

Regress+

A Tool for Mathematical Modeling

Version 2.3

© Dr. Michael P. McLaughlin
1993-1999

This software and its documentation are distributed free of charge and may neither be sold nor repackaged for sale in whole or in part.

No modification may be made to any part of this package without the written permission of the author.

Regress+ is provided solely as a service to analysts. Its fitness for any particular purpose is at the discretion of the user. No warranty, expressed or implied, is made for any use or misuse of the software, its output, or its documentation.

Macintosh and SimpleText are registered trademarks of Apple Computer, Inc.
PostScript is a registered trademark of Adobe Systems, Inc.

I would like to thank all those individuals who were kind enough to act as beta-testers for this software package. They provided many helpful comments and suggestions. Any faults remaining are entirely my own.

MPM

For deeds do die, however nobly done,
And thoughts of men do as themselves decay,
But wise words taught in numbers for to run,
Recorded by the Muses, live for ay.

—E. Spenser [1591]

PREFACE

When a textbook may cost as much as your weekly groceries and software packages often retail for more than a thousand dollars, you might wonder about any computer program that is given away for free. Does it do anything useful? Is it any good? And, if it is, then why is it free?

Regress+ (“Regress_plus”) is a software tool for developing univariate mathematical models. These models may be either equations or distributions. As an added bonus, the software will generate files of random variates from a large selection of distributions. Compared to commercial statistical packages, *Regress+* has a relatively limited repertoire. However, in some ways, it is not only superior to those packages but, indeed, unique in both the richness of its functionality and its advanced methodology.

This program and its documentation are the product of several years of experiment and refinement. Originally, it was created because I needed it. Later, it was found to be useful to many others and so it evolved to its present phenotype. To the greatest practicable extent, no effort has been spared to make it as accurate and robust as it could reasonably be. No corners have been cut and nothing avoidable has been left to chance. Within its domain, *Regress+* can be trusted to do exactly what it says it can do. What that is has been explained in detail in the accompanying documentation.

There are several reasons why this package is free. First, given the amount of time I spent on it, and given the hourly rate that are charged to clients for my services, there was no hope that my share of any sales would ever come close to the break-even point. Shareware was a possibility, but that usually just makes the user feel guilty when he/she does not send in the fee. Finally, as a scientist and expatriate from the world of academia, my motivations were partly due to pedagogical urges that never seem to ebb with the recurring tides of experience.

Nothing in *Regress+* is novel. No great discoveries were made during its development. Nevertheless, even though it does not advance the state-of-the-art, it is my sincere hope that it will help to advance the practice of the art.

Please email any comments to the author:

mpmcl@mitre.org

Michael P. McLaughlin
McLean, VA
October, 1999

TABLE OF CONTENTS

TITLE	PAGE
...the very game... (Tutorial)	1
Data	2
From Data to Model	8
Two Ways to Construct a Model	8
Stochastic Information	9
Deterministic Information and Error	19
Finding Optimum Parameters	28
Five Criteria	28
Searching for Optima	31
Summary	31
Is the Model Any Good?	32
The Bootstrap	32
Stochastic Models	33
Deterministic Models	35
Summary	36
How Precise are the Parameters?	37
The Nonparametric Bootstrap	39
Stochastic Models	40
Deterministic Models	42
Parameter Correlation	45
Summary	46
99.999999999% Perfect!	47
Users' Guide	51
Installation	52
Using <i>Regress+</i>	53
Deterministic Modeling	53
Stochastic Modeling	70
Generating Random Samples	73
Summary	74
What Can Possibly Go Wrong?	75
Failure to Converge	75
Convergence to an Incorrect Solution	75
Flat Response	76
Systematic Error	76
Very Slow Progress	77
Wishful Thinking	77
<i>Regress+</i> Reference	79
Input/Output	79
Menus	80
Dialogs	83
Miscellaneous	84

TABLE OF CONTENTS (CONCLUDED)	
TITLE	PAGE
Technical Details	87
Optimization	87
Random Number Generator	87
Random Variates	88
Bootstrap	90
Parametric Bootstrap	90
Nonparametric Bootstrap	91
References	93
Appendix A: A Compendium of Common Probability Distributions	
Appendix B: Error Messages	

LIST OF TABLES

TABLE	PAGE
Table 1. Daytime—Boston, Massachusetts (1995-1997)	3
Table 2. Rate of Production of CN in Comet Hale-Bopp	5
Table 3. Weights for Non-prime Totals for Two Dice	9
Table 4. Eccentricities of Conic Sections	22
Table 5. Orbital Data for Mars	23
Table 6. World Track Records (Men)	26
Table 7. Modeling Criteria	28
Table 8. Daytime Data, Observed and Modeled	43
Table 9. Observed Transitions/Colors for Hydrogen ($1H^1$)	47
Table 10. Bode's Law of Planetary Distance	53
Table 11. Daytime Model—Amplitude Confidence Intervals	62
Table 12. User Model—Valid Operators and Functions	69
Table 13. Stochastic Model—Goodness-of-fit Assessments	70

FIGURE	LIST OF FIGURES	PAGE
Figure 1. Raw Daytime Data		4
Figure 2. Hale-Bopp CN Data		6
Figure 3. Annual Best Baseball Batting Average		7
Figure 4. Batting-Average Data (Histogram and PDF)		11
Figure 5. Batting-Average Data (CDF)		13
Figure 6. Rolls3 Data (Histogram and PDF)		15
Figure 7. Rolls3avg Data (Histogram and PDF)		17
Figure 8. Rolls3avg Data (CDF)		17
Figure 9. Ellipse (eccentricity = 0.5)		22
Figure 10. Kepler's Data and Model		24
Figure 11. World Track Records		27
Figure 12. Distribution of K-S Statistic from Equation 25		34
Figure 13. Least-squares Sine Fit to Daytime Data		37
Figure 14. ML Fit of 50 Radii to a Chi Distribution		41
Figure 15. Comparison of 50 Radii to their Theoretical Rayleigh Distribution		42
Figure 16. Rydberg Data for Hydrogen		48
Figure 17. Setup Dialog		54
Figure 18. Parameter Dialog		56
Figure 19. Display Dialog		57
Figure 20. Graph of Bode Data and Linear Model		58
Figure 21. Working on Confidence Intervals		61
Figure 22. True Exponential Model		64
Figure 23. Log-transformed Model		65
Figure 24. Hale_Bopp Data -- Without Weights		66

FIGURE	LIST OF FIGURES (CONCLUDED)	PAGE
Figure 25.	Hale_Bopp Data -- With Weights	67
Figure 26.	User-Model Dialog	68
Figure 27.	Sample Dialog	74
Figure 28.	An Outline of <i>Regress+</i> Functionality	85

“...the very game...”

HIS mother was a witch (or so they said). Had she not poisoned the glazier's wife with a magic drink and cursed her neighbors' livestock so that they became mad? Did she not wander about the town pestering people with her recipes for medicines? Yet, in spite of such proofs, his filial devotion did not go unrewarded and, with the aid of a good lawyer plus the support of his friend and patron Rudolph II, Emperor of the Romans, King of Germany, Hungary, Bohemia, &c., Archduke of Austria, &c., the old woman made her final exit with less flamboyance than some of His Holy Imperial Majesty's subjects might have wished. However, this is not about the mother but the son—and Mars. Johannes Kepler is remembered, to this day, for his insight and his vision. Even more than his contemporary, Galileo, he is honored not just for what he saw but because he invented a new way of looking.

In astronomy, as in most disciplines, how you look determines what you see and, here, Kepler had a novel approach. He began with data whereas all of his predecessors had begun with circles. The Aristotelian/Ptolemaic syllogism decreed that perfect motion was circular. Heavenly bodies were perfect. Therefore, they moved in circles, however many it took to save appearances.

It took a lot. When, more than a quarter of a century before Kepler, Nicolaus Copernicus finally laid down his compass, he had, quite rightly, placed the Sun in the center of the remaining seven known bodies but he had also increased the number of celestial circles to a record forty-eight!

Kepler commenced his intellectual journey along the same path. Indeed, in those days, it was the only path. After many false starts, however, he realized that a collection of circles just would not work. It was the wrong model; the data demanded something else. Kepler bowed to Nature and, without apology, substituted ellipses for circles. He was the first scientist to subjugate theory to observation in a way that we would recognize and applaud.

