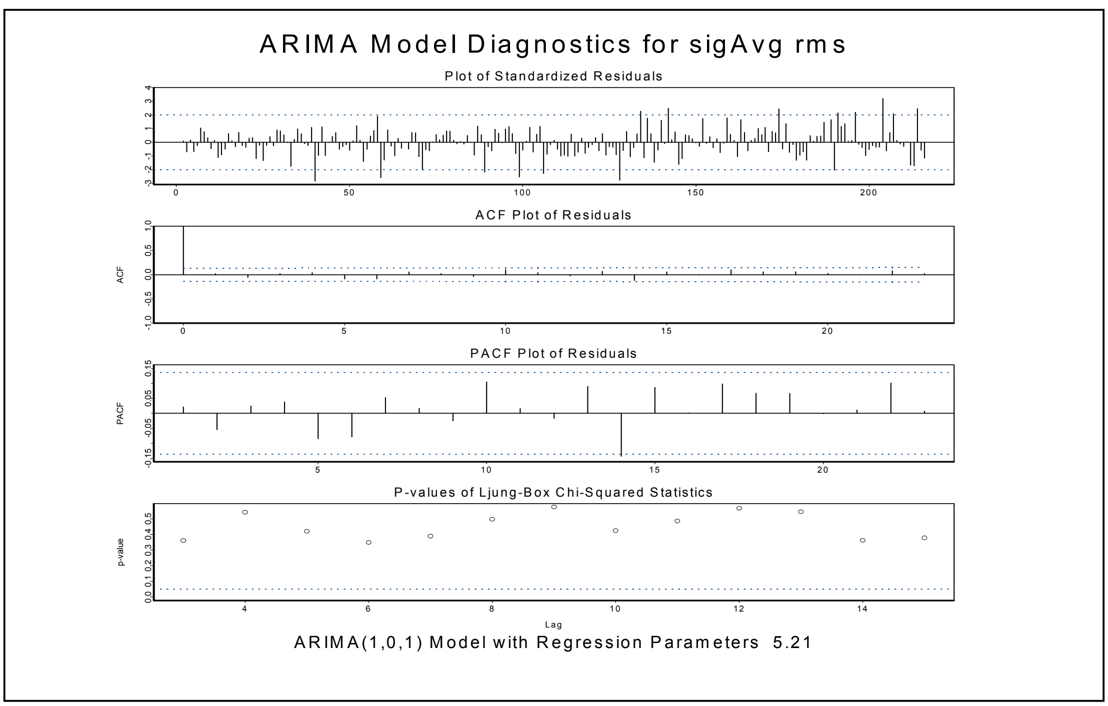

Figure 13. ARMA Model Diagnostics for CI “sigAvg.rms” of Gear Type “AGB Wch Pmp Shft Spur” and Tail Number “164539”

Ljung-Box, a randomness test based on autocorrelation plot, is commonly used to test the quality of fit of a time series model. The model is determined to pass the test if a significant correlation is not observed. However, instead of testing randomness at each distinct lag, Ljung-Box tests the overall randomness based on a number of lags. For this reason, it is often referred to as a “portmanteau” test (Burn Statistics, 2003).

First, we examined standardized residuals graphs for each CI ARMA model. All the standardized residuals behave approximately like a Gaussian white noise process and there is no obvious structure in time. Standardized residuals of each model for CIs are uncorrelated and approximately Gaussian random variables with mean zero and unit variance.

As a second test for model validation, we examined the autocorrelation function of the residuals. It was observed that the autocorrelations of the residuals were uncorrelated and approximately Gaussian random variables with mean zero and variance n^{-1}. For this case our sample size n was equal to 216. Almost no large residual values were observed. Similar results were obtained from the other models for different gear types and tail numbers. Therefore, we concluded that our models were adequate.

However, in 28 out of 252 models, for a particular gear type and tail number, the arima.mle() function did not converge, presumably because of the small sample sizes. A list of these data sets is provided in Table 5.

Gear Index/Name	Tail Number	Sample Size
19 AGB Stg2 Srv Pmp Shft Spur	162464	90
22 Aux Lb Vn Pmp Shft Blades	163075	15
23 Aux Lb Vn Pmp Shft Gear	163075	15
24 Grnd Stg 1 Ring	163075	15
25 Grnd Stg 2 Ring	163075	15
26 IGB Inp Shft Pin	163075,164539	4,9
29 Main Rtr Shft OP Spur	163075	15
32 Outer Shaft Main Bev	163075	15
33 Outer Shaft Sun Gear	163075	15
35 Port Aft Inp Drv Shft Pin	163075	15
45 Smp Rot Pmp Shft Blades	163075	15
46 Smp Rot Pmp Shft Gear	163075	15
47 Stbd Aft Inp Drv Shft Pin	163075	15
53 Stg 1 Hyd Pmp Shft Spur	163086	52
54 Stg 1 Plnt Shft Gear	163075	15
55 Stg 2 Plnt Shft Gear	163075	15
56 Stg 2 Sun Shft Gear	163075	15
58 TGB Inp Shft Gear	163075,163086,164539	4,20,9
59 TGB Inp Shft Pin	163075,163086,164539	4,20,9
60 TGB Oil Pmp Shft Blades	163075,164539	4,9
61 TGB Oil Pmp Shft Gear	163075,164539	4,9

Table 5. The List of Data Sets, which could not be Modeled Due to the Small Sample Sizes.

After modeling each of the seven CIs (See Table 1 in Chapter II) for a particular data set, the new CI matrices were created by taking the standardized residuals of each of those CI models. Then for the rest of the analysis we used these new matrices. To accomplish this, we used the fuction “make.newci” in Appendix A. See Chapter II, Section B.3.e for details.

B. MAHALANOBIS METRIC ANALYSIS

We have a multivariate data set of CIs for each data set of a particular gear type and tail number. As we stated previously, since no failure occurred during the collection of our data, we can assume that all the gears in each aircraft are healthy. Therefore, calculating the Mahalanobis metric, a multivariate distance metric, should give us an insight about the expected range of the Mahalanobis distances for a specific healthy gear type. Then, we can use this information for each data set to set a threshold value in order to detect any unusual level in these CI values relating to pre-existing machining or manufacturing-induced defects, loss of lubrication, corrosive environments and resulting pitting damage or excessive loading.

Using the new CIs of 224 data sets which we managed to model, we calculated the Mahalanobis distances. To accomplish this, the function “make.mahanew” in Appendix A was used. See Chapter II, Section B.5 for Mahalanobis Metric details.

Next, we wanted to determine which distribution would fit best to each of these Mahalanobis distances data sets. By accomplishing this, we would be able to set threshold values of “Warning” and “Alarm” for each particular gear type and tail number. Then this threshold values can be used to detect any defective gears.

Histograms graphically summarize the distribution of a univariate data set and provide strong indications of the proper distributional model of the data. Therefore, we used histograms to have an idea about which population distribution the Mahalanobis data sets might come from. Figure 14 provides some of these histograms for different gear types and tail numbers. These histograms looked very much like those from exponential distributions. But we needed to verify that. To accomplish this, goodness of fit tests were used.

