Fitting:
A Subroutine to Fit Four-Moment Probability Distributions to Data

S. R. Winterstein, C.H. Lange, S. Kumar
Civil Engineering Department
Stanford University
Stanford, CA 94305
FITTING:
A SUBROUTINE TO FIT FOUR-MOMENT PROBABILITY DISTRIBUTIONS TO DATA

S.R. Winterstein, C.H. Lange, S. Kumar
Civil Engineering Department
Stanford University
Stanford, CA 94305

Sandia Contract: AC-9051

Abstract

FITTING is a Fortran subroutine that constructs a smooth, generalized four-parameter probability distribution model. It is fit to the first four statistical moments of the random variable $X$ (i.e., average values of $X$, $X^2$, $X^3$, and $X^4$) which can be calculated from data using the associated subroutine calmom. The generalized model is produced from a cubic distortion of the parent model, calibrated to match the first four moments of the data. This four-moment matching is intended to provide models that are more faithful to the data in the upper tail of the distribution. Examples are shown for two specific cases.
Acknowledgement

This program, developed at Stanford University, was supported by Sandia National Laboratories Wind Energy Technology Department, and the Reliability of Marine Structures Program of the Civil Engineering Department at Stanford University.
Contents

Executive Summary ................................................. 7

1 The fitting Subroutine ........................................... 8
   1.1 What fitting Does ........................................ 8
       1.1.1 Overview of Capabilities ............................ 9
       1.1.2 Subroutine calmom .................................. 11
       1.1.3 How to Read or Not Read This Manual ............... 11
   1.2 fitting Input and Output ................................ 12
   1.3 calmom Input and Output ................................ 14
   1.4 The Driver Program ...................................... 14

2 Fatigue Load Modelling: A Generalized Weibull Example ...... 17
   2.1 Wind Turbine Loads Example .............................. 17
   2.2 Alternate Usage of fitting ............................... 21

3 Extreme Values: A Generalized Gumbel Example ............... 23
   3.1 Generalized Gumbel Results ............................. 24
   3.2 Comparison of Three Generalized Distributions for Wave Height . 25

4 Technical Background and Additional Details ................... 29
### CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Motivation</td>
<td>29</td>
</tr>
<tr>
<td>4.2</td>
<td>Underlying Methodology: fitting</td>
<td>30</td>
</tr>
<tr>
<td>4.3</td>
<td>Underlying Methodology: calmom</td>
<td>30</td>
</tr>
<tr>
<td>4.4</td>
<td>Notes on Usage</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>4.4.1 Errors in Matching Moments</td>
<td>33</td>
</tr>
<tr>
<td></td>
<td>4.4.2 Lower Tail Limiting Values</td>
<td>33</td>
</tr>
</tbody>
</table>

References                                                                 | 34   |

A Driver Source Code Listing                                               | 35   |
Executive Summary

**fitting** is a Fortran subroutine that constructs a smooth, generalized four-parameter probability distribution model. It is fit to the first four statistical moments of the random variable $X$ (i.e., average values of $X$, $X^2$, $X^3$, and $X^4$) which can be calculated from data using the associated subroutine **calmom**.

This distribution is said to be "generalized" in that it generalizes three conventional, standard two-parameter "parent" distribution models. The user may select here between Gaussian, Gumbel, or Weibull parent models. The generalized model is produced from a cubic distortion of the parent model, calibrated to match the first four moments of the data. This four-moment matching is intended to provide models that are more faithful to the data in the upper tail of the distribution.

Examples are shown here for two specific cases: modelling rainflow-counted load ranges and extreme wave heights, based respectively on Weibull and Gumbel parent models. To use **fitting** to fit a distribution to data, a separate subroutine, **calmom**, is included to determine the first four statistical moments of the input data set. Because these moments are required input to **fitting**, the routines **calmom** and **fitting** together serve as a general distribution fitting algorithm. A sample driver program is included to illustrate the usage and interpretation of **fitting** and **calmom** for the two examples.
Chapter 1

The fitting Subroutine

1.1 What fitting Does

fitting is a Fortran subroutine that constructs a smooth, generalized four-parameter probability distribution model. The first four statistical moments of the random variable $X$ (i.e., average values of $X$, $X^2$, $X^3$, and $X^4$) are used by the subroutine to establish the generalized distribution. These moments can be based on theory; however, they are almost always derived from data. A separate subroutine calmom is provided to compute the required moments for an arbitrary data set.

This distribution is said to be “generalized” in that it generalizes three conventional, standard two-parameter “parent” distribution models. The user may select here between Gaussian, Gumbel, or Weibull parent models. The generalized model is then produced from a cubic distortion of the parent model, calibrated to match the first four moments of the data. (Depending on the numerical values of the moments, an inverse cubic distortion may also be used.) This four-moment matching is intended to provide models that are more faithful to the extreme values of the data, commonly referred to as the upper tail region.

By invoking various parent models, the user is able to reflect reasonable “prior” probability distribution choices based on the context at hand. For example, values from a random process may be assigned Gaussian distribution if sampled at an arbitrary time, Weibull distribution if sampled at an arbitrary peak, or Gumbel distribution if sampled at a global peak in a fixed duration (Benjamin and Cornell, 1970). These three distributions are included here as possible parent distribution choices.

Examples are shown here for two specific cases: modelling rainflow-counted load ranges and extreme wave heights, based respectively on Weibull and Gumbel parent models. Notably, we find that over a range of practical values, these applications are controlled by the four moment values and are relatively insensitive to the underlying parent distribution choice. Because this conclusion may change with the application,
1.1. What fitting Does

<table>
<thead>
<tr>
<th>Values $x_i$ [kip-ft]:</th>
<th>Probabilities $p_i = F(x_i)$:</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.250</td>
<td>0.004</td>
</tr>
<tr>
<td>9.500</td>
<td>0.058</td>
</tr>
<tr>
<td>9.750</td>
<td>0.146</td>
</tr>
<tr>
<td>10.000</td>
<td>0.251</td>
</tr>
<tr>
<td>10.250</td>
<td>0.360</td>
</tr>
<tr>
<td>10.500</td>
<td>0.465</td>
</tr>
<tr>
<td>10.750</td>
<td>0.561</td>
</tr>
<tr>
<td>11.000</td>
<td>0.645</td>
</tr>
<tr>
<td>12.000</td>
<td>0.862</td>
</tr>
<tr>
<td>13.000</td>
<td>0.949</td>
</tr>
<tr>
<td>14.000</td>
<td>0.981</td>
</tr>
<tr>
<td>15.000</td>
<td>0.993</td>
</tr>
</tbody>
</table>

Table 1.1: Predicted probabilities, $p_i$, of not exceeding various $x$ levels.

fitting allows the user to implement various parent distributions and assess the sensitivity to this choice.

### 1.1.1 Overview of Capabilities

The subroutine fitting has two options. In the first option, the user can provide arbitrary values $x_1, \ldots, x_N$ of the physical variable, and the routine returns corresponding probability values, $p_i$, that the random variable is less than the value $x_i$. Formally, $p_i$ is known as $F(x_i)$, the cumulative distribution function (CDF) of $X$. The second option works in the opposite direction: the user provides specified cumulative probability values $p_i$, and the routine returns levels $x_i$ of the physical variable. In this case, the output levels $x_i$ are known as specific fractiles of the probability distribution. Note that both options require the first four statistical moments of $X$ to be input.