Of course, Kepler was in a unique position. Thanks to Tycho Brahe, he had the best data in the world and he was duly impressed. Still, he could have voted the party line and added yet more circles. Sooner or later, he would have accumulated enough parameters to satisfy every significant figure of every measurement. But it was wrong and it was the wrongness of it that impressed Kepler most of all. Although he drew his inspiration from the ancient Pythagoreans and the religious fervor of his own time, his words leave little doubt of his sincerity:

“...Now, as God the maker played,
he taught the game to Nature
whom he created in his image:
taught her the very game
which he played himself...”

Kepler searched through the data and found the game. He learned the rules and showed that, if you played well enough, sometimes even emperors take notice. Today, our motivation is different but the game goes on. We begin, of course, with data.

DATA

There are two kinds of data: measurements and opinions. This discussion will focus exclusively on the former. In fact, although there are useful exceptions in many disciplines, here we shall discuss only quantitative measurements. Adopting this mild constraint provides two enormous advantages. The first is the advantage of being able to speak very precisely, yielding minimal concessions to the vagaries of language. The second is the opportunity to utilize the power of mathematics and, especially, of statistics.

Statistics, albeit a discipline in its own right, is primarily an ever-improving cumulation of mathematical tools for extracting information from data. It is information, not data, that leads ultimately to understanding. Whenever you make measurements, perform experiments, or simply observe the Universe in action, you are collecting data. However, real data always leave something to be desired. There is an open interval of quality stretching from worthless to perfect and, somewhere in between, will be your numbers, your data. Information, on the other hand, is not permitted the luxury of imperfection. It is necessarily correct, by definition. Data are dirty; information is golden.

To examine data, therefore, is to sift the silt of a riverbed in search of gold. Of course, there might not be any gold but, if there is, it will take some knowledge and considerable skill to find it and separate it from everything else. Not only must you know what gold looks like, but you also have to know what sorts of things masquerade as gold. Whatever the task, you will need to know the properties of what you seek and what you wish to avoid, the chemistry of gold and not-gold. It is through these properties one can be separated from the other.

An example of real data is shown in Table 1 and Figure 1.¹ This dataset consists of values for the duration of daytime (sunrise to sunset) at Boston, Massachusetts over three years. The first day of each month has been tabulated along with the longest and shortest days occurring during this period. Daytime has been rounded off to the nearest minute.

What can be said about data such as these? It can be safely assumed that they are correct to the precision indicated. In fact, Kepler's data, for analogous measurements, were much more precise. It is also clear that, at this location, the length of the day varies quite a bit during the year. This is not what one would observe near the Equator but Boston is a long way from tropical climes. Figure 1 discloses that daytime is almost perfectly repetitive from year to year, being long in the (Northern hemisphere) Summer and short in the Winter.

Such qualitative remarks, however, are scarcely sufficient. Any dataset as rich as this one deserves to be further quantified in some way and, moreover, will have to be if the goal is to gain some sort of genuine understanding. With scientific data, proof of understanding implies the capability to make accurate predictions. Qualitative conclusions are, therefore, inadequate.

Quantitative understanding starts with a set of well-defined *metrics*. There are several such metrics that may be used to summarize/characterize any set of N numbers, y_i , such as these daytime values. The most common is the *total-sum-of-squares*, TSS, defined in Equation 1.

¹ see file *Examples:Daytime.in* [FAM95]

Table 1. Daytime—Boston, Massachusetts (1995-1997)

Daytime (min.)	Day	Date
545	1	1 Jan 1995
595	32	
669	60	
758	91	
839	121	
901	152	
915	172	21 Jun 1995
912	182	
867	213	
784	244	
700	274	
616	305	
555	335	
540	356	22 Dec 1995
544	366	1 Jan 1996
595	397	
671	426	
760	457	
840	487	
902	518	
915	538	21 Jun 1996
912	548	
865	579	
782	610	
698	640	
614	671	
554	701	
540	721	21 Dec 1996
545	732	1 Jan 1997
597	763	
671	791	
760	822	
839	852	
902	883	
915	903	21 Jun 1997
912	913	
865	944	
783	975	
699	1005	
615	1036	
554	1066	
540	1086	21 Dec 1997
545	1097	1 Jan 1998

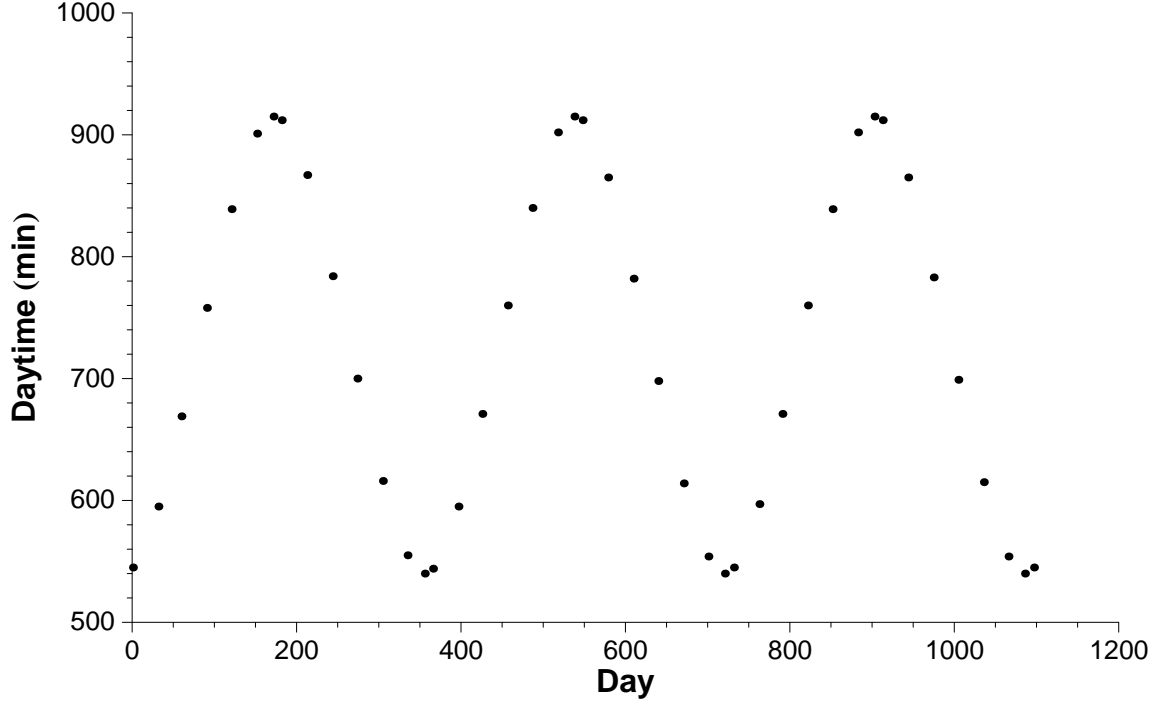


Figure 1. Raw Daytime Data

$$\text{Total-sum-of-squares} \equiv \text{TSS} = \sum_{i=1}^N (y_i - \bar{y})^2 \quad 1.$$

where \bar{y} is the average value of y .

TSS is a positive number summarizing how much the y -values vary about their average (*mean*). The fact that each x (day) is paired with a unique y (daytime) is completely ignored. By discounting this important relationship, even a very large dataset may be characterized by a single number, i.e., by a *statistic*. The average amount of TSS attributable to each point (Equation 2) is known as the *variance* of the variable, y . Lastly, the square-root of the variance is the *standard deviation*, another important statistic.

$$\text{Variance of } y \equiv \text{Var}(y) = \frac{1}{N} \sum_{i=1}^N (y_i - \bar{y})^2 \quad 2.$$

In Figure 1, the y -values come from a continuum but the x -values do not. More often, x is a continuous variable, sampled at points chosen by the observer. For this reason, it is called the *independent variable*. The *dependent variable*, y , describes measurements made at chosen values of x and is almost always inaccurate to some degree. Since the x -values are selected in advance by the observer, they are most often assumed to be known exactly. Obviously, this cannot be true if x is a real number but, usually, uncertainties in x are negligible compared to uncertainties in y . When this is not true, some very subtle complications arise.