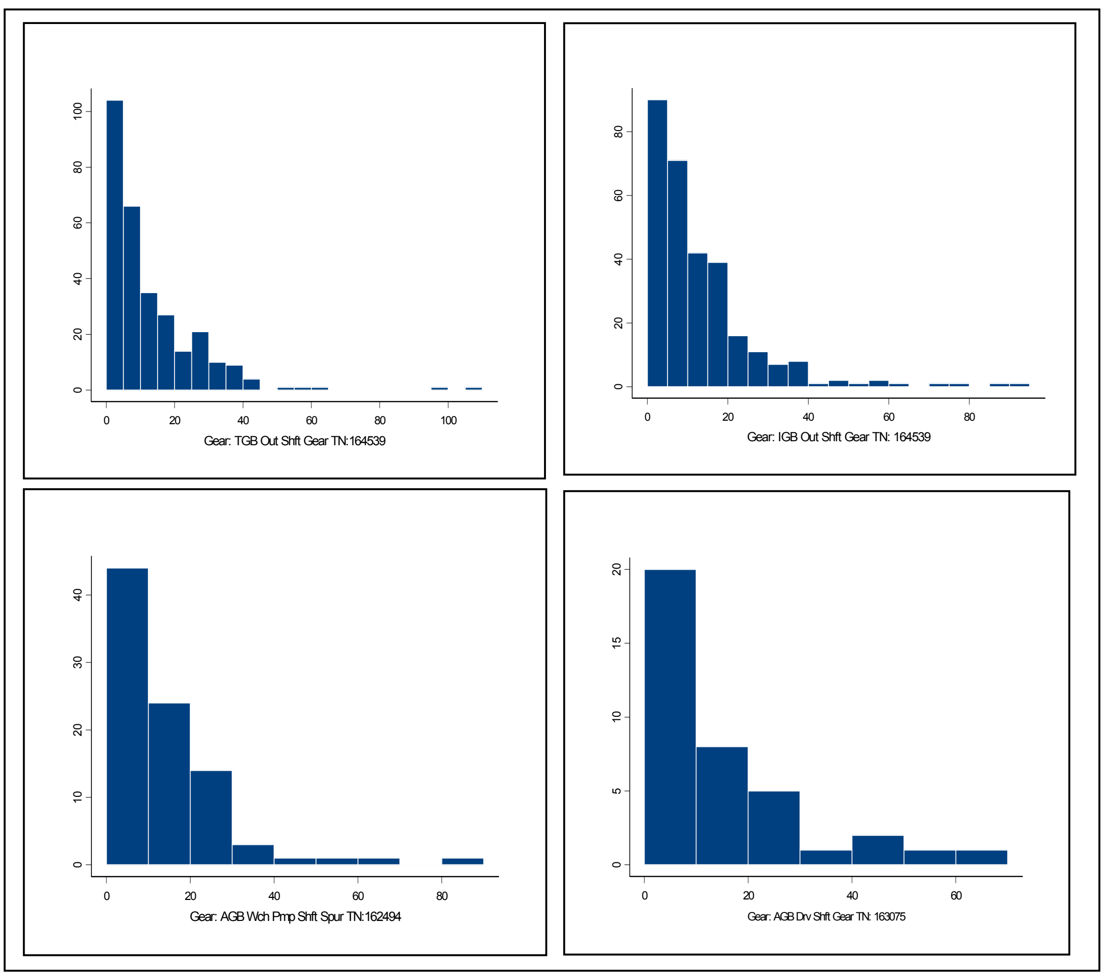

Figure 14. Mahalanobis distances histograms for different gear types and tail numbers

C. GOODNESS OF FIT TESTS ANALYSIS AND BONFERRONI CORRECTION PROCEDURE

Chi-Square and K-S goodness-of-fit tests were applied to decide if a sample set of Mahalanobis distances came from an exponential distribution. See Chapter II, Section B.4 for details about Chi-Square and K-S goodness-of-fit tests.

For each goodness-of-fit test, the α level is conventionally set to 0.05. Since we performed 224 tests on the same hypothesis, the probability of making a Type I error would increase from the conventional α value of 0.05, but we wanted to control Type I error. In order to do this, we applied the Bonferroni multiple comparison correction. Therefore our new Bonferroni adjusted level of significance was calculated as *0.0002193.*

Any test that results in a probability value of less than *0.0002193* was accepted as statistically significant. Similarly, any test statistic with a probability value of greater than *0.0002193* (including values that fall between *0.0002193* and 0.05) was deemed non-significant. Chi Square and K-S goodness-of-fit test results are provided in Appendix B. The Bonferroni adjusted goodness-of-fit test results are summarized in the Table 6.

Test Type	# of Non-significant tests	Percentage
K-S	196	87.5
Chi-Square	188	84

Table 6. The Summary of the Goodness-of-Fit Test results

Then, using the population exponential distributions the threshold values for “Alarm” and “Warning” were set for a particular gear type and tail number. If any Mahalanobis distance occurs greater than 0.999 quantile level, the related gear was considered in “Warning” and if any Mahalanobis distance greater than 0.999999 quantile, the gear was considered in “Alarm”. The calculated threshold values for specific gear types and tail numbers are provided in Appendix C.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. SUMMARY, LIMITATIONS AND RECOMMENDATIONS

The purpose of this study was to establish a threshold level for each particular gear in CH-53 aircraft such that, while minimizing in-flight risk, a negligible false alarm rate is obtained. This would help us decrease the costs, which include the replacement and/or maintenance of any aircraft component, as a result of a false alarm. These thresholds must be set high enough such that a false alarm is a rare event.

The vibration data collected during operational tests was provided by the Goodrich Corporation Fuel & Utility Systems. The dataset included different CIs related to accelerometers and gears for each specific tail number. The entire data consisted of 23187 observations and 20 variables for 63 gear types and four aircraft. Only seven CI columns, related to a particular gear type and tail number, were used to set a threshold value through the analysis.

To calculate a threshold value, first, 252 individual data sets were created from the entire data, each for a particular gear and tail number. Each of the seven CI columns was considered as a univariate time series.

Box-Jenkins ARMA Models were used to model each of these univariate time series. In developing a time series model, the characteristics of each univariate time series data set was analyzed. Time plots were used to accomplish this. Examining each of these time plots, it was observed that almost all of them were plausibly stationary. Since the univariate data sets did not exhibit any significant non-stationarity, the autocorrelations and partial autocorrelation plots were then examined to determine the order of the AR and MA components. Most of them looked like AR(1) and MA(1). Therefore, based on these plots ARMA(1,1) was suggested. Torque was added to our models as a regression variable because it was believed that torque affected the CIs. However, in 28 out of 252 data sets for a particular gear type and tail number the arima.mle() function did not converge, presumably because of small sample sizes.

Having developed the models, standardized residual plots were used to check the diagnostics to determine if the models were reasonable. These plots proved that our model, ARMA(1,1) with torque as a regression variable, was very often an adequate model.

Since the variances of every modeled CI varied considerably from one CI to another, it was rather difficult to know whether a fitted residual should be considered large or small. Therefore, standardized residuals from each CI model for a particular gear type and tail number were saved as a single vector and then these seven CI vectors were used as our new CIs for the rest of the analysis in order to set threshold values of “Warning” and “Alarm.”

As stated previously in Chapter II, our analysis was based on detecting any unusual level in CI values relating to pre-existing machining or manufacturing-induced defects, loss of lubrication, corrosive environments and resulting pitting damage or excessive loading. For this purpose, we used the Mahalanobis distance, which is a multivariate distance metric. Mahalanobis metric provided insight about the expected range of this distance for a specific healthy gear type. Therefore, Mahalanobis distances were calculated using the new CIs for each of the 224 data sets which were modeled successfully.

Next, we needed to find the distribution which would fit each Mahalanobis distance data set. Most of the histogram plots for the Mahalanobis distance data sets for a particular gear type and tail number looked very much as if they came from an exponential distribution. However, we applied Chi-Square and K-S goodness-of-fit to verify that.

We performed 224 individual goodness-of-fit tests. In order to control Type I error, we applied the Bonferroni multiple comparison correction, which allowed 224 comparisons while still assuring an overall alpha value no greater than 0.05. In each case the null hypothesis specified that the CI’s came from the exponential distribution, and in 84% of the data sets using the (Bonferroni-adjusted) chi-square goodness-of-fit test, and in 87.5% using the Kolmogorov-Smirnov, that null hypothesis was not rejected. We set threshold values for “Warning” and “Alarm” for those data sets reported as plausibly exponential using quantiles of the exponential distribution with the parameter estimated from the data. The basic concept for threshold setting was to pick a threshold high enough that the worst aircraft, while still healthy, would not give a false alarm. For this reason, as a rule of thumb, we used 0.999 quantile level for the Warning threshold, and 0.999999 quantile level for the Alarm threshold. But when we checked if there was any warning and alarm situation according to these new threshold values, 200 outliers for “Warning” and 69 outliers for “Alarm” were detected. These outliers would be evaluated as false alarms. Of course, this was not expected since we knew that no failure occurred during the collection of the data used in this analysis. Even when we used the 0.999999999 quantile we still observed 38 outliers. Additionally, sometimes there was a big difference between the threshold levels set for each aircraft for the same gear type.

One of the reasons that this technique may not be sufficient to provide a reasonable warning and alarm rate is that we do not have all the information we might need. Different aircraft might have used different torque level patterns during their flights. Data gathered with torque slowly increasing might be very different than data with torque decreasing, especially in time series modeling. For instance, data collected during a flight pattern with torque level small, medium and then large might be very different than the data with torque level large, medium and then small. We might try to set different threshold values for the same gear type if we had data collected applying different torque levels.

Another reason for setting different and unreasonable threshold values for the same gear type might be that different aircraft had different amount of vibration data for the same gear type.

In future studies attention needs to be paid to patterns of data gathering. It would be valuable to have large data sets, from a number of aircraft, covering some of the torque patterns most often encountered during real operations. We expect that these patterns might be quite different depending on the different missions assigned to the aircraft. Further studies might help determine whether torque history has an effect on CI or whether it is sufficient to consider only the instantaneous value of torque.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. S-PLUS FUNCTIONS

Special thanks to Professor Samuel Buttrey for supplying his knowledge in writing the following S-PLUS functions.