As a simple example, consider the edgewise bending moment range $X$ on a wind turbine blade (Coleman, 1989). Figure 1.1 shows the cycle counts (rainflow counted) for a 71 minute time history of observed edgewise moments. The clustering of counts in the moment ranges around 9-15 and 0-5 kip-ft is attributed to the dominant gravity induced loading combined with the turbulent effects of the wind respectively. From the viewpoint of fatigue damage, ranges less than 9 kip-ft contribute less than 5% to the total damage and are considered insignificant. Table 1.1 shows the cumulative distribution, $F(x)$, of applied loads above this level as predicted by fitting.

For example, Table 1.1 shows that among all loads above $x_{\text{min}}=9$ [kip-ft], the load level 9.5 [kip-ft] is not exceeded 5.8% of the time—and hence exceeded the remaining
94.2% of the time. Similarly, this table shows that a typical, central load value is around 10.5 [kip-ft]. Strictly, this value is not exceeded 46.5% of the time; i.e., the median value of the load $X$—which has 50% chance of being exceeded—is between 10.5 and 10.75 [kip-ft]. We show in Section 2.2 how these values are estimated from the fitting routines. We also show how, if we invoke the second option of fitting, we can directly estimate the median level (for which $F(x)=p_i$ is specified to be 0.5) to be 10.59 [kip-ft]. Notice also from Table 1.1 that probabilities for values that exceed the range of observed values can be requested.

In general, one may consider three distinct ways to use fitting:

- One option of the subroutine fitting takes input values $x_i$ and estimates the cumulative probability $p_i$ of falling below each $x_i$. For example, data in the first column of Table 1.1) are input and the second column values are output.

- With this same option, the differences $p_i - p_{i-1}$ between these cumulative probabilities can be used to give estimates of a theoretical histogram (i.e., probability content in various discretized “cells” or “bins.”) For example, Table 1.1 can be used to directly produce a histogram, with probability .032 assigned to the interval (9.25, 9.50), probability .126-.032=.094 to the interval (9.50, 9.75), and so forth.

- The other option of the subroutine fitting takes input values $p_i$ and returns corresponding values $x_i$. For example, data in the second column of Table 1.1

Figure 1.1: Histogram of Edgewise Bending Moment Time Series Data

![Histogram of Edgewise Bending Moment Time Series Data](image)
are input and the first column values are output. This is useful, for example, if the user can more easily specify interesting values of \( p_i \) rather than \( x_i \) values a priori.

### 1.1.2 Subroutine `calmom`

`fitting` requires the first four statistical moments of the random variable, \( X \), as input. These moments can be based on either theoretical considerations or derived from a particular set of data. To use `fitting` to fit a distribution to data, a separate subroutine, `calmom`, is included to accurately estimate the first four statistical moments of the input data set. Input and output for the `calmom` subroutine are described in Section 1.3.

Because these moments are required input to `fitting`, the routines `calmom` and `fitting` together serve as a general distribution fitting algorithm. A sample driver program is included to illustrate the usage and interpretation of `fitting` and `calmom` for two example problems given in Chapters 2 and 3.

### 1.1.3 How to Read or Not Read This Manual

We recognize that there are two distinct types of computer users: those who read manuals thoroughly and those who go to great lengths to avoid doing so. For this latter group, who prefer to learn by example, we have included a driver program with detailed comments, and the sample input and output used to generate Table 1.1. Those users may wish to proceed to the driver source code listing, also given in Appendix A. Additional description of the driver and this example, based on a generalized Weibull model, is given in Chapter 2. Chapter 3 describes an alternate application to extreme wave height levels, using a generalized Gumbel model.

Those who prefer a more precise overview of `fitting` are referred to the remainder of Chapter 1. Section 1.2 describes its input and output arguments and calling syntax, while Section 1.3 discusses the usage and arguments of the subroutine `calmom` which computes statistical moments for a given data set.

Finally, Chapter 4 brings together a number of more detailed technical issues concerning `fitting`. These range from underlying motivation (Section 4.1) to the basic methodology underlying `fitting` (Section 4.2) and `calmom` (Section 4.3). Section 4.4 includes various additional practical notes on their usage, limitations and potential error conditions.
1.2 fitting Input and Output

The user can call fitting with the following command:

\[
\text{call fitting}(\text{itype}, \text{xmom}, \text{ndata}, \text{xmin}, \text{x}, \text{cdf}, \text{nx}, \text{pmom}, \text{iflag}, \text{ioout}, \text{etol})
\]

Each component of the fitting argument list is defined below. Output quantities include the array pmom and, depending on the value of iflag, either x or cdf. All other quantities are input.

- **itype**: index used to define the parent distribution used by fitting
  
  \( \text{itype} = 1 \): Gaussian distribution
  
  \( \text{itype} = 2 \): Gumbel distribution
  
  \( \text{itype} = 3 \): Weibull distribution

- **xmom(1)**\(^1\): mean, \( \mu_x = E[X] = \int_{-\infty}^{\infty} xf(x)dx \); \( f(x) \) = PDF of \( X \)

- **xmom(2)**\(^1\): standard deviation, \( \sigma_x = \{ E[(X-\mu_x)^2] \}^{1/2} = \{ \int_{-\infty}^{\infty} (x-\mu_x)^2 f(x)dx \}^{1/2} \)

- **xmom(3)**\(^1\): skewness, \( \alpha_3 = E[(\frac{x-\mu_x}{\sigma_x})^3] = \int_{-\infty}^{\infty} (\frac{x-\mu_x}{\sigma_x})^3 f(x)dx \)

- **xmom(4)**\(^1\): kurtosis, \( \alpha_4 = E[(\frac{x-\mu_x}{\sigma_x})^4] = \int_{-\infty}^{\infty} (\frac{x-\mu_x}{\sigma_x})^4 f(x)dx \)

- **ndata**\(^1\): Number of sample data used to estimate moments in xmom. If \( \text{ndata} < 100 \), fitting checks by simulation that these moments do not have excessive bias (Section 4.3). The user can avoid this simulation check by setting \( \text{ndata} \geq 100 \) on input.

- **xmin**: Optional shift parameter to be applied in the Weibull case (\( \text{itype} = 3 \)). Note that the standard Weibull model produces values for \( X \geq 0 \), while the Gaussian and Gumbel models are unbounded. If the user inputs a nonzero value of xmin, a shifted Weibull model (standard Weibull model of \( X - \text{xmin} \)) is then used as a parent distribution. Accordingly, in this case xmom(1) ... xmom(4) should contain moments of the shifted variable \( X - \text{xmin} \). (This is precisely what is returned by the routine calmom when xmin is nonzero.) The data shown in figure 1.1 is a good example of using this variable (xmin \( \approx 9.0 \)) when only the upper portion of the data is important. Note finally that fitting ignores the value of xmin if the Gaussian or Gumbel distribution is selected.

- **x**: array containing values of the physical variable.