Table 2 lists data from a recent astrophysics experiment, with measurement uncertainties explicitly recorded.² These data come from observations, made in 1996-1997, of comet Hale-Bopp as it approached the Sun [RAU97]. Here, the independent variable is the distance of the comet from the Sun. The unit is AU, the average distance (approximately) of the Earth from the Sun. The dependent variable is the rate of production of cyanide, CN, a decomposition product of hydrogen cyanide, HCN, with units of molecules per second divided by 10^{25} . Thus, even when Hale-Bopp was well beyond the orbit of Jupiter (5.2 AU), it was producing cyanide at a rate of $(6 \pm 3) \times 10^{25}$ molecules per second, that is, nearly 2.6 kg/s.

Table 2. Rate of Production of CN in Comet Hale-Bopp

Rate (molecules per second)/ 10^{25}	Distance from Sun (AU)	Uncertainty in Rate (molecules per second)/ 10^{25}
130	2.9	40
190	3.1	70
90	3.3	20
60	4.0	20
20	4.6	10
11	5.0	6
6	6.8	3

In this example, the uncertainties in the measurements (Table 2, column 3) are a significant fraction of the observations themselves. Establishing the value of the uncertainty for each data point and assessing the net effect of uncertainties are crucial steps in any analysis. Had Kepler's data been as poor as the data available to Copernicus, his name would be known only to historians.

The data of Table 2 are presented graphically in Figure 2. For each point, the length of the *error bar* indicates the uncertainty³ in y . These uncertainties vary considerably and with some regularity. Here, as often happens with observations made by electronic instruments which measure a physical quantity proportional to the target variable, the uncertainty in an observation tends to increase with the magnitude of the observed value.

Qualitatively, these data suggest the hypothesis that the comet produced more and more CN as it got closer to the Sun. This would make sense since all chemical reactions go faster as the temperature increases. On the other hand, the observed rate at 2.9 AU seems too small. Did the comet simply start running out of HCN? How likely is it that the rate at 3.1 AU was really bigger than the rate at 2.9 AU? Are these values correct? Are the uncertainties correct? If the uncertainties are correct, what does this say about the validity of the hypothesis? All of these are legitimate questions.

² see file *Examples:Hale_Bopp.CN.in*

³ In spite of its name, this bar does not indicate error. If it did, the error could be readily removed.

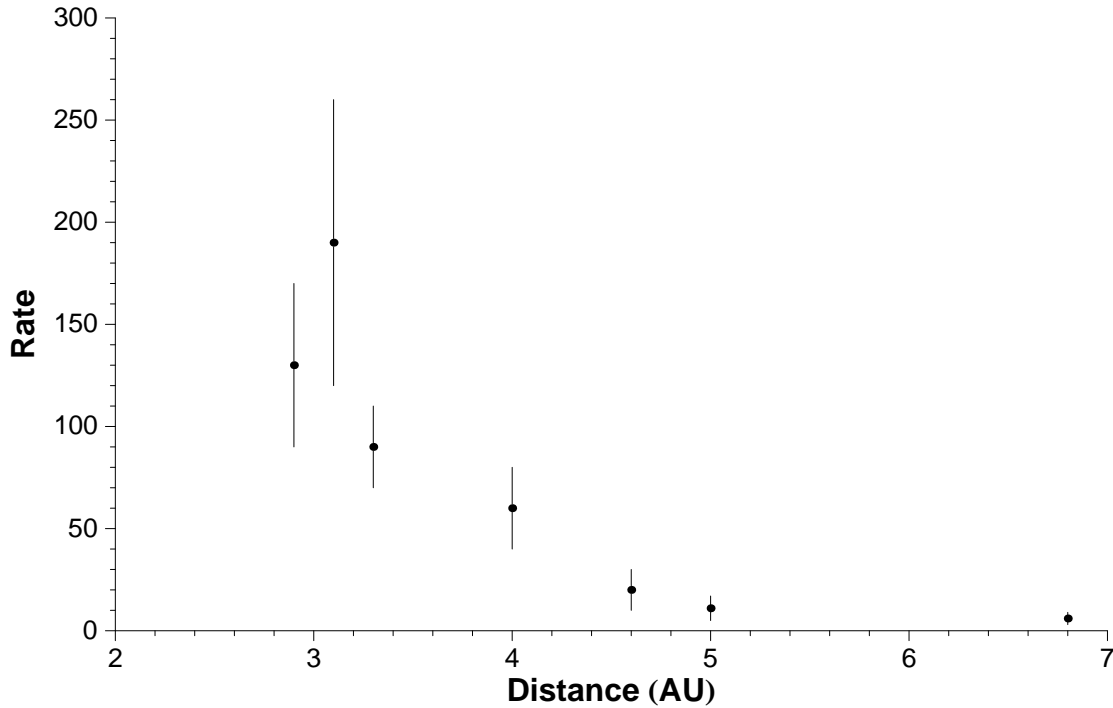


Figure 2. Hale-Bopp CN Data

Finally, consider the very “unscientific” data shown in Figure 3. This figure is a plot of the highest major-league baseball batting averages in the United States, for the years 1901-1997, as a function of time.⁴

A player’s batting average is the fraction of his “official at-bats” in which he hit safely. Thus, it varies continuously from zero to one. It is fairly clear that there is a large difference between these data and those shown in Figure 1. The latter look like something from a math textbook. One gets the feeling that a daytime value could be predicted rather well from the values of its two nearest neighbors. There is no such feeling regarding the data in Figure 3. At best, it might be said that batting champions did better before World War II than afterwards. However, this is not an impressive conclusion given nearly a hundred data points.

Considering the data in Figure 3, there can be little doubt that maximum batting average is not really a function of time. Indeed, it is not a function of anything. It is a *random variable* and its values are called *random variates*, a term signifying no pretense whatever that any of these values are **individually** predictable.⁵ When discussing random (stochastic) variables, the terms “independent” and “dependent” have no relevance and are not used, nor are scatter plots such as Figure 3 ever drawn except to illustrate that they are almost meaningless.

⁴ see files *Examples:BattingAvgEq.in* and *Examples:BattingAvg.in* [FAM98]

⁵ The qualification is crucial; it makes random data comprehensible.

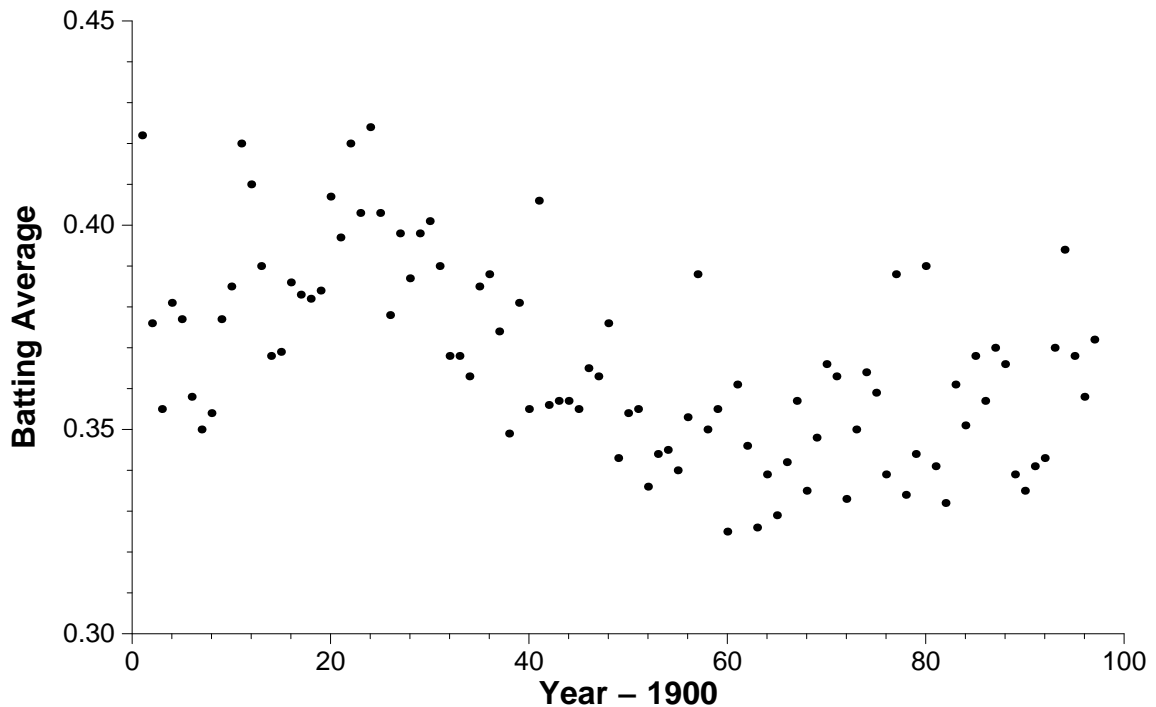


Figure 3. Annual Best Baseball Batting Average

Variables appear random for one of two reasons. Either they are inherently unpredictable, in principle, or they simply appear so to an observer who happens to be missing some vital information that would render them *deterministic* (non-random). Although deterministic processes are understandably of primary interest, random variables are actually the more common simply because that is the nature of the Universe. In fact, as the next section will describe in detail, understanding random variables is an essential prerequisite for understanding any real dataset.