1. make.datanew function

```
function(gears, tails, names.only = F)
{
	#
	# Create a gear data set for a particular gear and tail
	# number. If "names.only" is TRUE, just produce the set of
	# names and return them.
	#
	# Arguments:
	# gears: vector of character string with name of gear
	# tails: vector of character string with tail number

	if(names.only) {
		out <- character(length(gears) * length(tails))
		nm.ctr <- 1
	}
	for(i in 1:length(gears)) {
		datagear <- ac[ac[, "GearName"] == gears[i], ]
		i.txt <- gears[i]
		if(substring(i.txt, 1, 1) == "#")
			i.txt <- substring(i.txt, 2, nchar(i.txt))
		if(substring(i.txt, 9, 9) == "#")
			i.txt    <-    paste(substring(i.txt,    1,8),
			substring(i.txt, 10, nchar( i.txt)),  collapse  =
			"")
		i.txt <- unlist(unpaste(i.txt, sep = " "))
		i.txt <- i.txt[i.txt != ""]
		i.txt <- paste(i.txt, collapse = "")
		for(j in 1:length(tails)) {
			j.txt <-tails[j]
			datageartail <- datagear[datagear
					[,"Tail"]== j.txt,]
			#
			# Construct the name of the thing to be
			# saved
			#
			nm <- paste("ac.", i.txt, ".", j.txt,
				sep = "")
```

```
            if(names.only) {
                out[nm.ctr] <- nm
                nm.ctr <- nm.ctr + 1
            }
            else {
                if(!exists(nm, where = 1)) {
                    cat("Creating object",nm, "\n")
                    assign(nm,datageartail,where = 1)
                }
                else    cat(nm,    "already    exists;    not
            created\n")
            }
        }
    }
    if(names.only)
    return(out)
    return(invisible())
    return(datageartail)
}
```

2. timeplot function

```
function(gears, tails)
{
# Arguments:
# gears: vector of character string with name of gear
# tails: vector of character string with tail number

    for(i in 1:length(gears)) {
        countgears = i
        i.txt <- gears[i]
        if(substring(i.txt, 1, 1) == "#")
            i.txt <- substring(i.txt, 2, nchar(i.txt))
        if(substring(i.txt, 9, 9) == "#")
            i.txt <- paste(substring(i.txt, 1,
                8), substring(i.txt, 10, nchar(
                    i.txt)), collapse = "")
        i.txt <- unlist(unpaste(i.txt, sep = " "))
        i.txt <- i.txt[i.txt != ""]
        i.txt <- paste(i.txt, collapse = "")
        for(j in 1:length(tails)) {
            counttails = j
            j.txt <-   tails[j]
            nm <- paste("ac.", i.txt,".",j.txt,sep = "")
            if(!exists(nm))
                stop(paste("No data set named",nm))
            #
            #it exists go and get it
            #
            data <- get(nm)
```

```
				plotname = paste("Gear:",i.txt,"TN:", j.txt)
				par(mfrow = c(3, 3))
				gearCIs <- names(ac[14:20])
				k = 0
				for(i in 14:20) {
					k = k + 1
					plot(data[, i], type = "l",
						xlab="time",ylab =gearCIs[k])
					if(i == 14) {
						title(main = plotname)
					}
				}
				if(counttails != length(tails)) {
					graphsheet()
				}
			}
			if(countgears != length(gears)) {
				graphsheet()
			}
		}
}
```

3. draw.acf.plots function

```
function(gears, tails)

# Arguments:
# gears: vector of character string with name of gear
# tails: vector of character string with tail number

{
		for(i in 1:length(gears))
		{
			countgears = i
			i.txt <- gears[i]
			if(substring(i.txt, 1, 1) == "#")
				i.txt <- substring(i.txt, 2, nchar(i.txt))
			if(substring(i.txt, 9, 9) == "#")
				i.txt<-paste(substring(i.txt,1,8),
				substring(i.txt, 10, nchar(i.txt)),collapse = "")
			i.txt <- unlist(unpaste(i.txt, sep = " "))
			i.txt <- i.txt[i.txt != ""]
			i.txt <- paste(i.txt, collapse = "")
			for(j in 1:length(tails)) {
				counttails = j
				j.txt <- tails[j]
				nm <- paste("ac.", i.txt, ".", j.txt, sep = "")
				if(!exists(nm))
					stop(paste("No data set named", nm))
				#
```

```
                #it exists go and get it
                #
                data <- get(nm)
                data <- data[14:20]
                data <- as.matrix(data)
                par(mfrow = c(3, 3))
                k = 0
                for(i in 1:7) {
                      k = k + 1
                      acf(data[, i])
                }
                par(mfrow = c(3, 3))
                k = 0
                for(i in 1:7) {
                      k = k + 1
                      acf(data[, i], type = "p")
                }
                if(counttails != length(tails)) {
                      graphsheet()
                }
          }
          if(countgears != length(gears)) {
                graphsheet()
          }
    }
}
```

4. make.newci function

```
function(gears, tails)
{
      #
      # Create a gear data set for a particular gear and
      # tail number.
      #
      # Arguments:
      # gears: vector of character string with name of gear
      # tails: vector of character string with tail number

      for(i in 1:length(gears)) {
            i.txt <- gears[i]
            if(substring(i.txt, 1, 1) == "#")
                  i.txt <- substring(i.txt, 2, nchar(i.txt))
            if(substring(i.txt, 9, 9) == "#")
                  i.txt<-paste(substring(i.txt,1,8),
                  substring(i.txt,10,nchar(i.txt)),collapse= "")
            i.txt <- unlist(unpaste(i.txt, sep = " "))
            i.txt <- i.txt[i.txt != ""]
            i.txt <- paste(i.txt, collapse = "")
            for(j in 1:length(tails)) {
                  j.txt <- tails[j]
```

```
#
#Construct the name of the thing to be saved
#
nm<-paste("ac.",i.txt, ".", j.txt, sep = "")
if(!exists(nm))
      stop(paste("No data set named", nm))
#
#it exists go and get it
#
data <- get(nm)
for(k in 1:7) {
      if(k == 1) {
            zap.Res.kur<-
            arima.mle(data$Residual.kurtosis,
            model = list(ar = 0.8, ma = 0.2), xreg
            = cbind(1, data$Torque))
            zap.Res.kur.ARIMA.res<-
            arima.diag(zap.Res.kur,plot=F)$std.res
            id
            first <- zap.Res.kur.ARIMA.res
      }
      if(k == 2) {
            zap.Res.rms<-
            arima.mle(data$Residual.rms,model=
            list(ar  =  0.8,  ma  =  0.2),  xreg  =
            cbind(1, data$Torque))
            zap.Res.rms.ARIMA.res<-
            arima.diag(zap.Res.rms,plot=F)$std.res
            id
            second <- zap.Res.rms.ARIMA.res
      }
      if(k == 3) {
            zap.Geardisfault<-
            arima.mle(data$GearDisFault,model=
            list(ar  =  0.8,  ma  =  0.2),  xreg  =
            cbind(1, data$Torque))
            zap.Geardisfault.ARIMA.res<-
            arima.diag(zap.Geardisfault,  plot  =
            F)$std.resid
            third<- zap.Geardisfault.ARIMA.res
      }
      if(k == 4) {
            zap.fmP2P <-
            arima.mle(data$fmP2P,model= list(ar  =
            0.8,  ma  =  0.2),  xreg  =  cbind(1,
            data$Torque))
            zap.fmP2P.ARIMA.res<-
            arima.diag(zap.fmP2P,plot=F)$std.
            resid
            fourth <- zap.fmP2P.ARIMA.res
      }
```

```
                    if(k == 5) {
                        zap.sm.1<-
                        arima.mle(data$sm.1,model=
                        list(ar  =  0.8,  ma  =  0.2),  xreg  =
                        cbind(1, data$Torque))
                        zap.sm.1.ARIMA.res<-
                        arima.diag(zap.sm.1,plot=F)$std.
                        resid
                        fifth <- zap.sm.1.ARIMA.res
                    }
                    if(k == 6) {
                        zap.sm.2<-
                        arima.mle(data$sm.2,model=
                        list(ar  =  0.8,  ma  =  0.2),  xreg  =
                        cbind(1, data$Torque))
                        zap.sm.2.ARIMA.res<-
                        arima.diag(zap.sm.2,plot=F)$std.
                        resid
                        sixth <- zap.sm.2.ARIMA.res
                    }
                    if(k == 7) {
                        zap.sigAvg.rms<-
                        arima.mle(data$sigAvg.rms,model=
                        list(ar  =  0.8,  ma  =  0.2),  xreg  =
                        cbind(1, data$Torque))
                        zap.sigAvg.rms.ARIMA.res<-
                        arima.diag(zap.sigAvg.rms,plot=F)$std.
                        resid
                        seventh<- zap.sigAvg.rms.ARIMA.res

                        NewCIName<-   paste("NewCI.",   i.txt,
                        ".", j.txt, sep = "")
                        NewCI   <-   matrix(c(first,   second,
                        third, fourth, fifth, sixth, seventh),
                        ncol = 7)
                        assign(NewCIName,NewCI,where = 1)
                    }
                }
            }
        }
        return(invisible())
        return(NewCI)
}
```