\(^1\)Section 1.3 explains these moments further and subroutine calmom used to compute them
1.2. fitting Input and Output

- **cdf**: array containing the cumulative (non-exceedance) probabilities corresponding to each \( x \), in the \( x \) array above.
- **nx**: number of \( x \) or cdf values requested.
- **pmom**: array of the absolute moments of the fitted distribution:
  \[
  pmom(n) = \text{the } n\text{-th absolute moment, } E[X^n] = \int \limits_{-\infty}^{\infty} x^n f(x) \, dx
  \]
  The first four moments will be consistent with the input moments given in \( xmom \), within the error tolerance described below. Higher moments may be of interest in other applications; e.g., fatigue damage of various materials. Here \( n=10 \) moments are output, using the probability density function \( f(x) \) estimated by the Generalized distribution model.
- **iflag**: index used to define the type of calculation to be performed by **fitting**
  - \( iflag = 0 \): returns output estimates of \( x \) for each of the cumulative probabilities input in the array \( cdf \).
  - \( iflag = 1 \): returns output estimates of probabilities \( cdf \) for each of the physical values input in the array \( x \).
- **ioout**: logical unit number for writing error messages. The calling program should make a file available for error messages by opening a file with \( ioout \) as its logical unit number. (The sample driver illustrates this in Chapter 2.)
- **etol**: the error tolerance in matching higher moments. This is defined formally in Eq. 4.2. In general, there is a tradeoff between moment accuracy and computation time. Based on experience with various tolerances, we use the value \( etol=.01 \) in our examples. This may be changed by the user.

If the theoretical moments \( xmom(i) \) are known, the **fitting** routine can be applied directly. In practice, it is often necessary to estimate these statistical moments from a set of data. A separate subroutine, **calmom**, is supplied here to compute the required moments from data; i.e., to act as a pre-processor for **fitting** routine. The procedure used to compute these moments is discussed in section 4.3, and its use is demonstrated in Chapter 2.

**The Role of the Lower Threshold \( x_{\text{min}} \)**. In most applications of the Weibull model we seek to model all possible values of a positive quantity (e.g., stress range, number of cycles to failure, etc.). In certain applications, however, the user may wish to impose a non-zero lower-bound \( x_{\text{min}} \). This is useful, for example, if we wish to exclude lower values as non-physical, or due to a fundamentally different probability distribution. We have found this useful, for example, in modelling some edgewise bending loads on a turbine blade. In this case, we seek to exclude small, non-damaging loads produced by a different mechanism: low amplitude (turbulence induced) cycles superimposed on marked gravity-driven bending moment ranges. This case is illustrated further in the example of Chapter 2.
1.3 calmom Input and Output

The subroutine calmom estimates the first four statistical moments \( \text{xmom}(i), i=1...4 \), from an input data set. It can thus serve as a preprocessor to fitting.

The calmom argument list is:

```fortran
subroutine calmom(xmom,data,ndata,nrmax,xmin,itype)
```

The input to calmom are the following:

- **data**: array containing the data for which the moments are to be calculated.
- **ndata**: number of data points in array data.
- **nrmax**: dimension size of array data (should be consistent with that used in calling program).
- **xmin**: threshold value of the physical variable, as used in fitting (see description of xmin in section 1.2 and the example problem in Chapter 2).
- **itype**: index used to define the parent distribution used by fitting
  
  - \( \text{itype} = 1 \): Gaussian distribution
  - \( \text{itype} = 2 \): Gumbel distribution
  - \( \text{itype} = 3 \): Weibull distribution

The sample data input via the array data, can be arranged in any order and does not need to be sorted in increasing magnitude as shown in the example input of Table 2.1. Also, calmom screens the array data removing any values that are below the threshold xmin.

On output the array xmom contains the sample moments of the data, as defined in section 1.2. These can then be used directly as input to the routine fitting. The theoretical background for calmom is described in Section 4.3.

1.4 The Driver Program

A single driver program is used to demonstrate the use of subroutines fitting and calmom for two examples.

In general, the source code is distributed in three separate files:
1.4. The Driver Program

calmom.for: The \texttt{calmom} subroutine to estimate moments from an input data set.

fitting.for: The \texttt{fitting} subroutine, to use these moments to estimate the entire
distribution function of $X$.

driver.for: A separate driver program that calls these routines (listing in Appendix
A).

This driver program is included to help speed the reader's understanding and imple-
mentation of \texttt{fitting}. The example shown here can thus be run without creating
any additional source code. One needs merely to compile and link the three source
codes listed above, and execute with the input files provided.

Of course, prospective users are encouraged to modify the driver program accord-
ing to their needs. Toward that end, it is hoped that \texttt{driver.for} can provide a useful
template. For those users who prefer to learn by example, we recommend reading the
source code of \texttt{driver.for} as a useful starting point.

Analysis Steps. As implemented in \texttt{driver.for}, the analysis proceeds in the
following steps:

1. Read control data: \texttt{itype, xmin, iflag}, and the array \texttt{cdf} or \texttt{x} used as input
to \texttt{fitting}.

2. Read input data: the array \texttt{data} used as input to \texttt{calmom}, which calculates the
necessary moment input for \texttt{fitting}.

3. Call \texttt{calmom} to estimate moments.

4. Call \texttt{fitting} to estimate the corresponding full distribution function.

5. Write results.

These steps are clearly delineated in comments contained within the source code of
\texttt{driver}.

File Architecture. In the current coding of the driver (Appendix A), two input
files are expected:

driver.in: Input file containing control data read in step 1 above.

driver.dat: Input file containing physical data read in step 2 above.

Together with the three source code files, we are distributing input files for two
examples: (1) \texttt{weibull.in, weibull.dat}; and (2) \texttt{gumbel.in, gumbel.dat}. The user
Chapter 1. The fitting Subroutine

should note that to implement one of these examples, the input files *.in and *.dat (*='weibull' or 'gumbel') should be copied to driver.in and driver.dat before executing. The corresponding output file, driver.out, should then agree with the file *.out that has been distributed.

The examples described in Chapters 2 and 3 provide tables that identify more clearly the contents of the files driver.in and driver.dat.
Chapter 2

Fatigue Load Modelling: A Generalized Weibull Example

This chapter describes the first example, which relates to fatigue load modelling. The next chapter describes an alternate application to extreme wave height modelling.

2.1 Wind Turbine Loads Example

This example concerns the edgewise bending moments shown in figure 1.1 of Chapter 1 (Coleman, 1989). We consider here 8913 values of $X =$ bending moment range [kip-ft], as found by rainflow counting (Fuchs and Stephens, 1980). The data are stored in the file weibull.dat. Table 2.1 gives a partial listing of these values. For the sake of clarity they are input in increasing order; however, this is not required by the program.

A separate analysis of these data (Winterstein and Lange, 1994) has shown that bending moment ranges below 9 kip-ft do not contribute to the total fatigue damage given by this data set. Since the application for the load model is a fatigue analysis of the HAWT blade, we choose to fit the model above a lower threshold $x_{\text{min}} = 9$ [kip-ft]. Note that only 4819 ranges are above this threshold.

This threshold is set in the first line of the input file weibull.in. Table 2.2 lists this file. The first line also contains the values $i\text{flag}=1$ and $i\text{type}=3$. The value $i\text{flag}=1$ indicates that values of $X$ are to follow on the subsequent lines in the file, and the program is to calculate corresponding cdf values. The value $i\text{type}=3$ tells the fitting routine to select Weibull as the parent distribution. The remaining lines list the actual $X$ values requested, which are the same as in column 1 of Table 1.1.

Output. The driver produces a single output file, driver.out, which we have stored here as weibull.out. Table 2.3 lists weibull.out for our example. The
Table 2.1: Abridged listing of edgewise moment ranges [kip-ft] from weibull.dat.
File contains column 2 data only; line numbers are inserted here for clarity.