Making sense of randomness is not the paradox it seems. Some of the metrics that apply to deterministic variables apply equally well to random variables. For instance, the mean and variance (or standard deviation) of these batting averages could be computed just as easily as with the daytime values in Example 1. A computer would not care where the numbers came from. No surprise then that statistical methodology may be profitably applied in both cases. Even random data have a story to tell.

Which brings us back to the point of this discussion. We have data; we seek insight and understanding. How do we go from one to the other? What's the connection?

The answer to this question was Kepler's most important discovery. Data are connected to understanding by a *model*. When the data are quantitative, the model is a mathematical model, in which case, not only does the form of the model lead directly to understanding but one may query the model to gain further information and insight.

But, first, one must have a model.

FROM DATA TO MODEL

Marshall McLuhan is widely recognized for his oft-quoted aphorism, “The medium is the message.” It is certainly true in the present context. The model is the medium between data and understanding and it is, as well, the message of this tutorial. The current section presents an outline of fundamental concepts essential for initiating the first step, from data to model. The methodology for implementing these concepts is described in later sections. The second step, from model to understanding, will be left as an exercise for the reader.

As can be seen, through the examples above, data are not transparent. They do not reveal their secrets to casual observers. Yet, when acquired with due diligence, they contain useful information. The operative word is “contain.” Data and information are not equivalent even though, in colloquial speech, they are treated as such.

Why not? What is there in a dataset that is not information? The answer is *error*.

$$\text{Data} = \text{Information} + \text{Error}$$

Were it not for error, every observation would be accurate, every experiment a paragon of perfection, and nearly every high school Science Fair project a fast track to the Nobel Prize. Alas, error exists. It exists not just in every dataset but in every data point in every dataset. It contaminates everything it touches and it touches everything. To make any progress, it must be identified and filtered out.

A good model does this very nicely. In its simplest, most superficial aspect, a model is a filter designed to separate data into these two components. A mathematical model is a statistical filter that not only attempts the separation but also quantifies its success in that effort.

Two Ways to Construct a Model

To construct a model, it is necessary to proceed from the known to the unknown or, at the very least, from the better known to the less well known. There are two approaches. The choice depends upon whether it is the information or the error that is better known, bearing in mind that “known” and “assumed” are not synonyms. In the first case, the model is designed to utilize the known properties of the embedded information to extract the latter and leave the error behind. This approach is commonly employed with stochastic data. Alternatively, if the error is the better known, the model is designed to operate on the error, filtering it out and leaving the information behind. This approach is nearly universal with deterministic data. In either case, the separation will be imperfect and the information still a bit erroneous.

The notion of deterministic information is commonplace and requires no further elaboration but what about the putative “stochastic information” contained in a dataset of random variables? Is it real? Likewise, can anything useful be said about error? Do random variables and error really have “properties” that one can understand? In other words, are they comprehensible?

They are indeed. The remainder of this section elucidates some properties of i) stochastic information and ii) error. Such properties are quantitative, leading directly to the identification of optimization criteria which are crucial to any modeling process.

Stochastic Information

Stochastic. Information. The juxtaposition seems almost oxymoronic. If something is stochastic (random), does that not imply the absence of information? Can accurate conclusions really be induced from random variables?

Well, yes, as a matter of fact. That a variable is stochastic signifies only that its next value is unpredictable, no matter how much one might know about its history. The data shown in Figure 3 are stochastic in this sense. If you knew the maximum batting average for every year except 1950, that still would not be enough information to tell you the correct value for the missing year. In fact, no amount of ancillary data would be sufficient. This quality is, for all practical purposes, the quintessence of *randomness*.

Yet, we have a very practical purpose here. We want to assert something definitive about stochastic information. We want to construct models for randomness.

Such models are feasible. Although future values for a random variable are unpredictable in principle, their **collective behavior** is not. Were it missing from Figure 3, the maximum batting average for 1950 could not be computed using any algorithm. However, suggested values could be considered and assessed with respect to their **probability**. For instance, one could state with confidence that the missing value is probably not less than 0.200 or greater than 0.500. One could be quite definite about the improbability of an infinite number of candidate values because any large collection of random variates will exhibit consistency in spite of the randomness of individual variates. It is a matter of experience that the Universe is consistent even when it is random.

A mathematical model for randomness is expressed as a *probability density function* (PDF). The easiest way to understand a PDF is to consider the process of computing a weighted average. For example, what is the *expectation* (average value) resulting from tossing a pair of ordinary dice if prime totals are discounted? There are six possible outcomes but they are not equally probable. Therefore, to get the correct answer, one must calculate the **weighted** average instead of just adding up the six values and dividing the total by six. This requires a set of weights, as shown in Table 3.

Table 3. Weights for Non-prime Totals for Two Dice

Value	Weight
4	3/21
6	5/21
8	5/21
9	4/21
10	3/21
12	1/21

The random total, y , has an expected value, denoted $\langle y \rangle$ or \bar{y} , given by Equation 3.

$$\text{Expectation of } y \equiv \langle y \rangle = \sum_{i=1}^6 w_i y_i \quad 3.$$

The true expectation, 7.62, is quite different from the unweighted average, 8.17.

The set of weights in Table 3 constitutes a *discrete* PDF, meaning that there are a **countable** number of possible values, each with its own weight (density). Given a discrete PDF, $f(y)$, for the random variable, y , any arbitrary function of y , $g(y)$, will have an expectation computed as shown in Equation 4. Equation 3 is just a special case of Equation 4.

$$\langle g(y) \rangle = \sum_{\text{all } i} f(y_i) g(y_i) \Delta y \quad 4.$$

where Δy is equal to the *binwidth*, defined below in Example S1.

In the *continuous* case, neighboring values are separated by the infinitesimal binwidth, dy . Nevertheless, a continuous PDF has exactly the same meaning as a discrete PDF and, thus, the continuous analogue of Equation 4 is Equation 5.

$$\langle g(y) \rangle = \int_{-\infty}^{\infty} f(y) g(y) dy \quad 5.$$

For example, suppose $f(y) = e^{-y}$ and $y \geq 0$. Then, the expectation of $g(y) = \sqrt{y}$ is

$$\langle \sqrt{y} \rangle = \int_0^{\infty} e^{-y} \sqrt{y} dy = \frac{\sqrt{\pi}}{2} \quad 6.$$

The product $(f(y) \Delta y)$ or $(f(y) dy)$ equals the probability of a given y . Probability is a dimensionless quantity. That is, it is a pure number with no units. Consequently, the units of a PDF must always be the reciprocal of the units, if any, of y .⁶ It is for this reason that the function $f(y)$ is referred to as a “density” function.

Example S1—Batting Averages

The dataset shown in Figure 3 will be the first example. The collective behavior of these data can be summarized numerically and graphically. For instance, their mean (expectation) is 0.367 and their variance is 0.0005596 [st. dev. = 0.0237]. Any statistics textbook will list several additional metrics that could be used to characterize this set of 97 values. For a graphical depiction, the device most commonly employed is the *histogram*, constructed by grouping the variates into numerical intervals of equal width and plotting these *bins* against their *frequencies*, the number of variates in the respective bins. These frequencies may be converted to probabilities (*normalized*) by dividing each by the sample size and then converted to probability densities by further dividing by the binwidth (here, 0.011). When all of this is done with the batting-average data, we get the histogram shown in Figure 4 (gray boxes).