5. make.mahanew function

```
function(gears, tails, name.only = F)
{
      #
      # make.maha: compute Mahalanobis distance for
      # particular gear and tail number
      #
      # Arguments:
      # gears: vector of character string with name of gear
      # tails: vector of character string with tail number
      # name.only: if TRUE, just return condensed version of #
      name
      #
      # Construct name of data set, then go get it
      #
      if(missing(gears)||missing(tails))
            stop("Both arguments must be supplied!")
      if(name.only&&(length(gears)>1||length(tails) > 1))
            stop("Not set up for vectorized names!")
      for(i in 1:length(gears)) {
            i.txt <- gears[i]
            if(substring(i.txt, 1, 1) == "#")
                  i.txt <- substring(i.txt, 2, nchar(i.txt))
            if(substring(i.txt, 9, 9) == "#")
                  i.txt <- paste(substring(i.txt, 1, 8),
                  substring(i.txt, 10, nchar(i.txt)), collapse =
                  "")
            i.txt <- unlist(unpaste(i.txt, sep = " "))
            i.txt <- i.txt[i.txt != ""]
            i.txt <- paste(i.txt, collapse = "")
            for(j in 1:length(tails)) {
                  j.txt <- tails[j]
                  #
                  # Construct the names of the things to be
                  # saved ("maha") and the data ("NewCI")
                  #
                  maha.nm<-paste("Maha.",i.txt,".",j.txt,sep="")
                  if(name.only)
                        return(maha.nm)
                  data.nm<-paste("NewCI.",i.txt,".",j.txt,
                  sep="")
                  if(!exists(data.nm))
                        stop(paste("No data set named", data.nm))
                  #
                  # It exists. Go get it.
                  #
                  data <- get(data.nm)
                  #
                  # If it's character data, fix it
                  #
```

```
                if(is.character(data))
                      data<-matrix(as.numeric(data),ncol=
                      ncol(data))
                #
                # Compute column-wise means, assemble into a #
                px1 row matrix
                #
                m <- apply(data, 2, mean, na.rm = T)
                m <- matrix(m, nrow = 1)
                #
                # Compute (x - mean) by replicating mean as #
                necessary
                #
                th <- data - m[rep(1, nrow(data)),  ]
                # (x - mean(x))
                # Compute covariance matrix, get Maha
                # distance
                #
                vmat <- var(data, na.method = "omit")
                maha <- diag(th %*% vmat %*% t(th))
                assign(maha.nm, maha, where = 1)
           }
     }
     return(invisible())
     return(maha)
}
```