<table>
<thead>
<tr>
<th>Line number i:</th>
<th>Data $x_i$ [kip-ft]:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0190</td>
</tr>
<tr>
<td>2</td>
<td>0.0190</td>
</tr>
<tr>
<td>3</td>
<td>0.0190</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>4092</td>
<td>8.9810</td>
</tr>
<tr>
<td>4093</td>
<td>8.9820</td>
</tr>
<tr>
<td>4094</td>
<td>8.9820</td>
</tr>
<tr>
<td>4095</td>
<td>9.0010</td>
</tr>
<tr>
<td>4096</td>
<td>9.0010</td>
</tr>
<tr>
<td>4097</td>
<td>9.0010</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>8911</td>
<td>18.0420</td>
</tr>
<tr>
<td>8912</td>
<td>18.9290</td>
</tr>
<tr>
<td>8913</td>
<td>20.4990</td>
</tr>
</tbody>
</table>

Table 2.2: Listing of input file, weibull.in, with control data for driver program.
2.1. Wind Turbine Loads Example

Fractile results reported by fitting are precisely those given in Chapter 1 (Table 1.1). Note also that the output confirms that 4819 data points have been found above the input lower threshold of $x=9$. It also reports the first four moments from these data, as estimated by calmom, that are used as input to fitting. Finally, the model predicts the first 10 absolute moments, $E[X^n]$. Note that these are consistent with the first four moments found for the data. For example, $E[X^1]=1.831$, the mean value $\mu_x$, while $E[X^2]=\sigma_x^2 + \mu_x^2$, or $1.154^2 + 1.831^2=4.685$. Similarly, the predicted third and fourth moments can be shown to be consistent with those estimated from the data. The main virtue of the routine, of course, is that it seeks to predict still higher moments more accurately—through introduction of a smooth distribution model—than would be possible from the data alone.

Figure 2.1 compares the fitted distribution function $F(x)$ with the data. (These results have been obtained by running the fitting routine over a larger range of $x$ values than shown in the example.) Results are shown on “Weibull probability scale,” on which the parent Weibull model appears as a straight line. It appears that the generalized Weibull model reflects the curvature of the data shown on this scale, particularly in the upper tail of interest (which is most heavily weighted by the third and fourth moments).
Lower Threshold Value: 9.000
Number of Data Processed: 4819

** MOMENT RESULTS **

Mean: 1.831
Standard Deviation: 1.154
Skewness: 1.552
Kurtosis: 7.449

** FRACTILE RESULTS (FITTING) **

X: CDF:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>9.250</td>
<td>0.004</td>
</tr>
<tr>
<td>9.500</td>
<td>0.058</td>
</tr>
<tr>
<td>9.750</td>
<td>0.146</td>
</tr>
<tr>
<td>10.000</td>
<td>0.251</td>
</tr>
<tr>
<td>10.250</td>
<td>0.360</td>
</tr>
<tr>
<td>10.500</td>
<td>0.465</td>
</tr>
<tr>
<td>10.750</td>
<td>0.561</td>
</tr>
<tr>
<td>11.000</td>
<td>0.645</td>
</tr>
<tr>
<td>12.000</td>
<td>0.862</td>
</tr>
<tr>
<td>13.000</td>
<td>0.949</td>
</tr>
<tr>
<td>14.000</td>
<td>0.981</td>
</tr>
<tr>
<td>15.000</td>
<td>0.993</td>
</tr>
</tbody>
</table>

** PREDICTED MOMENTS (FITTING) **

N: E[X**N]:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>0.183E+01</td>
</tr>
<tr>
<td>2.000</td>
<td>0.469E+01</td>
</tr>
<tr>
<td>3.000</td>
<td>0.159E+02</td>
</tr>
<tr>
<td>4.000</td>
<td>0.688E+02</td>
</tr>
<tr>
<td>5.000</td>
<td>0.377E+03</td>
</tr>
<tr>
<td>6.000</td>
<td>0.259E+04</td>
</tr>
<tr>
<td>7.000</td>
<td>0.221E+05</td>
</tr>
<tr>
<td>8.000</td>
<td>0.234E+06</td>
</tr>
<tr>
<td>9.000</td>
<td>0.302E+07</td>
</tr>
<tr>
<td>10.000</td>
<td>0.472E+08</td>
</tr>
</tbody>
</table>

Table 2.3: Listing of output file, weibull.out, produced by driver program.
2.2. Alternate Usage of fitting

Finally, we illustrate the \texttt{iflag=0} option of fitting. For example, if the \texttt{weibull.in} content is modified as shown in Table 2.4, Table 2.5 shows the corresponding output. The data file \texttt{weibull.dat} remains the same, and hence all moment results are unchanged. The only difference is that in this case, the distribution fractiles $x$ have been evaluated at the requested probability levels given in the input file \texttt{driver.in} (Table 2.4). For example, as noted in Chapter 1, the median value of $X$ (with 50\% chance of being exceeded) is found to be 10.589. The values of bending moment that are not exceeded 99\% and 99.9\% of the time were also determined.

\begin{table}[h]
\begin{tabular}{ll}
9.00 & 0.3 \\
0.10 & \\
0.20 & \\
0.30 & \\
0.40 & \\
0.50 & \\
0.60 & \\
0.70 & \\
0.80 & \\
0.90 & \\
0.99 & \\
0.999 & \\
\end{tabular}
\caption{Listing of input file for \texttt{iflag=0} option.}
\end{table}
Lower Threshold Value: 9.000
Number of Data Processed: 4819

** MOMENT RESULTS **

Mean: 1.831
Standard Deviation: 1.154
Skewness: 1.552
Kurtosis: 7.449

** FRACTILE RESULTS (FITTING) **

X: CDF:

9.627 0.100
9.882 0.200
10.114 0.300
10.345 0.400
10.589 0.500
10.861 0.600
11.188 0.700
11.620 0.800
12.324 0.900
14.664 0.990
17.255 0.999

** PREDICTED MOMENTS (FITTING) **

N: E[X**N]:

1.000 0.183E+01
2.000 0.469E+01
3.000 0.159E+02
4.000 0.688E+02
5.000 0.377E+03
6.000 0.259E+04
7.000 0.221E+05
8.000 0.234E+06
9.000 0.302E+07
10.000 0.472E+08

Table 2.5: Listing of output file, generated from iflag=0 input file given in Table 2.4.
Chapter 3

Extreme Values: A Generalized Gumbel Example

This chapter illustrates the use of the \texttt{fitting} routine to fit a generalized Gumbel distribution to extreme values. The driver program used for this demonstration is described in Chapter 1. This driver program reads the relevant input data for this example and passes them to the \texttt{calmom} and \texttt{fitting} routines to construct the generalized Gumbel distribution.

This example concerns the significant wave height $H_s$ in a Southern North Sea location, for which 19 years of hindcast data are available (Danish Hydraulic Institute, 1989). For each of these 19 years, a single storm event has been identified with maximum significant wave height $H_s$ (i.e. the extreme values). This value ranges from $H_s = 6.92\text{m} (1972/1973)$ to $9.66\text{m} (1981/1982)$. A sorted list of all 19 values is reported in Table 3.1.

This chapter has two sections. The first section deals with the generalized Gumbel model for the significant wave height data. The second section compares the three different generalized distribution models for the same data set.

Finally, it should be noted that a generalized Gumbel model has previously been fit to this data set (Winterstein and Haver, 1991). The results shown here are an improvement in two senses: (1) \texttt{fitting} permits greater accuracy to be achieved in matching moments; and (2) \texttt{fitting} includes an inverse cubic transformation, which is particularly important in reflecting the narrower-than-Gumbel tails similar to the data in Table 3.1.
Table 3.1: Listing of annual significant wave height [m] from gumbel.dat. File contains column 2 data only; line numbers are inserted here for clarity.