⁶ a useful check for complicated density formulas (see Appendix A)

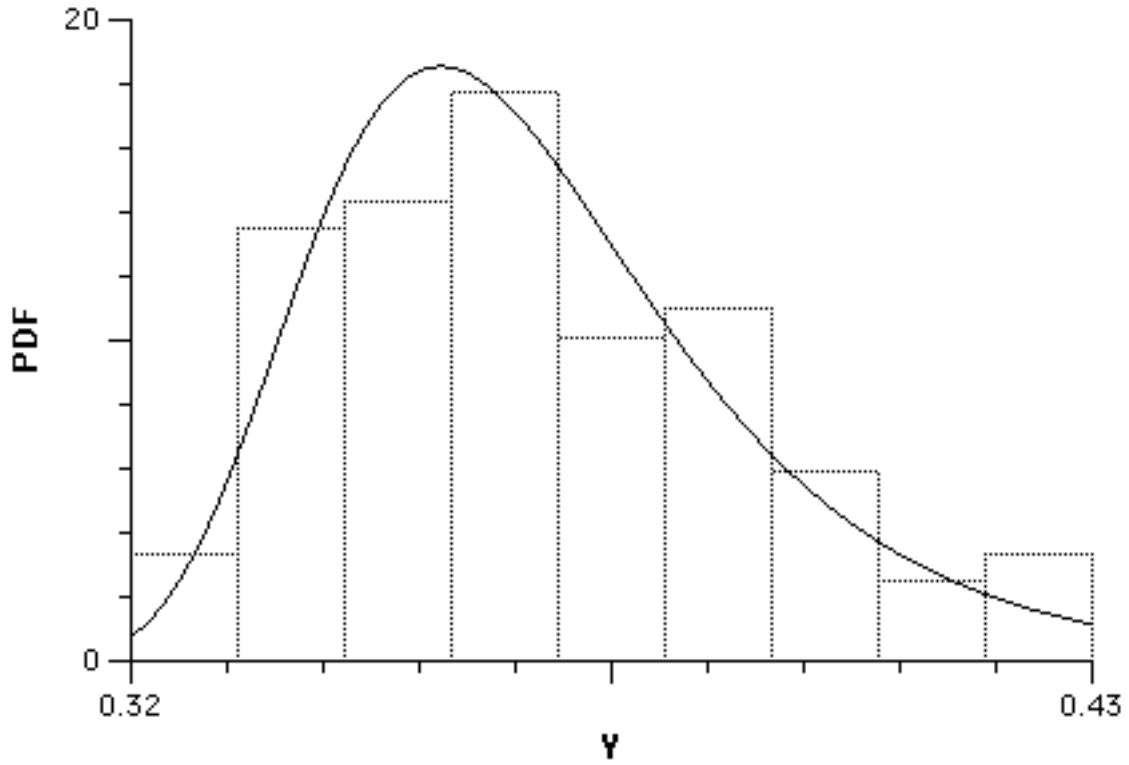


Figure 4. Batting-Average Data (Histogram and PDF)

The overall shape of a histogram is intended to describe how realizable values of a random variable are scattered across the infinite landscape of conceivable locations. The way it works is that the histogram is constructed to have a total **area** = 1. Therefore, the probability that any random variate falls in a given bin is equal to the area of the corresponding rectangle. This scattering of values/probabilities is referred to as the *distribution* of the random variable. Note that the Y-axis measures probability density, not probability. Given the computational process just outlined, and recalling Equations 4 and 5, measurements along this axis have units of 1/u, where u is the unit of the random variable (e.g., inches, grams, etc.)

A continuous density function for these data is also shown in Figure 4 (solid line). The corresponding analytical expression, given in Equation 7, is called the Gumbel distribution.⁷ Like the histogram, the total area under this curve, including the tails (not shown), equals 1.

$$\text{PDF} = \frac{1}{B} \exp\left(\frac{A-y}{B}\right) \exp\left(-\exp\left(\frac{A-y}{B}\right)\right) \quad 7.$$

where, for this example, A = 0.3555 and B = 0.01984.

⁷ see pg. A-49

We see that Equation 7 does have the proper units. Parameter A has the same units as the random variable, y . Since any exponent is necessarily dimensionless, parameter B must have the same units as well. Hence, the entire expression has units of $1/u$. Of course, these particular data happen to be dimensionless already, but variates described by a PDF usually do have units of some sort.

The Gumbel distribution is one example of a continuous distribution. [There are many others described in Appendix A.] If $f(y)$ is the PDF, the probability of any value of y would equal $f(y)*dy$ which, since dy is vanishingly small, is zero. Indeed, intuition demands that the probability of randomly picking any given value from a continuum must be zero. However, the probability of picking a value in a finite range $[a, b]$ may be greater than zero (Equation 8).

$$\text{Prob} (a \leq y \leq b) = \int_a^b f(y) dy \quad 8.$$

The integral (or sum) of a density function, from minus infinity to some chosen value, y , is referred to as the *cumulative distribution function* (CDF), usually denoted $F(y)$.⁸ For this example, it is plotted in Figure 5. In this figure, the solid line is the theoretical CDF and the gray line the empirical CDF. The CDF-axis gives the probability that a randomly selected variate will be less than or equal to b (Equation 9).

$$\text{Prob} (y \leq b) = F(b) = \int_{-\infty}^b f(y) dy \quad 9.$$

Some of the reasons for choosing the Gumbel distribution for these data are discussed in its entry in Appendix A. For now, it is sufficient to appreciate that any sort of “theoretical” expression would be considered relevant. This supports the conclusion that useful statements can be made about stochastic data. However, it does not explain where these parameter values came from. They are another matter entirely.

One of the reasons for using the values of A and B given above is that they produce a curve that encloses an area with approximately the same shape and size as the empirical histogram. There are much better reasons but their explication requires some preliminary concepts.

The first is the *likelihood* of a dataset. The term means just what it says. It indicates how “likely” this set of observations would be had they been selected randomly from the given PDF. **If the variates are independent**, the likelihood is also a measure of their *joint probability*.⁹ For any PDF, $f(y)$, the likelihood of a sample of N variates is defined by Equation 10.

$$\text{Likelihood} \equiv \prod_{k=1}^N f(y_k) \quad 10.$$

⁸ The term “distribution” is sometimes taken to mean the cumulative distribution.

⁹ Two random quantities, x and xx , are independent if and only if their joint (simultaneous) probability, $\text{Prob}(x \text{ AND } xx)$, is **always** equal to $\text{Prob}(x)*\text{Prob}(xx)$.