6. make.maha.analysis function

```
function(maha, delete.extremes = 0.999999)
{
     if(missing(maha))
          stop("Mahalanobis argument must be supplied!")
     #
     # Strip off that leading NA
     if(is.na(maha[1]))
          maha <- maha[-1]
     #
     # If "delete.extremes" is TRUE, cut off any distances
     # more extreme than the "delete.extreme" th
     # percentage point of the exponential. By default it's
     # a percentage point; turn this off by passing "FALSE."
     #
     if(is.logical(delete.extremes)&&delete.extremes==TRUE)
     delete.extremes <- 0.999999
     if(is.numeric(delete.extremes)) {
          gof.save<-chisq.gof(maha,distribution = "exponential",
          rate = 1/(mean(maha)), n.param.est = 1)
          cutoff <- qexp(delete.extremes, rate = 1/mean(maha))
          num.cutoff <- sum(maha > cutoff)
          if(num.cutoff > 0) {
```

```
            warning(paste("Cut  off",  num.cutoff,  "outliers  in",
            substitute(deparse(maha))),  ";  old  p-value  was  ",
            signif(gof.save$ p.value, 4), "\n")
            maha <- maha[maha <= cutoff]
            }
        }
        final.chisq<-chisq.gof(maha,  distribution  =  "exponential",
        rate = 1/(mean(maha)),n.param.est = 1)
        final.ks <- ks.gof(maha, distribution = "exponential", rate
        = 1/(mean(maha)))
        print(final.chisq)
        print(final.ks)
        return(c(final.chisq$p.value,final.ks$p.value, num.cutoff))
}
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B GOF TEST RESULTS FOR EXPONENTIAL DISTRIBUTION USING BONFERRONI CORRECTION

The dark colored cells mean that the related p-values are greater than the Bonferroni adjusted significance level of *0.0002193.*

Index	Mahalanobis Distance Name	No	Chi-Square gof	KS gof
1	Maha.2EngFCDrvShftSpur.162494	83	**0.3748739000**	**0.0791666800**
1	Maha.2EngFCDrvShftSpur.163075	34	**0.3657477000**	**0.5025112000**
1	Maha.2EngFCDrvShftSpur.163086	52	**0.0012292210**	**0.0439324800**
1	Maha.2EngFCDrvShftSpur.164539	185	0.0000498706	0.0000038110
2	Maha.2EngFrWhShftCamGear.162494	83	**0.0369843500**	**0.6079170000**
2	Maha.2EngFrWhShftCamGear.163075	34	**0.0041954820**	**0.0441714200**
2	Maha.2EngFrWhShftCamGear.163086	52	**0.0340014000**	**0.0438103100**
2	Maha.2EngFrWhShftCamGear.164539	185	0.0000945735	**0.0023453220**
3	Maha.2EngFrWhShftDrvSpur.162494	83	**0.0033658590**	**0.0170915500**
3	Maha.2EngFrWhShftDrvSpur.163075	34	**0.0699241400**	**0.0161819700**
3	Maha.2EngFrWhShftDrvSpur.163086	52	**0.3594478000**	**0.4238913000**
3	Maha.2EngFrWhShftDrvSpur.164539	185	0.0000000350	0.0000002973
4	Maha.2EngFrWhShftSpur.162494	83	**0.1479257000**	**0.0016803680**
4	Maha.2EngFrWhShftSpur.163075	34	**0.3168349000**	**0.6142077000**
4	Maha.2EngFrWhShftSpur.163086	52	**0.0148596500**	**0.0306954200**
4	Maha.2EngFrWhShftSpur.164539	185	0.0001565313	0.0000727556
5	Maha.2EngInpShftSpur.162494	83	**0.1365716000**	**0.3819240000**
5	Maha.2EngInpShftSpur.163075	34	**0.0430359500**	**0.0014587820**
5	Maha.2EngInpShftSpur.163086	52	**0.5114722000**	**0.0958062800**
5	Maha.2EngInpShftSpur.164539	185	0.0000000053	0.0000003074
6	Maha.2EngTachShftSpur.162494	83	**0.0636493900**	**0.1905592000**
6	Maha.2EngTachShftSpur.163075	34	**0.0580401100**	**0.0515810500**
6	Maha.2EngTachShftSpur.163086	52	**0.1699629000**	**0.6952633000**
6	Maha.2EngTachShftSpur.164539	185	0.0000218801	**0.0007740179**
7	Maha.2GenShftSpur.162494	83	**0.0637712400**	**0.3091500000**
7	Maha.2GenShftSpur.163075	34	**0.0480404400**	**0.1267582000**
7	Maha.2GenShftSpur.163086	52	**0.0954713500**	**0.0223861700**
7	Maha.2GenShftSpur.164539	185	**0.0015239180**	**0.0040696120**
8	Maha.2InpShftAftIdler.162494	83	**0.0829717000**	**0.3815830000**
8	Maha.2InpShftAftIdler.163075	34	**0.4776873000**	**0.5649648000**
8	Maha.2InpShftAftIdler.163086	52	**0.0005505253**	**0.2728207000**
8	Maha.2InpShftAftIdler.164539	185	**0.0007459824**	**0.0095353930**

Index	Mahalanobis Distance Name	No	Chi-Square gof	KS gof
9	Maha.2InpShftIdler.162494	83	**0.1729916000**	**0.1782416000**
9	Maha.2InpShftIdler.163075	34	**0.1688859000**	**0.2399904000**
9	Maha.2InpShftIdler.163086	52	**0.4315571000**	**0.3862667000**
9	Maha.2InpShftIdler.164539	185	0.0000512785	**0.0396911700**
10	Maha.2InpShftPin.162494	83	**0.0019748370**	**0.0022502640**
10	Maha.2InpShftPin.163075	34	**0.1688859000**	**0.0535416900**
10	Maha.2InpShftPin.163086	52	**0.5536479000**	**0.2674302000**
10	Maha.2InpShftPin.164539	185	**0.0106145300**	**0.0187534900**
11	Maha.AGBActShftIdler.162494	90	**0.1109026000**	**0.1122277000**
11	Maha.AGBActShftIdler.163075	39	**0.0298144600**	**0.2070951000**
11	Maha.AGBActShftIdler.163086	38	**0.2916541000**	**0.3702179000**
11	Maha.AGBActShftIdler.164539	216	**0.0056000150**	**0.1704107000**
12	Maha.AGBActShftSpur.162494	90	**0.0400923300**	**0.2174343000**
12	Maha.AGBActShftSpur.163075	39	**0.2969329000**	**0.3026031000**
12	Maha.AGBActShftSpur.163086	38	**0.0385024600**	**0.0033220030**
12	Maha.AGBActShftSpur.164539	216	**0.0009973003**	**0.0108290100**
13	Maha.AGBDrvShftGear.162494	90	**0.1105259000**	**0.1543901000**
13	Maha.AGBDrvShftGear.163075	39	**0.8648678000**	**0.9787855000**
13	Maha.AGBDrvShftGear.163086	38	**0.2213292000**	**0.3685031000**
13	Maha.AGBDrvShftGear.164539	216	**0.0026133800**	**0.1420335000**
14	Maha.AGBDrvShftSpur.162494	90	**0.0366537800**	**0.5991601000**
14	Maha.AGBDrvShftSpur.163075	39	**0.4798044000**	**0.9160088000**
14	Maha.AGBDrvShftSpur.163086	38	**0.1218870000**	**0.1098200000**
14	Maha.AGBDrvShftSpur.164539	216	**0.0031395750**	**0.0425268100**
15	Maha.AGBEngStrtShftSpur.162494	90	0.0000286636	**0.0041275020**
15	Maha.AGBEngStrtShftSpur.163075	39	**0.2272758000**	**0.7600296000**
15	Maha.AGBEngStrtShftSpur.163086	38	**0.3774363000**	**0.3392619000**
15	Maha.AGBEngStrtShftSpur.164539	216	**0.0245094900**	**0.0650491900**
16	Maha.AGBGen1ShftSpur.162494	90	**0.0039748860**	**0.1567953000**
16	Maha.AGBGen1ShftSpur.163075	39	**0.3812781000**	**0.6079638000**
16	Maha.AGBGen1ShftSpur.163086	38	**0.0009150034**	**0.0127535200**
16	Maha.AGBGen1ShftSpur.164539	216	**0.0025338340**	**0.0306988000**
17	Maha.AGBGen3ShftSpur.162494	90	**0.0006621484**	**0.0319469700**
17	Maha.AGBGen3ShftSpur.163075	39	**0.2603022000**	**0.4511475000**
17	Maha.AGBGen3ShftSpur.163086	38	**0.2546023000**	**0.0899478600**
17	Maha.AGBGen3ShftSpur.164539	216	0.0001620264	**0.0218363500**
18	Maha.AGBOPShftSpur.162494	90	**0.0192180900**	**0.0281025400**
18	Maha.AGBOPShftSpur.163075	39	**0.4288799000**	**0.6370583000**
18	Maha.AGBOPShftSpur.163086	38	**0.2213292000**	**0.6146984000**
18	Maha.AGBOPShftSpur.164539	216	**0.0245094900**	**0.0747140200**
19	Maha.AGBStg2SrvPmpShftSpur.163075	39	**0.1480693000**	**0.2531048000**
19	Maha.AGBStg2SrvPmpShftSpur.163086	38	**0.2213292000**	**0.4064030000**
19	Maha.AGBStg2SrvPmpShftSpur.164539	216	0.0000621991	**0.0136492000**

Index	Mahalanobis Distance Name	No	Chi-Square gof	KS gof
20	Maha.AGBUtPmpShftSpur.162494	90	**0.2342394000**	**0.4256110000**
20	Maha.AGBUtPmpShftSpur.163075	39	**0.2969329000**	**0.5324835000**
20	Maha.AGBUtPmpShftSpur.163086	38	**0.1041706000**	**0.3052580000**
20	Maha.AGBUtPmpShftSpur.164539	216	**0.0130552000**	**0.1373432000**
21	Maha.AGBWchPmpShftSpur.162494	90	**0.0800294500**	**0.6985457000**