### 3.1 Generalized Gumbel Results

The annual significant wave height data consists of 19 values listed in Table 3.1. The data are stored in the file gumbel.dat. For the sake of clarity they are input in decreasing order; however, this is not required by the program.

The control data are stored in gumbel.in. Table 3.2 lists this file. The first line of this file contains three values. The first value is $\text{xmin}=0.0$, which sets the lower threshold value. Note, however, that this is not used in this case of Gumbel distribution since there is no cutoff value. This threshold value is used when generalized Weibull distribution is fit to the data (see Chapter 2). The second argument, $\text{iflag}=1$, indicates that $x$ values are to follow in the file and the program will calculate corresponding cdf values. The third argument, $\text{itype}=2$, indicates that a generalized Gumbel distribution is to be fit to the data in file gumbel.dat. The remaining lines list the actual $x$ values requested.

**Output.** Table 3.3 lists the corresponding output file gumbel.out for this example. It also reports the first four moments from these data, as estimated by calmom,
which are used as input to fitting. Finally, the model predicts the first 10 absolute moments, $E[X^n]$. Note that these are consistent with the first four moments found for the data. For example, $E[X^1] = 8.275$, the mean value $\mu_x$, while $E[X^2] = \sigma_x^2 + \mu_x^2$, or $0.819^2 + 8.275^2 = 69.1$. Similarly, the predicted third and fourth moments can be shown consistent with those estimated from the data.

As discussed in Chapter 2, note that to use the input files `gumbel.in` and `gumbel.dat` with the driver program they must be copied to the files `driver.in` and `driver.dat` respectively. The output, written to `driver.out`, should then be comparable to `gumbel.out`.

In order to generate a smooth plot of the generalized Gumbel distribution, an input file similar to `driver.in` with a greater number of input values to compute corresponding CDF values was used. This distribution is plotted in Figure 3.1 along with the observed data values. It appears to capture fairly well the systematic curvature of the data on the Gumbel probability scale used.

### 3.2 Comparison of Three Generalized Distributions for Wave Height

Because we deal here with annual extreme values, the Gumbel distribution is the natural choice of parent distribution. We may ask, however, what effect is achieved if
Number of Data Processed: 19

** MOMENT RESULTS **

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>8.275</td>
</tr>
<tr>
<td>SD</td>
<td>0.819</td>
</tr>
<tr>
<td>Skewness</td>
<td>-0.053</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>1.905</td>
</tr>
</tbody>
</table>

** FRACTILE RESULTS (FITTING) **

<table>
<thead>
<tr>
<th>X</th>
<th>CDF</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.000</td>
<td>0.998</td>
</tr>
<tr>
<td>9.660</td>
<td>0.978</td>
</tr>
<tr>
<td>9.440</td>
<td>0.937</td>
</tr>
<tr>
<td>9.180</td>
<td>0.846</td>
</tr>
<tr>
<td>9.170</td>
<td>0.842</td>
</tr>
<tr>
<td>8.850</td>
<td>0.690</td>
</tr>
<tr>
<td>8.790</td>
<td>0.662</td>
</tr>
<tr>
<td>8.600</td>
<td>0.580</td>
</tr>
<tr>
<td>8.580</td>
<td>0.572</td>
</tr>
<tr>
<td>8.540</td>
<td>0.557</td>
</tr>
<tr>
<td>8.490</td>
<td>0.539</td>
</tr>
<tr>
<td>8.090</td>
<td>0.427</td>
</tr>
<tr>
<td>8.080</td>
<td>0.424</td>
</tr>
<tr>
<td>8.060</td>
<td>0.419</td>
</tr>
</tbody>
</table>

** PREDICTED MOMENTS (FITTING) **

<table>
<thead>
<tr>
<th>N</th>
<th>E[X**N]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000</td>
<td>0.827E+01</td>
</tr>
<tr>
<td>2.000</td>
<td>0.691E+02</td>
</tr>
<tr>
<td>3.000</td>
<td>0.583E+03</td>
</tr>
<tr>
<td>4.000</td>
<td>0.496E+04</td>
</tr>
<tr>
<td>5.000</td>
<td>0.426E+05</td>
</tr>
<tr>
<td>6.000</td>
<td>0.369E+06</td>
</tr>
<tr>
<td>7.000</td>
<td>0.322E+07</td>
</tr>
<tr>
<td>8.000</td>
<td>0.282E+08</td>
</tr>
<tr>
<td>9.000</td>
<td>0.250E+09</td>
</tr>
<tr>
<td>10.000</td>
<td>0.222E+10</td>
</tr>
</tbody>
</table>

Table 3.3: Listing of output file, gumbel.out, produced by driver program.
3.2. Comparison of Three Generalized Distributions for Wave Height

Figure 3.1: Generalized Gumbel Distribution for Annual Extreme Wave Height Data.
a different choice of parent distribution is selected. This is investigated in this section.

We again use the same data set as listed in Table 3.1. Thus the input data file is same as driver.dat of Section 3.1. However, the control input file driver.in is varied so that itype is either 1, 2, or 3. These three different values of itype give three generalized distributions: generalized Gaussian (itype = 1), generalized Gumbel (itype = 2), and generalized Weibull (itype = 3).

The three distribution are shown in Figure 3.2. The figure shows wave height results up to the 1000-year fractile, i.e. for which $p=.999$ and hence $-\ln(-\ln(p))=6.9$. The pattern of variation follows that of the underlying parent distributions: the Weibull has the narrowest upper tail and hence predicts the lowest extreme values, while the Gumbel predicts the largest. Most notably, however, all three parent distributions predict quite similar wave heights over this domain of interest.

This suggests that knowledge of four moments is sufficient to control the tail behavior of interest. This apparent robustness of the four-moment description is encouraging, particularly in cases where the optimal parent distribution is not obvious. Of course this conclusion may be problem-dependent; the user is encouraged to vary the choice of itype for the problem at hand.
Chapter 4

Technical Background and Additional Details

4.1 Motivation

The fitting routine has been developed to modify standard, commonly used distributions to better match observed tail behavior. In particular, cubic distortions of these standard “parent” distributions are sought to match the first four moments of the data. We may then ask why precisely four moments are used to fit the probability distribution of $X$—and not two, three, five, ten, etc. Conventional models are of lower order, requiring only one or two moments. The problem is that a number of plausible models, with very different tail behavior and hence fatigue reliability, can be fit to the same first two moments. This scatter in reliability estimates is said to be produced by model uncertainty. This is prevalent in low-order, one- or two-moment models. (Note that many common fatigue load models include only one parameter; e.g., the Rayleigh and exponential models.)

To avoid this model uncertainty, which is difficult to quantify, one is led to try to preserve higher moments as well. This will help to discriminate between various models, and hence reduce model uncertainty. The benefit does not come without cost, however: higher moments are more sensitive to rare extreme outcomes, and hence are more difficult to estimate from a limited data set. This is known as statistical uncertainty, which reflects the limitations of our data set.