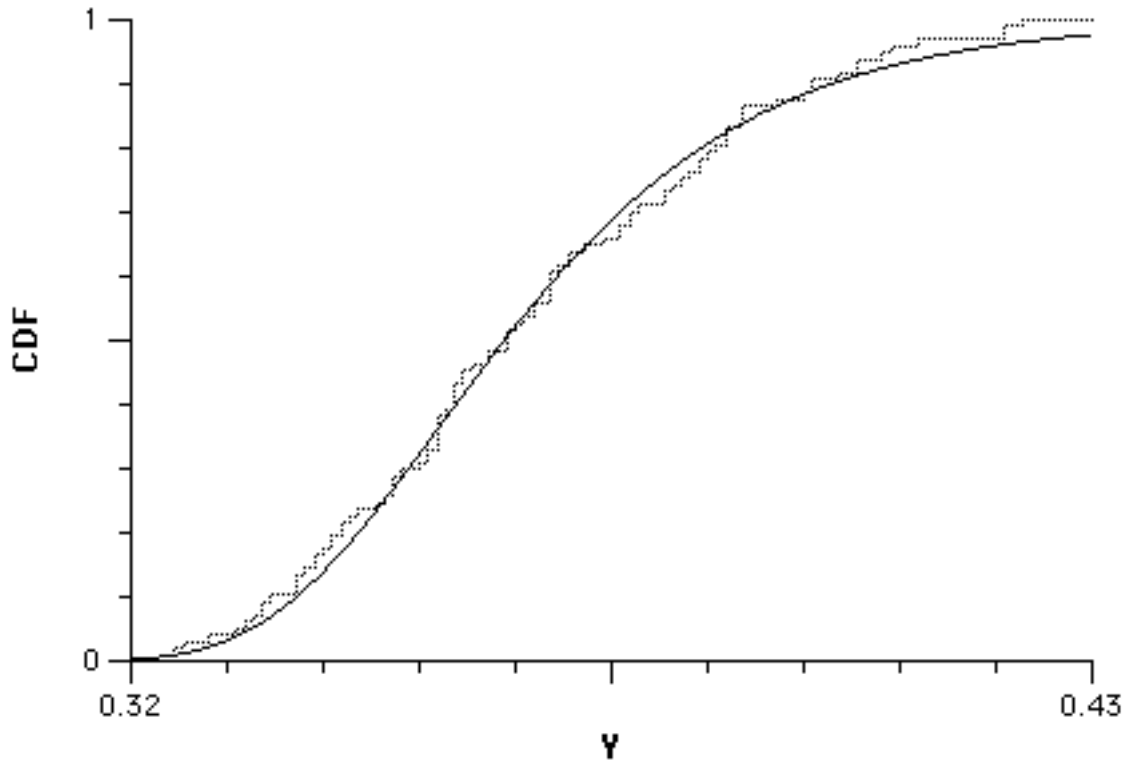


Figure 5. Batting-Average Data (CDF)

The second concept is that of *maximum likelihood* (ML). This notion is also very intuitive. Given a form for the density function, the ML parameters are those parameters which make the likelihood as large as possible. In other words, any other set of parameters would make the observed dataset less likely to have been observed. Since the dataset was observed, the ML parameters are generally considered to be optimal.¹⁰ However, this does not preclude the use of some alternate criterion should circumstances warrant.

The values given above for A and B are the ML parameters. This is an excellent reason for choosing them to represent the data and it is no coincidence that the theoretical curve in Figure 4 matches the histogram so well. Had there been a thousand data points, with proportionately narrower histogram bins, the match would have been even better.

Thus, a Gumbel distribution, with these ML parameters, is a model for the data shown in Figure 3. It summarizes and characterizes the stochastic information contained in these data. Not only does it describe the probability density of the data, but one could query the model. One could ask the same questions of the model that could be asked of the data. For example, “What are the average and standard deviation of these data?” or “What is the probability that next year’s batting champion will have a batting average within five percent of the historical

¹⁰ Their uniqueness is usually taken for granted.

mean?” The answers to such questions may be computed from the model without looking at the data at all.

For instance, the last question is answered in Equation 11.⁷

$$\text{Probability} = \int_{0.95 * \text{mean}}^{1.05 * \text{mean}} f(y) dy = F(0.385) - F(0.349) = 0.55 \quad 11.$$

As a check, note that Equation 11 predicts that 53 of the values in Figure 3 are within the given range. The data show 52 in this range.

One could even ask questions of the model that one could not ask of the data. For instance, “What is the probability that next year’s batting champion will have a batting average greater than the maximum in the dataset (0.424)?” Obviously, there is no such number in the historical data and that suggests an answer of zero. However, this is clearly incorrect. Sooner or later, someone is bound to beat 0.424 provided that major-league baseball continues to be played.

It is easy to pose a theoretical question like this to a model and, **if the model is good**, to obtain a correct answer. This model (Equation 7) gives an answer of three percent, suggesting that we are already overdue for such a feat.

Whether or not a model is good (valid) is a question yet to be addressed.

Example S2—Rolling Dice

The Gumbel distribution was chosen to model the batting-average data because it is known to be appropriate, in many cases, for samples of continuous-distribution maxima (so-called *record values*). However, there is no *a priori* guarantee that it is valid for these particular data. Occasionally, there is enough known about some type of stochastic information that theory alone is sufficient to specify one distribution to the exclusion of all others and, perhaps, even specify the parameters as well. The second stochastic example illustrates this situation *via* the following experiment.

Step 1

Think of a number, a whole number, from one to six.

Step 2

Roll a standard, cubical die repeatedly until your chosen number appears three times.

Step 3

Record the number of rolls, N_r , required.

After only a few semesters of advanced mathematics courses, it would be relatively easy to prove that the random variable, N_r , is $\sim \text{NegativeBinomial}(1/6, 3)$.¹¹ This assertion could be tested by performing the experiment a very large number of times and comparing the theoretical distribution to the empirical data.

¹¹ see pg. A-81; the \sim is read “(is) distributed as”

Carrying out this exercise by hand would be a bit tedious. Fortunately, it is a simple matter to *simulate* the experiment on a computer. Such simulations are very common. Simulated results for a 1,000-fold repetition of this experiment are shown in Figure 6.¹²

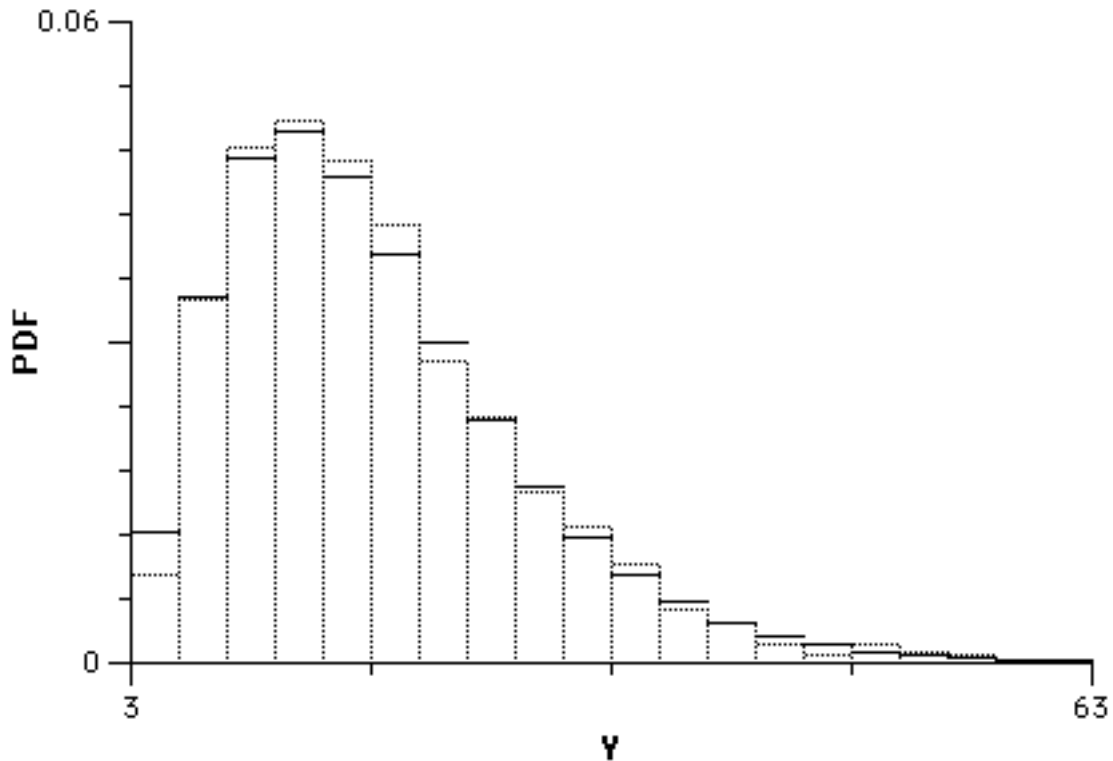


Figure 6. Rolls3 Data (Histogram and PDF)

The NegativeBinomial model shown in Figure 6 (solid lines) is another example of a discrete PDF. Typically, this means that the random variable can take on only integer values. That is the case here since N_r is a count of the number of rolls required.

In computing the model shown, the first parameter was not fixed at its theoretical value, $1/6$. Instead, it was estimated from the data. The ML estimate = 0.1691, very close to the theoretical 0.1667. It would have been even closer had the sample size been much larger than 1,000. Likewise, the mean predicted by this model = 17.74, very close to the theoretical mean, 18. The second parameter, 3, was assumed given although it, too, could have been considered unknown and estimated from the data.

The observed data (gray histogram) may be compared to the estimated, ML model using the *Chi-square* statistic, defined in Equation 12.

¹² see file *Examples:Rolls3.in*

$$\text{Chi-square} \equiv \chi^2 = \sum_{\text{all } k} \frac{(o_k - e_k)^2}{e_k} \quad 12.$$

where k includes all possible values of the random variable and where o and e are the observed and expected frequencies, respectively.