21	Maha.AGBWchPmpShftSpur.163075	39	**0.2969329000**	**0.3709352000**
21	Maha.AGBWchPmpShftSpur.163086	38	**0.0025404140**	**0.2168790000**
21	Maha.AGBWchPmpShftSpur.164539	216	0.0000010566	0.0000018322
22	Maha.AuxLbVnPmpShftBlades.162494	64	**0.0261544000**	**0.0529617500**
22	Maha.AuxLbVnPmpShftBlades.163086	23	**0.3062189000**	**0.3174911000**
22	Maha.AuxLbVnPmpShftBlades.164539	68	**0.0013595550**	**0.0071380010**
23	Maha.AuxLbVnPmpShftGear.162494	64	**0.0520471600**	**0.2971274000**
23	Maha.AuxLbVnPmpShftGear.163086	23	**0.4500622000**	**0.3625316000**
23	Maha.AuxLbVnPmpShftGear.164539	68	**0.0052367110**	**0.0486329300**
24	Maha.GrndStg1Ring.162494	64	**0.3959118000**	**0.2298150000**
24	Maha.GrndStg1Ring.163086	23	**0.9077655000**	**0.8541475000**
24	Maha.GrndStg1Ring.164539	68	**0.0179124000**	**0.3808594000**
25	Maha.GrndStg2Ring.162494	64	**0.2856284000**	**0.7498076000**
25	Maha.GrndStg2Ring.163086	23	**0.1286267000**	**0.5020362000**
25	Maha.GrndStg2Ring.164539	68	**0.0821774600**	**0.0777073500**
26	Maha.IGBInpShftPin.162494	19	**0.1005221000**	**0.2943942000**
26	Maha.IGBInpShftPin.163086	20	**0.1246928000**	**0.4000300000**
27	Maha.IGBOutShftGear.162494	173	0.0000460976	0.0002195990
27	Maha.IGBOutShftGear.163075	80	**0.1425494000**	**0.0588983300**
27	Maha.IGBOutShftGear.163086	147	0.0000000000	0.0000002919
27	Maha.IGBOutShftGear.164539	296	**0.0746511600**	**0.2949536000**
28	Maha.IGBOutShftPumpBlades.162494	173	0.0000529905	0.0001117883
28	Maha.IGBOutShftPumpBlades.163075	80	**0.1425494000**	**0.3137684000**
28	Maha.IGBOutShftPumpBlades.163086	147	0.0001247872	0.0000183008
28	Maha.IGBOutShftPumpBlades.164539	296	**0.0016775260**	**0.0146860100**
29	Maha.MainRtrShftOPSpur.162494	64	0.0001068427	**0.0040404190**
29	Maha.MainRtrShftOPSpur.163086	23	**0.3062189000**	**0.9158680000**
29	Maha.MainRtrShftOPSpur.164539	68	**0.0200847200**	**0.0481922700**
30	Maha.MainRtrTachShftSpur.162494	83	**0.1166787000**	**0.0409124800**
30	Maha.MainRtrTachShftSpur.163075	34	**0.0326580900**	**0.7861486000**
30	Maha.MainRtrTachShftSpur.163086	52	**0.6845130000**	**0.7474594000**
30	Maha.MainRtrTachShftSpur.164539	185	**0.1072321000**	**0.0692328500**
31	Maha.OilCoolShftSpur.162494	83	**0.0104796900**	**0.0701250500**
31	Maha.OilCoolShftSpur.163075	34	**0.3168349000**	**0.2736407000**
31	Maha.OilCoolShftSpur.163086	52	**0.1724879000**	**0.0271953700**
31	Maha.OilCoolShftSpur.164539	185	**0.0811854500**	**0.0747964600**
32	Maha.OuterShaftMainBev.162494	64	**0.3109492000**	**0.4856828000**
32	Maha.OuterShaftMainBev.163086	23	**0.0233787700**	**0.1957528000**

Index	Mahalanobis Distance Name	No	Chi-Square gof	KS gof
32	Maha.OuterShaftMainBev.164539	68	**0.1480942000**	**0.3900619000**
33	Maha.OuterShaftSunGear.162494	64	**0.5991661000**	**0.3332178000**
33	Maha.OuterShaftSunGear.163086	23	**0.4500622000**	**0.2655298000**
33	Maha.OuterShaftSunGear.164539	68	**0.0150263200**	**0.0170488000**
34	Maha.PortAftInpDrvShftACCPi.162494	147	**0.0219083200**	**0.0141299400**
34	Maha.PortAftInpDrvShftACCPi.163075	78	**0.0709423700**	**0.3166578000**
34	Maha.PortAftInpDrvShftACCPi.163086	113	**0.0019700140**	**0.2014757000**
34	Maha.PortAftInpDrvShftACCPi.164539	271	0.0000000387	0.0001546579
35	Maha.PortAftInpDrvShftPin.162494	64	**0.0995602000**	**0.0371758200**
35	Maha.PortAftInpDrvShftPin.163086	23	**0.4500622000**	**0.4384801000**
35	Maha.PortAftInpDrvShftPin.164539	68	**0.4666952000**	**0.2765908000**
36	Maha.PortNGBEngInpShftPin.162494	147	**0.2535967000**	**0.0164562100**
36	Maha.PortNGBEngInpShftPin.163075	78	**0.0188582600**	**0.0114002600**
36	Maha.PortNGBEngInpShftPin.163086	113	**0.0348115800**	**0.0067354960**
36	Maha.PortNGBEngInpShftPin.164539	271	**0.1258339000**	**0.0063320690**
37	Maha.PortNGBFCDrvShftGear.162494	90	**0.0035879460**	**0.0186399300**
37	Maha.PortNGBFCDrvShftGear.163075	39	**0.3325939000**	**0.3163830000**
37	Maha.PortNGBFCDrvShftGear.163086	38	**0.4782304000**	**0.9824403000**
37	Maha.PortNGBFCDrvShftGear.164539	216	**0.0062491490**	**0.0002623695**
38	Maha.PortNGBFCDrvShftLHZerl.162494	90	**0.0943744600**	**0.1783191000**
38	Maha.PortNGBFCDrvShftLHZerl.163075	39	**0.5334521000**	**0.5918447000**
38	Maha.PortNGBFCDrvShftLHZerl.163086	38	**0.2546023000**	**0.7759360000**
38	Maha.PortNGBFCDrvShftLHZerl.164539	216	0.0000611931	0.0000458461
39	Maha.PortNGBFCDvnShftLHZerl.162494	90	**0.0005328204**	0.0000846702
39	Maha.PortNGBFCDvnShftLHZerl.163075	39	**0.8221806000**	**0.8943008000**
39	Maha.PortNGBFCDvnShftLHZerl.163086	38	**0.7092986000**	**0.5676922000**
39	Maha.PortNGBFCDvnShftLHZerl.164539	216	**0.0386258400**	**0.1650291000**
40	Maha.PortNGBOPDrvShftSpur.162494	90	**0.1109026000**	**0.5885142000**
40	Maha.PortNGBOPDrvShftSpur.163075	39	**0.4260713000**	**0.9965774000**
40	Maha.PortNGBOPDrvShftSpur.163086	38	**0.0887824000**	**0.0401910400**
40	Maha.PortNGBOPDrvShftSpur.164539	216	0.0000000000	0.0000002919
41	Maha.PortNGBOutShftACCSpur.162494	147	**0.0248275200**	**0.2222217000**
41	Maha.PortNGBOutShftACCSpur.163075	78	0.0000120340	**0.0056900410**
41	Maha.PortNGBOutShftACCSpur.163086	113	**0.0011579580**	**0.0005197010**
41	Maha.PortNGBOutShftACCSpur.164539	271	**0.0724376400**	**0.3204947000**
42	Maha.PortNGBOutShftGear.162494	147	**0.0264978700**	**0.2391644000**
42	Maha.PortNGBOutShftGear.163075	78	**0.0089832700**	**0.0005709405**
42	Maha.PortNGBOutShftGear.163086	113	0.0000204813	0.0000048697
42	Maha.PortNGBOutShftGear.164539	271	**0.0731116000**	**0.0007343518**
43	Maha.PortNGBTachShftSpur.162494	90	**0.0366537800**	**0.1025068000**
43	Maha.PortNGBTachShftSpur.163075	39	**0.8722260000**	**0.6554800000**
43	Maha.PortNGBTachShftSpur.163086	38	**0.1218870000**	**0.0103405600**
43	Maha.PortNGBTachShftSpur.164539	216	0.0000000000	0.0000002919

Index	Mahalanobis Distance Name	No	Chi-Square gof	KS gof
44	Maha.RrCovIdlerShftIdler.162494	83	0.0000337472	0.0000080497
44	Maha.RrCovIdlerShftIdler.163075	34	**0.4194763000**	**0.2027049000**
44	Maha.RrCovIdlerShftIdler.163086	52	**0.3591061000**	**0.7727719000**
44	Maha.RrCovIdlerShftIdler.164539	185	0.0000000195	0.0000003229
45	Maha.SmpRotPmpShftBlades.162494	64	**0.0581608200**	**0.3562893000**
45	Maha.SmpRotPmpShftBlades.163086	23	**0.2011345000**	**0.4502288000**
45	Maha.SmpRotPmpShftBlades.164539	68	**0.0168386900**	**0.1265429000**
46	Maha.SmpRotPmpShftGear.162494	64	**0.0909359800**	**0.1115068000**
46	Maha.SmpRotPmpShftGear.163086	23	**0.2491190000**	**0.6280054000**
46	Maha.SmpRotPmpShftGear.164539	68	**0.0761383800**	**0.0684803000**
47	Maha.StbdAftInpDrvShftPin.162494	64	**0.0053022850**	**0.0474350100**
47	Maha.StbdAftInpDrvShftPin.163086	23	**0.1613175000**	**0.7514104000**
47	Maha.StbdAftInpDrvShftPin.164539	68	**0.4358976000**	**0.1771743000**
48	Maha.StbdNGBEngInpShftPin.162494	147	0.0000048610	**0.0006049340**
48	Maha.StbdNGBEngInpShftPin.163075	78	0.0001173497	0.0002072291
48	Maha.StbdNGBEngInpShftPin.163086	113	**0.0012476810**	**0.1539161000**
48	Maha.StbdNGBEngInpShftPin.164539	271	0.0000440885	0.0000395430
49	Maha.StbdNGBFCDrvShftGear.162494	90	**0.6208552000**	**0.7571400000**
49	Maha.StbdNGBFCDrvShftGear.163075	39	**0.1480693000**	**0.0486905700**
49	Maha.StbdNGBFCDrvShftGear.163086	38	**0.0754701900**	**0.1372257000**