Thus, our search for an “optimal” model reflects an attempt at balance between model and statistical uncertainties. Practical experience (e.g., Winterstein, 1988) suggests that four moments are often sufficient to define upper distribution tails over the range of interest. This experience motivates the generalized models developed here. It is again supported by the results of Section 3.2, in which extreme wave heights are insensitive to the choice of parent distribution, once four moments have been specified.
4.2 Underlying Methodology: fitting

The fitting routine begins with a theoretical, 2-parameter "parent" distribution. In the current implementation, the user may choose Gaussian, Gumbel, or Weibull parent distributions. Denoting this parent variable as \( U \), fitting then models the physical random variable \( X \) through a cubic transformation of \( U \):

\[
X = c_0 + c_1 U + c_2 U^2 + c_3 U^3 \tag{4.1}
\]

The optimizer adjusts the coefficients \( c_n \) through an iterative scheme until the difference between the requested skewness, \( \alpha_3 \), and kurtosis, \( \alpha_4 \), (see \texttt{xmom(3)} and \texttt{xmom(4)}, section 1.2) and those of the generalized model in Eq. 4.1 (\( \alpha_{3X} \) and \( \alpha_{4X} \)) are minimized.

The optimizer also restricts the coefficients so that Eq. 4.1 remains monotonically increasing, producing a well-behaved model that only mildly modifies the underlying parent distribution. This leads, for example, to requiring \( c_3 \geq 0 \) so that \( X \) in Eq. 4.1 continues to grow as \( U \) becomes large. This in turn makes it difficult to model cases with tails that are narrower than those of the parent distribution. In particular, it is difficult to use Eq. 4.1, with positive \( c_3 \), to model situations in which the desired kurtosis, \( \alpha_4 \), is less than that of the parent variable, \( \alpha_{4U} \). In this case fitting inverts the model, seeking to fit a model analogous to Eq. 4.1 in which the roles of \( X \) and \( U \) are interchanged. (In this view, one seeks to expand the distribution tail of the actual variable \( X \) to produce a parent variable \( U \), so that \( c_3 \) becomes positive.)

Note that this switching between two dual models, based on the size of \( \alpha_4 \), occurs automatically within fitting and should be of no consequence to the user. Adding such a dual model, however, has been found to greatly increase modelling flexibility for small kurtosis cases. These have been found to arise both in extreme and fatigue loading applications.

Finally, in whichever form the model is defined, the coefficients \( c_n \) are chosen to minimize the error \( \epsilon \), defined as

\[
\epsilon = \sqrt{(\alpha_3 - \alpha_{3X})^2 + (\alpha_4 - \alpha_{4X})^2} \tag{4.2}
\]

The speed of executing fitting is governed largely by the speed of this optimization; i.e., by the amount of effort (trial \( c_n \) values) needed to achieve an acceptably small tolerance, \( \epsilon_{\text{tol}} \). The driver program sets \( \epsilon_{\text{tol}} = .01 \) for the examples shown. The user can vary this tolerance, with the resulting change in computation time to be expected.

4.3 Underlying Methodology: calmom

To motivate the need for this routine, we must consider a brief background in statistical moment estimation. If we seek to estimate the ordinary mean value \( E[X] = \mu \)
4.3. Underlying Methodology: calmom

from data $X_1, \ldots, X_n$, a natural estimate is the simple average value $\overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$. Similarly, the $k$-th order "ordinary" moment, $E[X^k]$, is naturally estimated by the corresponding average $\frac{1}{n} \sum_{i=1}^{n} X_i^k$.

The difficulties arise when we instead seek, as in many applications, to estimate not ordinary but central moments; i.e., of the form $\mu_k = E[(X - \mu)^k]$ for $k=2, 3, 4, \ldots$ (Note that fitting input stops at $k=4$: $\sigma_x = \mu_2^{1/2}$, $\alpha_3 = \mu_3 / \mu_2^{3/2}$, and $\alpha_4 = \mu_4 / \mu_2^{2}$.)

The problem here lies in its circular aspect: we must first estimate the unknown first moment $\mu$ before seeking to estimate $\mu_k = E[(X - \mu)^k]$. And, if we use the same data set for both purposes, we typically find too-low estimates of $\mu_2, \mu_3, \mu_4$, etc. because our $\mu$ value is artificially tuned to best match the mean of the observations. Those exposed to a standard statistics course will best recognize this phenomenon when estimating the variance $\mu_2$: to inflate the sample variance to account for this bias, the sum of squared deviations is divided by $n - 1$ rather than $n$.

While unbiased estimates of the higher moments $\mu_3, \mu_4, \ldots$ are less familiar, they are available in the statistical literature (Fisher, 1928):

$$\mu = \overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i \quad (4.3)$$

$$\mu_2 = \frac{n}{n - 1} m_2 \quad (4.4)$$

$$\mu_3 = \frac{n^2}{(n - 1)(n - 2)} m_3 \quad (4.5)$$

$$\mu_4 - 3\mu_2^2 = \frac{n^2}{(n - 1)(n - 2)(n - 3)} [(n + 1)m_4 - 3(n - 1)m_2^2] \quad (4.6)$$

in terms of the sample central moment $m_k = \sum_{i=1}^{n} (X_i - \overline{X})^k / n$. Eq. 4.4 is the conventional result for the sample variance.

Remaining Bias. Finally, the routine calmom uses these results to estimate the quantities $\sigma_x$ by $\mu_2^{0.5}$, $\alpha_3$ by $\mu_3 / (\mu_2^{1.5})$, and $\alpha_4$ by $\mu_4 / (\mu_2^{2})$. Because these vary nonlinearly with $\mu_n$, they may still contain some bias although the $\mu_n$ estimates do not.

For example, if we fit a Gumbel model to the 19 wave height data from Chapter 3, the true skewness and kurtosis values are 1.14 and 5.40. However, simulating 10000 data sets of size $n=19$ and running each through calmom, we find on average the skewness 0.79 and kurtosis 3.89 (Winterstein and Haver, 1991).

To address this problem, the fitting routine has an automatic check for remaining bias through simulation. This is why ndata is given as an input parameter. After fitting constructs a distribution with moments from the input array xmom, many similar data sets (of size ndata) are simulated from this distribution. If the moments
predicted from calmom differ appreciably on average from the input values, new theoretical estimates of the moments are constructed. This estimation-simulation loop is continued iteratively until satisfactory convergence is found.

Note that the fitting routine does not perform this simulation if its input parameter ndata $\geq 100$. This value can be hard-wired if the user wishes to bypass this option. Figure 4.1 shows the effect of enabling this "unbiased" option (the default) and disabling it (using "raw" moments from calmom directly) for the generalized Gumbel model produced for the example given in Chapter 3. There is relatively little difference found in these cases. Larger effects may be found for cases of (1) fewer data and/or (2) distributions with broader tails.

4.4 Notes on Usage

The fitting routine has limiting conditions that users should note. When these conditions are encountered, appropriate error messages are written to the output file/device corresponding to the input logical unit number ioout. This section explains the meaning of these messages and discusses other details regarding fitting usage.
4.4.1 Errors in Matching Moments

Note that in most practical cases, the coefficients $c_n$ in Eq. 4.1 can be chosen so that the error $\epsilon$, as given in Eq. 4.2, falls within the user-defined tolerance limit $\epsilon_{tol}$. In rare cases the minimized error exceeds $\epsilon_{tol}$. In these cases fitting writes an error message indicating the magnitude of $\epsilon$, the error norm of the skewness and kurtosis in Eq. 4.2.

4.4.2 Lower Tail Limiting Values

Lower tail limiting values are only a problem when the parent distribution is Weibull. In this case the variable $U$ in Eq. 4.1 has Weibull distribution, and hence a minimum value of 0. Because Eq. 4.1 is monotonic, the corresponding smallest possible value of the physical variable $X$ is $c_0$. This physical lower limit $c_0$ can be either greater or less than zero, since the optimized Weibull model in Eq. 4.1 will not in general have its $x$ intercept at exactly zero.