For this example, Chi-square = 65.235. Again, the question of whether this amount of discrepancy between theory and experiment is “acceptable” is a matter to be discussed below.

Example S3—Normality

We would be remiss in our illustrations of stochastic information were we to omit the most famous of all stochastic models, the Normal (Gaussian) distribution. The Normal distribution is a continuous distribution and it arises naturally as a result of the *Central Limit Theorem*. In plain English, this theorem states that, provided they exist, averages (means) of any random variable or combination of random variables tend to be $\sim \text{Normal}(A, B)$, where A is the sample mean and B is the unbiased estimate of the population standard deviation. As usual, with random data, the phrase “tend to” indicates that the validity of this Gaussian model for random means increases as the sample size increases.

In Example S2, a thousand experiments were tabulated and individual outcomes recorded. A sample size of 1,000 is certainly large enough to illustrate the Central Limit Theorem. Our final example will, therefore, be a 1,000-fold replication of Example S2, with one alteration. Instead of recording 1,000 values of N_r for each replicate, only their average, N_{avg} , will be recorded. This new experiment will thus provide 1,000 averages¹³ which, according to the Central Limit Theorem, should be normally distributed.

As before, the experiment was carried out in simulation. The observed results and the ML model are shown in Figure 7. The estimated values for A and B are 17.991 and 0.2926, respectively. With a Gaussian model, the ML parameter estimates may be computed directly from the data so one cannot ask how well they match the observed values.

A somewhat better test of the model would be to pick a statistic not formally included in the analytical form of the distribution. One such statistic is the interquartile range, that is, the range included by the middle half of the sorted data. In this example, the model predicts a range of [17.79, 18.19]¹⁴ while the data show a range of [17.81, 18.19].

The ideal test would be one that was independent of the exact form of the density function. For continuous distributions, the most common such test is the *Kolmogorov-Smirnov* (K-S) statistic. The K-S statistic is computed by comparing the empirical CDF to the theoretical CDF. For example S3, these two are shown in Figure 8. The K-S statistic is simply the magnitude (absolute value) of the maximum discrepancy between the two, measured along the CDF-axis. Here, the K-S statistic = 0.0220.

¹³ see file *Examples:Rolls3avg.in*

¹⁴ see pg. A-85

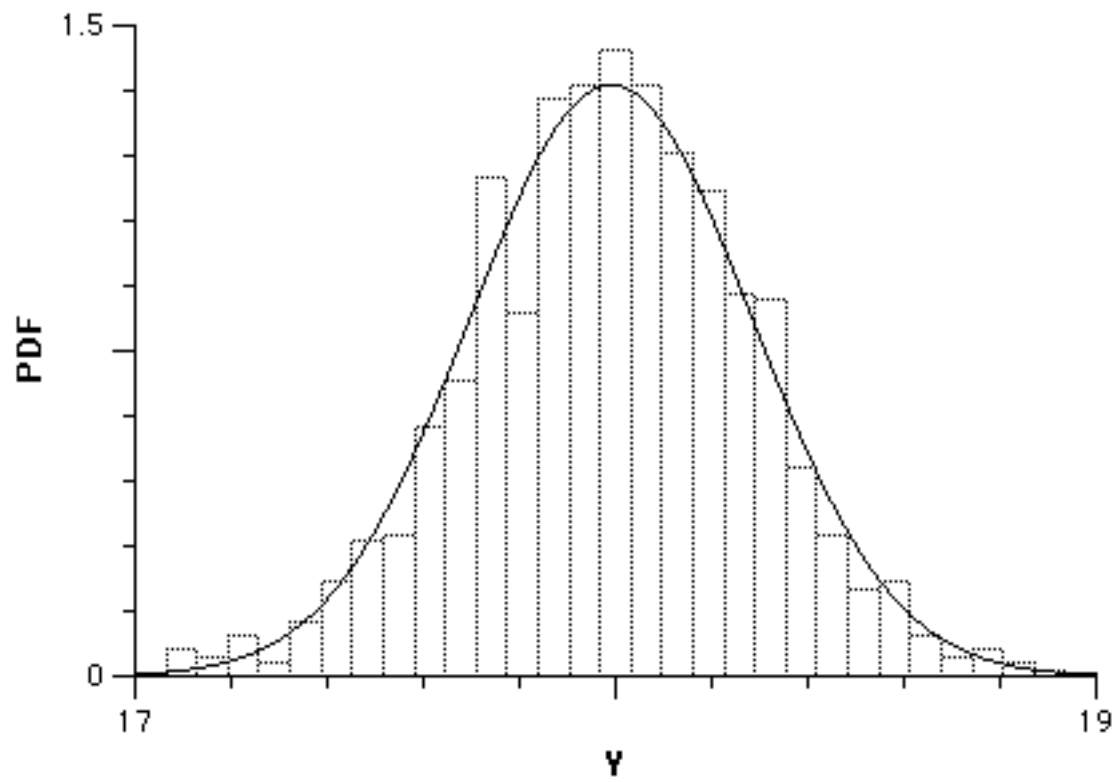


Figure 7. Rolls3avg Data (Histogram and PDF)

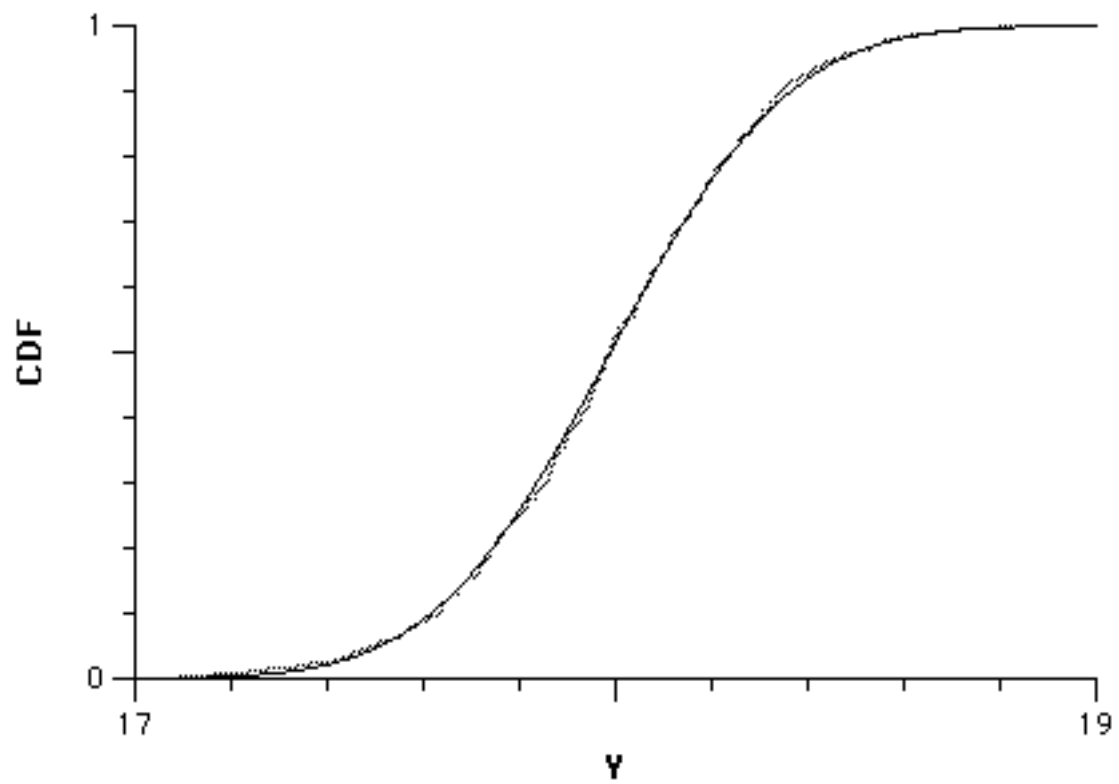


Figure 8. Rolls3avg Data (CDF)

Once again, the question naturally arises as to whether this result is acceptable and, once again, we defer the question until later.

Stochastic Information—Summary

In examples S1 to S3, we have endeavored to show that the term “stochastic information” is not the contradiction it would appear to be. It should now be clear that random variables are collectively, if not individually, predictable to a useful extent and that much can be said, with accuracy, about such variables.