49	Maha.StbdNGBFCDrvShftGear.164539	216	**0.0095454600**	**0.0036564800**
50	Maha.StbdNGBOPDrvShftSpur.162494	90	**0.4632375000**	**0.3308799000**
50	Maha.StbdNGBOPDrvShftSpur.163075	39	**0.2272758000**	**0.1876801000**
50	Maha.StbdNGBOPDrvShftSpur.163086	38	**0.4782304000**	**0.3899293000**
50	Maha.StbdNGBOPDrvShftSpur.164539	216	**0.0231873300**	**0.0004167416**
51	Maha.StbdNGBOutShftPin.162494	147	0.0000952857	0.0000941570
51	Maha.StbdNGBOutShftPin.163075	78	**0.3298978000**	**0.9418615000**
51	Maha.StbdNGBOutShftPin.163086	113	**0.3007083000**	**0.0625590400**
51	Maha.StbdNGBOutShftPin.164539	271	0.0000540052	0.0000003801
52	Maha.StbdNGBTachShftSpur.162494	90	**0.0060790820**	**0.0756782300**
52	Maha.StbdNGBTachShftSpur.163075	39	**0.6471191000**	**0.3443324000**
52	Maha.StbdNGBTachShftSpur.163086	38	**0.2916541000**	**0.0624214400**
52	Maha.StbdNGBTachShftSpur.164539	216	**0.0036782130**	**0.0021731880**
53	Maha.Stg1HydPmpShftSpur.162494	83	**0.0141408600**	**0.2324533000**
53	Maha.Stg1HydPmpShftSpur.163075	34	**0.5397494000**	**0.3713293000**
53	Maha.Stg1HydPmpShftSpur.164539	185	**0.0431039200**	**0.1071620000**
54	Maha.Stg1PlntShftGear.162494	64	**0.1353007000**	**0.0868184400**
54	Maha.Stg1PlntShftGear.163086	23	**0.2491190000**	**0.6823691000**
54	Maha.Stg1PlntShftGear.164539	68	**0.1026168000**	**0.0334438000**
55	Maha.Stg2PlntShftGear.162494	64	**0.0330268900**	**0.0175327600**
55	Maha.Stg2PlntShftGear.163086	23	**0.3062189000**	**0.2805596000**
55	Maha.Stg2PlntShftGear.164539	68	**0.1368143000**	**0.2409825000**
56	Maha.Stg2SunShftGear.162494	64	**0.0805968300**	**0.1295930000**

Index	Mahalanobis Distance Name	No	Chi-Square gof	KS gof
56	Maha.Stg2SunShftGear.163086	23	**0.1286267000**	**0.3223521000**
56	Maha.Stg2SunShftGear.164539	68	**0.0019751810**	**0.0828582000**
57	Maha.TRTakeoffShftSpur.162494	345	0.0000000030	0.0000003194
57	Maha.TRTakeoffShftSpur.163075	141	0.0000603434	**0.0018286950**
57	Maha.TRTakeoffShftSpur.163086	235	**0.0002357552**	0.0001431754
57	Maha.TRTakeoffShftSpur.164539	565	0.0000000000	0.0000002932
58	Maha.TGBInpShftGear.162494	19	**0.2955570000**	**0.5046692000**
59	Maha.TGBInpShftPin.162494	19	**0.5818332000**	**0.8910370000**
60	Maha.TGBOilPmpShftBlades.162494	19	**0.2955570000**	**0.2851809000**
60	Maha.TGBOilPmpShftBlades.163086	20	**0.7404781000**	**0.8606168000**
61	Maha.TGBOilPmpShftGear.162494	19	**0.3765676000**	**0.4244366000**
61	Maha.TGBOilPmpShftGear.163086	20	**0.5195206000**	**0.2929667000**
62	Maha.TGBOutShftGear.162494	173	0.0000062168	0.0000383685
62	Maha.TGBOutShftGear.163075	80	**0.1425494000**	**0.0092620290**
62	Maha.TGBOutShftGear.163086	147	0.0000000418	0.0000286994
62	Maha.TGBOutShftGear.164539	296	**0.0633566000**	**0.7861269000**
63	Maha.TTOIdlerShaftIdlerSpur.162494	345	0.0000001031	0.0000073137
63	Maha.TTOIdlerShaftIdlerSpur.163075	141	**0.0105821400**	**0.0096087650**
63	Maha.TTOIdlerShaftIdlerSpur.163086	235	**0.0014733450**	**0.0475561100**
63	Maha.TTOIdlerShaftIdlerSpur.164539	565	0.0000000000	0.0000006539

Table 7. Goodness of Fit Test Results for Exponential Distribution Using Bonferroni Correction

APPENDIX C WARNING AND ALARM THRESHOLD LEVELS

		THRESHOLD	
GEAR NAME	TAIL NUMBER	WARNING	ALARM
#2 Eng F C Drv Shft Spur	162494	87.887	175.774
#2 Eng F C Drv Shft Spur	163075	81.888	163.777
#2 Eng F C Drv Shft Spur	163086	80.927	161.854
#2 Eng FrWh Shft Cam Gear	162494	77.459	154.918
#2 Eng FrWh Shft Cam Gear	163075	87.868	175.736
#2 Eng FrWh Shft Cam Gear	163086	84.500	169.000
#2 Eng FrWh Shft Cam Gear	164539	75.094	150.188
#2 Eng FrWh Shft Drv Spur	162494	106.877	213.755
#2 Eng FrWh Shft Drv Spur	163075	95.255	190.509
#2 Eng FrWh Shft Drv Spur	163086	105.643	211.285
#2 Eng FrWh Shft Drv Spur	162494	105.091	210.182
#2 Eng FrWh Shft Drv Spur	163075	77.942	155.883
#2 Eng FrWh Shft Drv Spur	163086	82.881	165.761
#2 Eng Inp Shft Spur	162494	121.918	243.835
#2 Eng Inp Shft Spur	163075	154.378	308.755
#2 Eng Inp Shft Spur	163086	108.283	216.567
#2 Eng Tach Shft Spur	162494	85.158	170.317
#2 Eng Tach Shft Spur	163075	106.152	212.305
#2 Eng Tach Shft Spur	163086	104.602	209.205
#2 Eng Tach Shft Spur	164539	84.834	169.668
#2 Gen Shft Spur	162494	73.849	147.698
#2 Gen Shft Spur	163075	105.301	210.601
#2 Gen Shft Spur	163086	93.128	186.256
#2 Gen Shft Spur	164539	91.362	182.723
#2 Inp Shft Aft Idler	162494	80.911	161.821
#2 Inp Shft Aft Idler	163075	98.821	197.642
#2 Inp Shft Aft Idler	163086	74.685	149.371
#2 Inp Shft Aft Idler	164539	68.484	136.968
#2 Inp Shft Idler	162494	94.419	188.838
#2 Inp Shft Idler	163075	115.084	230.167
#2 Inp Shft Idler	163086	87.307	174.615
#2 Inp Shft Idler	164539	80.806	161.612
#2 Inp Shft Pin	162494	127.066	254.133
#2 Inp Shft Pin	163075	124.458	248.915
#2 Inp Shft Pin	163086	126.940	253.881
#2 Inp Shft Pin	164539	92.120	184.239
AGB Act Shft Idler	162494	89.184	178.368
AGB Act Shft Idler	163075	76.160	152.320
AGB Act Shft Idler	163086	86.037	172.073
AGB Act Shft Idler	164539	73.954	147.907

		THRESHOLD	
GEAR NAME	TAIL NUMBER	WARNING	ALARM
AGB Act Shft Spur	162494	82.259	164.518
AGB Act Shft Spur	163075	122.253	244.507
AGB Act Shft Spur	163086	119.947	239.894
AGB Act Shft Spur	164539	76.829	153.659
AGB Drv Shft Gear	162494	76.835	153.669
AGB Drv Shft Gear	163075	107.125	214.250
AGB Drv Shft Gear	163086	93.148	186.295
AGB Drv Shft Gear	164539	85.585	171.170
AGB Drv Shft Spur	162494	89.687	179.374
AGB Drv Shft Spur	163075	99.513	199.026
AGB Drv Shft Spur	163086	109.734	219.468
AGB Drv Shft Spur	164539	99.164	198.328
AGB Eng Strt Shft Spur	162494	64.040	128.080
AGB Eng Strt Shft Spur	163075	89.406	178.812
AGB Eng Strt Shft Spur	163086	81.641	163.281
AGB Eng Strt Shft Spur	164539	85.205	170.409
AGB Gen #1 Shft Spur	162494	78.054	156.108
AGB Gen #1 Shft Spur	163075	86.926	173.851
AGB Gen #1 Shft Spur	163086	76.632	153.264
AGB Gen #1 Shft Spur	164539	72.800	145.600
AGB Gen #3 Shft Spur	162494	78.310	156.619
AGB Gen #3 Shft Spur	163075	83.297	166.594
AGB Gen #3 Shft Spur	163086	80.509	161.017
AGB Gen #3 Shft Spur	164539	71.327	142.654
AGB O P Shft Spur	162494	65.855	131.711
AGB O P Shft Spur	163075	97.878	195.756
AGB O P Shft Spur	163086	85.815	171.629
AGB O P Shft Spur	164539	80.545	161.089
AGB Stg2 Srv Pmp Shft Spur	163075	81.008	162.016
AGB Stg2 Srv Pmp Shft Spur	163086	89.387	178.774
AGB Stg2 Srv Pmp Shft Spur	164539	86.943	173.886
AGB Ut Pmp Shft Spur	162494	73.659	147.317
AGB Ut Pmp Shft Spur	163075	92.068	184.136
AGB Ut Pmp Shft Spur	163086	100.069	200.138
AGB Ut Pmp Shft Spur	164539	79.165	158.330
AGB Wch Pmp Shft Spur	162494	92.945	185.890
AGB Wch Pmp Shft Spur	163075	91.347	182.694
AGB Wch Pmp Shft Spur	163086	111.340	222.680
Aux Lb Vn Pmp Shft Blades	162494	70.706	141.412
Aux Lb Vn Pmp Shft Blades	163086	74.667	149.334
Aux Lb Vn Pmp Shft Blades	164539	75.454	150.909
Aux Lb Vn Pmp Shft Gear	162494	68.671	137.342
Aux Lb Vn Pmp Shft Gear	163086	82.445	164.891
Aux Lb Vn Pmp Shft Gear	164539	73.366	146.732

		THRESHOLD	
GEAR NAME	TAIL NUMBER	WARNING	ALARM
Grnd Stg 1 Ring	162494	126.197	252.394
Grnd Stg 1 Ring	163086	107.926	215.852
Grnd Stg 1 Ring	164539	104.857	209.714
Grnd Stg 2 Ring	162494	94.023	188.047
Grnd Stg 2 Ring	163086	86.832	173.665
Grnd Stg 2 Ring	164539	76.246	152.492
IGB Inp Shft Pin	162494	153.372	306.745
IGB Inp Shft Pin	163086	102.281	204.561
IGB Out Shft Gear	163075	122.761	245.521