This may lead to situations that seem anomalous. If the lower limit $c_0$ is negative, for example, fitting may estimate negative values of $X$ for probabilities near zero. Conversely, if the lowest possible value $c_0$ is positive, an input $X$ value below $c_0$ cannot occur and fitting will return a zero cumulative probability (CDF=0). When $x_{\text{min}}$ is not zero the situation is entirely analogous: $c_0$ may be greater or less than $x_{\text{min}}$.

In practice we believe this to be a minor issue for the following reasons:

- The routine fitting is intended for applications where large $X$ values (upper distribution tails) are crucial. This is the motivation for preserving higher moments. Its accuracy at the lower end of the distribution may not be of great concern.

- If we wish to preserve a positive range of values, one can easily introduce a transformation to the data. For example, apply fitting not to the physical variable $X$ but rather $Y=\ln(X)$, based on the first four moments of $Y$. Then the reverse transformation $X=\exp(Y)$ will still be positive.

- The routine fitting is intended for applications where the true distribution is not too different than a Weibull model would predict. In such cases we may expect the nonlinear terms (proportional to $c_0$, $c_2$, and $c_3$) in Eq. 4.1 to be relatively small on average relative to the linear term. Thus, compared to the range of likely variation of $X$ values, $c_0$ may seem to lie rather "close" to zero.
References


Appendix A

Driver Source Code Listing

C=====================================================================================================================================
C
C THIS PROGRAM DEMONSTRATES THE USE OF THE SUBROUTINES
C
C CALMOM ... CALCULATES FOUR MOMENTS OF A GIVEN INPUT DATA SET
C FITTING... USES THESE MOMENTS TO FIT A GENERALIZED
C (GAUSSIAN, GUMBEL, WEIBULL) DISTRIBUTION FUNCTION
C
C INPUT FILES: DRIVER.DAT... INPUT DATA USED TO ESTIMATE MOMENTS
C DRIVER.IN... CONTROL DATA USED IN CALLING FITTING
C
C (NOTE: SAMPLE EXAMPLE FILES *.DAT AND *.IN SHOULD BE COPIED TO DRIVER.DAT AND DRIVER.IN BEFORE EXECUTING)
C
C OUTPUT FILE: DRIVER.OUT
C
C USAGE: COMPILER AND LINK DRIVER, CALMOM, AND FITTING
C
C=====================================================================================================================================

PROGRAM DRIVER
C=====================================================================================================================================
IMPLICIT REAL*8 (A-H,O-Z)
PARAMETER ( NDMIN = 20000 , NMAX = 2000 )
DIMENSION DATA(NMAX),CDF(NMAX),X(NMAX),XMOM(4),PMOM(10)

C=====================================================================================================================================
IODAT = 10
IOIN  = 11
IOOUT = 12
OPEN(IODAT,FILE='driver.dat')
OPEN(IOIN,FILE='driver.in')
OPEN(IOOUT,FILE='driver.out')

C-----------------------------------------------READ CONTROL DATA
C
C   XMIN   LOWER THRESHOLD DATA VALUE USED IN THE ANALYSIS
C   IFLAG  INPUT/OUTPUT FLAG:
          IFLAG = 0.....FINDS X FOR INPUT CDF VALUES
          IFLAG = 1.....FINDS CDF FOR INPUT X VALUES
C   ITYPE  CONTROL VARIABLE TO CHOOSE THE GENERALIZED
          DISTRIBUTION TYPE
          ITYPE = 1 :FIT GENERALIZED GAUSSIAN DISTRIBUTION
          ITYPE = 2 :FIT GENERALIZED GUMBEL DISTRIBUTION
          ITYPE = 3 :FIT GENERALIZED WEIBULL DISTRIBUTION
C   X(NXMAX) LEVELS OF X AT WHICH DISTRIBUTION IS REPORTED
C   CDF(NXMAX) CDF VALUES (NON-EVEXCESSION PROBABILITIES) FOR
          EACH X LEVEL
C   NXMAX  MAXIMUM NUMBER OF X OR CDF VALUES REQUESTED
C   NX     ACTUAL NUMBER OF X OR CDF VALUES REQUESTED

READ(IOIN,*) XMIN,IFLAG,ITYPE
IF (ITYPE .NE. 3) XMIN = 0.d0
IF (IFLAG.EQ.0) THEN
   DO 10 IX = 1,NXMAX
      10 READ(IOIN,*,ERR=30,END=30) CDF(IX)          READ CDF VALUES IF IFLAG=0
C
ELSE
   DO 20 IX = 1,NXMAX
      20 READ(IOIN,*,ERR=30,END=30) X(IX)              READ X VALUES IF IFLAG=1
C
ENDIF
30 NX = IX - 1

C-----------------------------------------------READ INPUT DATA
C
C   DATA(NDMAX) ARRAY OF INPUT DATA FOR WHICH MOMENTS ARE FOUND
C   NDMAX  MAXIMUM NUMBER OF INPUT DATA PERMISSIBLE
C   NDATA  ACTUAL NUMBER OF INPUT DATA
C
NDATA=0
40 READ(IODAT,*,ERR=50,END=50) X1
   NDATA=NDATA+1
   DATA(NDATA)=X1
GO TO 40
CALL CALMOM TO ESTIMATE MOMENTS

XMOM(4) ARRAY OF FOUR MOMENTS COMPUTED FROM DATA:
XMOM(1) = MEAN
XMOM(2) = STANDARD DEVIATION
XMOM(3) = SKEWNESS
XMOM(4) = KURTOSIS

CALL CALMOM(XMOM, DATA, NDATA, NDMIN, XMIN, ITYPE)

WRITE OUTPUT

IF (ITYPE .EQ. 3) THEN
WRITE(IOOUT,900) ' Lower Threshold Value:', XMIN
ENDIF
WRITE(IOOUT, *) ' Number of Data Processed:', NDATA
WRITE(IOOUT, *) ' ** MOMENT RESULTS **
WRITE(IOOUT, *) ' Mean:', XMOM(1)
WRITE(IOOUT, *) ' Standard Deviation:', XMOM(2)
WRITE(IOOUT, *) ' Skewness:', XMOM(3)
WRITE(IOOUT, *) ' Kurtosis:', XMOM(4)
WRITE(IOOUT, *)

CALL FITTING TO ESTIMATE X FOR GIVEN CDF (IFLAG=0) OR CDF FOR GIVEN X (IFLAG=1)

ETOL ERROR TOLERANCE IN MATCHING OBSERVED MOMENTS
... HERE WE ACCEPT 0.01 ERROR---USER CAN ALTER

PMOM(10) ARRAY OF PREDICTED ABSOLUTE MOMENTS FROM MODEL
PMOM(1) = PREDICTED AVERAGE OF X**N, N=1..10

ETOL = .01DO
call FITTING(ITYPE, XMOM, NDATA, XMIN, X, CDF, NX, PMOM, IFLAG, IOOUT, ETOL)

WRITE OUTPUT

** FRAC TILE RESULTS (FITTING) **
WRITE(IOOUT, *) ' X: CDF:'
WRITE(IOOUT, *)
DO 60 IX = 1, NX

60    WRITE(IOOUT,901) X(IX),CDF(IX)
    WRITE(IOOUT, * ) ', '
    WRITE(IOOUT, * ) ' ** PREDICTED MOMENTS (FITTING) **'
    WRITE(IOOUT, * ) ', '
    WRITE(IOOUT, * ) ',
               N: E[X**N]: '
    WRITE(IOOUT, * ) ',
               DO 70 IX = 1, 10
    70    WRITE(IOOUT,902) REAL(IX),PMOM(IX)