IGB Out Shft Gear	164539	90.691	181.382
IGB Out Shft Pump Blades	163075	107.090	214.180
IGB Out Shft Pump Blades	164539	81.243	162.486
Main Rtr Shft OP Spur	162494	91.471	182.943
Main Rtr Shft OP Spur	163086	105.649	211.297
Main Rtr Shft OP Spur	164539	78.317	156.634
Main Rtr Tach Shft Spur	162494	93.856	187.711
Main Rtr Tach Shft Spur	163075	88.492	176.984
Main Rtr Tach Shft Spur	163086	107.083	214.165
Main Rtr Tach Shft Spur	164539	94.542	189.083
Oil Cool Shft Spur	162494	57.687	115.373
Oil Cool Shft Spur	163075	88.019	176.037
Oil Cool Shft Spur	163086	118.071	236.141
Oil Cool Shft Spur	164539	103.964	207.929
Outer Shaft Main Bev	162494	128.397	256.793
Outer Shaft Main Bev	163086	99.226	198.452
Outer Shaft Main Bev	164539	97.479	194.959
Outer Shaft Sun Gear	162494	98.550	197.100
Outer Shaft Sun Gear	163086	117.402	234.805
Outer Shaft Sun Gear	164539	86.412	172.823
Port Aft Inp Drv Shft ACC Pi	162494	65.226	130.451
Port Aft Inp Drv Shft ACC Pi	163075	82.126	164.253
Port Aft Inp Drv Shft ACC Pi	163086	62.519	125.037
Port Aft Inp Drv Shft Pin	162494	176.083	352.165
Port Aft Inp Drv Shft Pin	163086	115.667	231.334
Port Aft Inp Drv Shft Pin	164539	142.257	284.514
Port NGB Eng Inp Shft Pin	162494	116.989	233.978
Port NGB Eng Inp Shft Pin	163075	126.625	253.251
Port NGB Eng Inp Shft Pin	163086	88.253	176.505
Port NGB Eng Inp Shft Pin	164539	93.199	186.398
Port NGB F C Drv Shft Gear	162494	82.212	164.423
Port NGB F C Drv Shft Gear	163075	152.383	304.766
Port NGB F C Drv Shft Gear	163086	84.661	169.322
Port NGB F C Drv Shft Gear	164539	122.758	245.516

GEAR NAME	TAIL NUMBER	THRESHOLD WARNING	THRESHOLD ALARM
Port NGB F C Drv Shft LH Zerl	162494	80.069	160.138
Port NGB F C Drv Shft LH Zerl	163075	104.075	208.149
Port NGB F C Drv Shft LH Zerl	163086	78.103	156.206
Port NGB F C Dvn Shft LH Zerl	162494	104.766	209.532
Port NGB F C Dvn Shft LH Zerl	163075	132.108	264.217
Port NGB F C Dvn Shft LH Zerl	163086	82.404	164.809
Port NGB F C Dvn Shft LH Zerl	164539	86.576	173.153
Port NGB O P Drv Shft Spur	162494	110.329	220.658
Port NGB O P Drv Shft Spur	163075	170.428	340.855
Port NGB O P Drv Shft Spur	163086	106.975	213.951
Port NGB Out Shft ACC Spur	162494	89.195	178.389
Port NGB Out Shft ACC Spur	163075	88.519	177.039
Port NGB Out Shft ACC Spur	163086	101.779	203.558
Port NGB Out Shft ACC Spur	164539	74.920	149.840
Port NGB Out Shft Gear	162494	122.729	245.457
Port NGB Out Shft Gear	163075	104.362	208.724
Port NGB Out Shft Gear	164539	85.718	171.436
Port NGB Tach Shft Spur	162494	109.157	218.313
Port NGB Tach Shft Spur	163075	163.493	326.985
Port NGB Tach Shft Spur	163086	121.593	243.186
Rr Cov Idler Shft Idler	163075	97.377	194.754
Rr Cov Idler Shft Idler	163086	79.261	158.521
Smp Rot Pmp Shft Blades	162494	72.580	145.160
Smp Rot Pmp Shft Blades	163086	104.873	209.747
Smp Rot Pmp Shft Blades	164539	80.353	160.706
Smp Rot Pmp Shft Gear	162494	73.926	147.852
Smp Rot Pmp Shft Gear	163086	98.650	197.300
Smp Rot Pmp Shft Gear	164539	75.347	150.694
Stbd Aft Inp Drv Shft Pin	162494	82.952	165.904
Stbd Aft Inp Drv Shft Pin	163086	99.848	199.697
Stbd Aft Inp Drv Shft Pin	164539	131.998	263.997
Stbd NGB Eng Inp Shft Pin	162494	88.123	176.246
Stbd NGB Eng Inp Shft Pin	163086	109.231	218.461
Stbd NGB F C Drv Shft Gear	162494	73.778	147.556
Stbd NGB F C Drv Shft Gear	163075	101.009	202.018
Stbd NGB F C Drv Shft Gear	163086	93.446	186.892
Stbd NGB F C Drv Shft Gear	164539	90.709	181.418
Stbd NGB O P Drv Shft Spur	162494	94.256	188.512
Stbd NGB O P Drv Shft Spur	163075	98.499	196.999
Stbd NGB O P Drv Shft Spur	163086	138.161	276.321
Stbd NGB O P Drv Shft Spur	164539	119.092	238.185
Stbd NGB Out Shft Pin	163075	93.032	186.064
Stbd NGB Out Shft Pin	163086	107.960	215.920

		THRESHOLD	
GEAR NAME	TAIL NUMBER	WARNING	ALARM
Stbd NGB Tach Shft Spur	162494	99.473	198.945
Stbd NGB Tach Shft Spur	163075	104.134	208.268
Stbd NGB Tach Shft Spur	163086	138.476	276.952
Stbd NGB Tach Shft Spur	164539	105.081	210.162
Stg 1 Hyd Pmp Shft Spur	162494	75.381	150.761
Stg 1 Hyd Pmp Shft Spur	163075	89.620	179.240
Stg 1 Hyd Pmp Shft Spur	164539	93.821	187.641
Stg 1 Plnt Shft Gear	162494	113.529	227.057
Stg 1 Plnt Shft Gear	163086	106.430	212.861
Stg 1 Plnt Shft Gear	164539	101.408	202.816
Stg 2 Plnt Shft Gear	162494	108.400	216.800
Stg 2 Plnt Shft Gear	163086	77.163	154.327
Stg 2 Plnt Shft Gear	164539	80.559	161.118
Stg 2 Sun Shft Gear	162494	89.453	178.906
Stg 2 Sun Shft Gear	163086	97.528	195.056
Stg 2 Sun Shft Gear	164539	81.921	163.843
T R Takeoff Shft Spur	163075	84.947	169.893
T R Takeoff Shft Spur	163086	84.517	169.033
TGB Inp Shft Gear	162494	102.241	204.483
TGB Inp Shft Pin	162494	141.919	283.838
TGB Oil Pmp Shft Blades	162494	90.062	180.123
TGB Oil Pmp Shft Blades	163086	95.846	191.692
TGB Oil Pmp Shft Gear	162494	77.453	154.906
TGB Oil Pmp Shft Gear	163086	102.383	204.765
TGB Out Shft Gear	163075	99.003	198.007
TGB Out Shft Gear	164539	87.637	175.275
TTO Idler Shaft Idler Spur	163075	93.493	186.985
TTO Idler Shaft Idler Spur	163086	77.718	155.437

Table 8. Warning and Alarm Threshold Levels

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Bechhoefer, <Eric.Bechhoefer@goodrich.com> "Need short info about data set." [E-mail to Mehmet Elyurek <melyurek@nps.navy.mil>]. 01 October 2003.

Brockwell, Peter J. and Davis, Richard A., *Introduction to Time Series and Forecasting, First Edition*, Springer Texts in Statistics, 1996.

Burn Statistics Web Site, [http://www.burns-stat.com/pages/Working/ljungbox.pdf], October 2003.

Chatfield, Chris, *The Analysis of Time Series An Introduction, Fifth Edition,* pp.11,19-20, Chapman & Hall, 1996.

Eric Weistein's World of Mathematics Web Site, [http://mathworld.wolfram.com/BonferroniCorrection.html], October 2003.

Goodrich Corporation Fuel & Utility Systems Report, *Mechanical Diagnostics Frequently Asked Questions*, by Goodrich Corporation Fuel & Utility Systems, 2003.

NIST SEMATECH Engineering Statistics Handbook Web Site, [http://www.itl.nist.gov/div898/handbook/eda/section3], September 2003.

NIST SEMATECH Engineering Statistics Handbook Web Site, [http://www.itl.nist.gov/div898/handbook/pmc/section4], September 2003.

Proposal 03P108 submitted in response to Defense Advanced Research Projects Agency (DARPA), *Prognosis of Rotating Machinery Health*, by D. Hochmann, E. Bechhoefer, R. Hess, M. Mitrovic, M. Roemer, C. Byington, A. Sarlashkar, A. Bayoumi, A. Reynolds, J. Kiddy, R. Read, L. Whitaker and D. Pines, 2003.

Ragsdale, Cliff T., *Spreadsheet Modelling and Decision Analysis*, Third Edition, p.509, South-western College Publishing, 2001.

S-PLUS 2000 Guide to Statistics Volume 2, pp.173, 177, 179, Data Analysis Products Division MathSoft, Inc., 1999.

SSI Scientific Software Web Site, [http://www.ssicentral.com/lisrel/lis00465.htm], October 2003.

Thermo Galactic Web Site, [http://www.galactic.com/algorithms/discrim_mahaldist.htm], October 2003.

INITIAL DISTRIBUTION LIST

1. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

2. Professor Samuel E. Buttrey
   Department of Operations Research
   Naval Postgraduate School
   Monterey, CA

3. Professor Robert Koyak
   Department of Operations Research
   Naval Postgraduate School
   Monterey, CA

4. Eric Bechhoefer, Ph.D.
   Goodrich Fuels and Utility Systems
   Vergennes, VT