C
900 FORMAT(A26, F10.3)
901 FORMAT(16X,2F10.3)
902 FORMAT(16X, F10.3,E10.3)

C
STOP
END
R. Heffernan  
Kenetech Windpower, Inc.  
6952 Preston Avenue  
Livermore, CA 94550

L. Helling  
Librarian  
National Atomic Museum  
Albuquerque, NM 87185

T. Hillesland  
Pacific Gas and Electric Co.  
3400 Crow Canyon Road  
San Ramon, CA 94583

E. N. Hinrichsen  
Power Technologies, Inc.  
P.O. Box 1058  
Schenectady, NY 12301-1058

S. Hock  
Wind Energy Program  
NREL  
1617 Cole Boulevard  
Golden, CO 80401

W. E. Holley  
Kenetech Windpower  
6952 Preston Avenue  
Livermore, CA 94550

M. A. Ilyan  
Pacific Gas and Electric Co.  
3400 Crow Canyon Road  
San Ramon, CA 94583

B. J. Im  
McGillim Research  
4903 Wagonwheel Way  
El Sobrante, CA 94803

K. Jackson  
Dynamic Design  
123 C Street  
Davis, CA 95616

O. Krauss  
Division of Engineering Research  
Michigan State University  
East Lansing, MI 48825

C. Lange  
Civil Engineering Dept.  
Stanford University  
Stanford, CA 94305

G. G. Leigh  
New Mexico Engineering Research Institute  
Campus P.O. Box 25  
Albuquerque, NM 87131

A. Liniecki  
Mechanical Engineering Department  
San Jose State University  
One Washington Square  
San Jose, CA 95192-0087

R. R. Loose, Director  
Wind/Hydro/Ocean Division  
U.S. Department of Energy  
1000 Independence Ave., SW  
Washington, DC 20585

R. Lynette  
AWT/RLA  
425 Pontius Avenue North  
Suite 150  
Seattle, WA 98109

D. Malcolm  
AWT/RLA  
425 Pontius Avenue North  
Suite 150  
Seattle, WA 98109

J. F. Mandell  
Montana State University  
302 Cableigh Hall  
Bozeman, MT 59717

R. N. Meroney  
Dept. of Civil Engineering  
Colorado State University  
Fort Collins, CO 80521

P. Migliore  
NREL  
1617 Cole Boulevard  
Golden, CO 80401

A. Mikhail  
Zond Systems, Inc.  
13000 Jameson Road  
P.O. Box 1910  
Tchachapi, CA 93561

S. Miller  
162636 NE 19th Place  
Bellevue, WA 98008-2552
R. H. Monroe  
Gougeon Brothers  
100 Patterson Avenue  
Bay City, MI 48706

D. Morrison  
New Mexico Engineering Research Institute  
Campus P.O. Box 25  
Albuquerque, NM 87131

W. Musial  
NREL  
1617 Cole Boulevard  
Golden, CO 80401

NWTC Library  
NREL  
1617 Cole Boulevard  
Golden, CO 80401

V. Nelson  
Department of Physics  
West Texas State University  
P.O. Box 248  
Canyon, TX 79016

G. Nix  
NREL  
1617 Cole Boulevard  
Golden, CO 80401

J. W. Oler  
Mechanical Engineering Dept.  
Texas Tech University  
P.O. Box 4289  
Lubbock, TX 79409

R. Osgood  
NREL  
1617 Cole Boulevard  
Golden, CO 80401

M. Papadakis  
Aero Engineering Department  
Wichita State University  
Wichita, KS 67260-0044

C. Paquette  
The American Wind Energy Association  
122 C Street NW  
Fourth Floor  
Washington, DC 20002

R. G. Rajagopalan  
Aerospace Engineering Department  
Iowa State University  
404 Town Engineering Bldg.  
Ames, IA 50011

R. L. Scheffler  
Southern California Edison Research and Development Dept.  
Room 497  
P.O. Box 800  
Rosemead, CA 91770

L. Schienbein  
Battelle-Pacific Northwest Laboratory  
P.O. Box 999  
Richland, WA 99352

T. Schweizer  
Princeton Economic Research, Inc.  
1700 Rockville Pike Suite 550  
Rockville, MD 20852

J. Sladky, Jr.  
Kinetics Group, Inc.  
P.O. Box 1071  
Mercer Island, WA 98040

M. Snyder  
Aero Engineering Department  
Wichita State University  
Wichita, KS 67208

L. H. Soderholm  
Agricultural Engineering  
Room 213  
Iowa State University  
Ames, IA 50010

K. Starcher  
AEI  
West Texas State University  
P.O. Box 248  
Canyon, TX 79016

W. J. Steeley  
Pacific Gas and Electric Co.  
3400 Crow Canyon Road  
San Ramon, CA 94583

F. S. Stoddard  
Dynamic Design-Atlantic Office  
P.O. Box 1373  
Amherst, MA 01004
W. V. Thompson  
410 Ericwood Court  
Manteca, CA 95336

R. W. Thresher  
NREL  
1617 Cole Boulevard  
Golden, CO 80401

K. J. Touryan  
NREL  
16 17 Colc Boulevard  
Golden. CO 80401

W. A. Vachon  
W. A. Vachon & Associates  
P.O. Box 149  
Manchester, MA 01944

B. Vick  
USDA, Agricultural Research Service  
P.O. Drawer 10  
Bushland, TX 79012

V. Wallace  
FloWind Corporation  
990 A Street, Suite 300  
San Rafael, CA 94901

L. Wendell  
Battelle-Pacific Northwest Laboratory  
P.O. Box 999  
Richland, WA 99352

W. Wentz  
Aero Engineering Department  
Wichita State University  
Wichita, KS 67208

R. E. Wilson  
Mechanical Engineering Dept.  
Oregon State University  
Corvallis, OR 97331

S. R. Winterstein  
Civil Engineering Department  
Stanford University  
Stanford, CA 94305

B. Wolff  
Renewable Energy Program Manager  
Conservation and Renewable Energy System  
6918 NE Fourth Plain Boulevard  
Suite B  
Vancouver, WA 98661

M. Zuteck  
MDZ Consulting  
931 Grove Street  
Kemah, TX 77565

S. Schuck  
Renewable Technology  
Pacific Power  
P.O. Box 5257  
GPO Sydney, New South Wales 2001  
AUSTRALIA

V. Lacey  
Indal Technologies, Inc.  
3570 Hawkestone Road  
Mississauga, Ontario L5C 2V8  
CANADA

A. Laneville  
Faculty of Applied Science  
University of Sherbrooke  
Sherbrooke, Quebec J1K 2R1  
CANADA

B. Masse  
Institut de Recherche d’Hydro-Quebec  
1800, Montee Ste-Julie  
Varennes, Quebec J3X 1S1  
CANADA

I. Paraschivoiu  
Dept. of Mechanical Engineering  
Ecole Polytechnique  
CP 6079  
Succursale A  
Montreal, Quebec H3C 3A7  
CANADA

R. Rangi  
Manager, Wind Technology  
Dept. of Energy, Mines and Resources  
580 Booth 7th Floor  
Ottawa, Ontario K1A 0E4  
CANADA