Since much can be said, it follows that it should be possible to use this information to filter out some of the error found in stochastic data. Just because variates are random, it does not mean that they are errorless. The data of Example S1, for instance, appeared to be described by a Gumbel distribution. However, as with all stochastic datasets, there were discrepancies between model and data. How much of this was due to inherent variation and how much was due to error was not discussed. It will be.

Finally, we have described three metrics: the ML criterion, the Chi-square statistic, and the K-S statistic. These three are valid and useful concepts for the description of stochastic data and it will soon become evident that all may be utilized as optimization criteria as well.

First, however, we discuss deterministic information and the kinds of error commonly associated with it.

Deterministic Information and Error

If “stochastic information” seems an oxymoron, “deterministic information” would, for similar reasons, appear to be a tautology. Neither is true and, in this discussion, the latter term will refer to any information gleaned from non-stochastic sources, typically experimentation. Since these sources are prone to error, it is essential to characterize that error in order to attempt to compensate for it. This section will focus on the kinds of errors found in deterministic data.

The amount of error in a dataset, or even a data point, is generally unknown. The reason, of course, is that error is largely random. We have just seen, however, that randomness does not imply total ignorance. To the contrary, modeling of deterministic data usually assumes that something valid is known about the error component of the data, providing a handle with which to filter it out. In fact, whenever data of a given type are investigated over an extended period of time, it is not uncommon for associated errors to become as well characterized as the information itself.

However, for the purposes of this tutorial, our goals are much less specific. We shall limit our illustrations to a few kinds of error and say something, as well, about error in general. As in the last section, we seek quantitative metrics that may be employed as optimization criteria.

Components of Error

Empirical datasets, almost without exception, are contaminated with error. Even the data shown in Table 1, which few would dispute, are recorded with limited precision. Therefore, they are subject, at the very least, to *quantization* error. A dataset without error might be imagined but hardly ever demonstrated.

Not only does a dataset, as a whole, contain an unknown amount of error but every data point in it contains some unknown fraction of that total error. Consequently, the amount of information in a dataset is unknown as well. Each data point makes some contribution to the total amount of information but there is no way to tell how much just by looking. Information and error have the same units and may be physically indistinguishable.

Error comprises some combination of *bias* and *noise*. A bias is a systematic deviation from the truth due either to a deterministic deviation, consistently applied, or to a random deviation with a non-zero mean. The former is seen, for instance, when your bathroom scale is not properly zeroed in. In this case, it will report your apparent weight as too high or too low, depending on the sign of the bias in the zero-point. The latter type of bias is exemplified in the case of an astronomer who records the image of a distant galaxy by counting the photons that impinge on the pixels of a light-sensitive array. Inevitably, each pixel is subject to a counting error. Such errors are typically random variables $\sim \text{Poisson}(A)$ and have a mean, $A > 0$.¹⁵

Noise refers either to random, zero-mean errors or to the residual portion of random, biased errors once the bias has been removed. For example, undergraduate chemistry students, performing their first quantitative, organic analysis usually get experimental results exhibiting lots of errors. These errors are largely random but often have a significant negative

¹⁵ see pg. A-101

bias because, in trying to isolate the target substance, some is lost. Quantifying the remainder is a process usually subject to zero-mean, random errors, i.e., noise.

Most often, total error represents the combined effect of several sources of different kinds of error, including biases which can sometimes be identified and removed, as well as noise. To the degree that error is random, only its average effects can be characterized and there is, of course, no guarantee that a given sample will be average.

As before, we shall eschew theoretical discussions in favor of concrete examples. There is ample literature available to anyone wishing to pursue this subject in depth. The examples below all relate to deterministic data.

Example D1—Quantization Error

The daytimes in Table 1 provide an illustration of one kind of error and, at the same time, of the utility of the variance.

One of the most useful properties of the variance statistic is its *additivity*. Whenever any given variance, V , is due to the combined effect of k **independent** sources of variation, V will equal the sum of the individual variances of these k sources. [Note the recursive nature of this property.]

The variance of the daytimes listed in Table 1 = $TSS/N = 836,510/43 = 19,453.72 \text{ min}^2$. As noted above, a portion of this variance derives from the fact that these values, having been rounded to the nearest minute, have limited precision. This round-off error has nothing to do with the mechanism responsible for the secular variation in daytime. Hence, the variance due to rounding (quantization) is independent of all remaining variance, so it can be factored out.

It is easy to demonstrate that quantization error is a random variable $\sim \text{Uniform}[-B, B]$, where B is one-half the unit of quantization.¹⁶ As such, the variance due to this single source of error, Q , is given by Equation 13. For these data, the unit of quantization is one minute. Therefore, $Q = 0.08 \text{ min}^2$, only 0.0004 percent of the total variance of the data.

$$\text{Quantization variance} \equiv Q = \frac{(\text{unit of quantization})^2}{12} \quad 13.$$

Continually citing squared quantities (with squared units) is inconvenient and the standard deviation is quoted more often than is the variance. Unfortunately, standard deviations are **not** additive. In this example, the “standard deviation of quantization error” is 0.29 min (about 17 seconds). Thus, although this quantization error is negligible compared to the overall variation of the data, it is clearly not negligible when compared to the recorded precision.

If quantization were the only source of error for this dataset, the deterministic information present would constitute 99.9996 percent of the observed variance. It would then be left to the chosen model to “explain” this information. Note that a deterministic model is not expected to

¹⁶ see pg. A-113

explain error. That task is delegated to a separate error model, implicitly specified as part of the optimization procedure.

Finally, the variance statistic is evidently useful not just for computational purposes but, as seen here, as a measure of information. In other words, were the errorless component of the total variance equal to zero, there would be nothing to explain. In Example D2, we utilize this aspect of the variance to develop a quantitative metric for deterministic information explained.

Example D2—Kepler’s First Law

In a deterministic model, the value of the independent variable is supposed to be sufficient to predict the value of the dependent variable to an accuracy matching that of the input. If the model is important enough, it is often referred to as a *law*. The implied reverence generally signifies that something of unusual significance has been discovered about the real world and summarized in the model so designated. Today, in the physical sciences, laws are particularly precise, as are their parameters. In fields related to physics, for instance, it is not uncommon to find model parameters accurate to more than nine significant figures (i.e., one part per billion). This mathematical approach to understanding we owe, in large part, to the efforts of Johannes Kepler (1571-1630) and Galileo Galilei (1564-1642).

This example was presented by Kepler as an illustration of his First Law, arguably the first correct mathematical model correctly devised. The law states that each planet revolves about the Sun in an ellipse with the Sun at one focus.

Figure 9 depicts an ellipse, together with some of its more important features. Like any ellipse, its size and shape may be completely specified by two numbers. Usually, one of these is the length of the *semi-major axis*, OP, and the other is the *eccentricity*, which is the ratio of two lengths, OS/OP. The ellipse in the figure has an eccentricity of 0.5 so one focus, S, is halfway between the center, O, and the circumference at P; the other focus, not shown, is the mirror image of S through the center.

Kepler spent much of his life investigating the orbit of Mars. This orbit has an eccentricity of only 0.09, so it is much more circular than the ellipse shown in Figure 9 as, indeed, are the orbits of all planets in the Solar System, except Mercury and Pluto. The point P represents the point closest to the Sun, S, and is called *perihelion*; the point farthest from the Sun, A, is called *aphelion*. A chord, SG, drawn through a focus and parallel to the *semi-minor axis*, OR, is called the *semi-latus rectum*. The latter, plus the eccentricity, are the only parameters required when an ellipse is described in polar coordinates (Equation 14).

$$\text{radius} = \frac{A}{1 + B \cos(v)} \quad 14.$$

where the origin is at S, from which the radius (e.g., SM) is measured, and where the angle, v , measured from perihelion in the direction of motion, is called the *true anomaly*.

In Equation 14, parameter A is the length of the semi-latus rectum and B is the eccentricity. This equation is actually a general equation describing all of the conic sections—circle, ellipse, parabola, or hyperbola, depending upon the eccentricity (see Table 4). The independent

variable is true anomaly, from which the dependent variable, the heliocentric radius, may be computed/predicted **if the model is valid**.

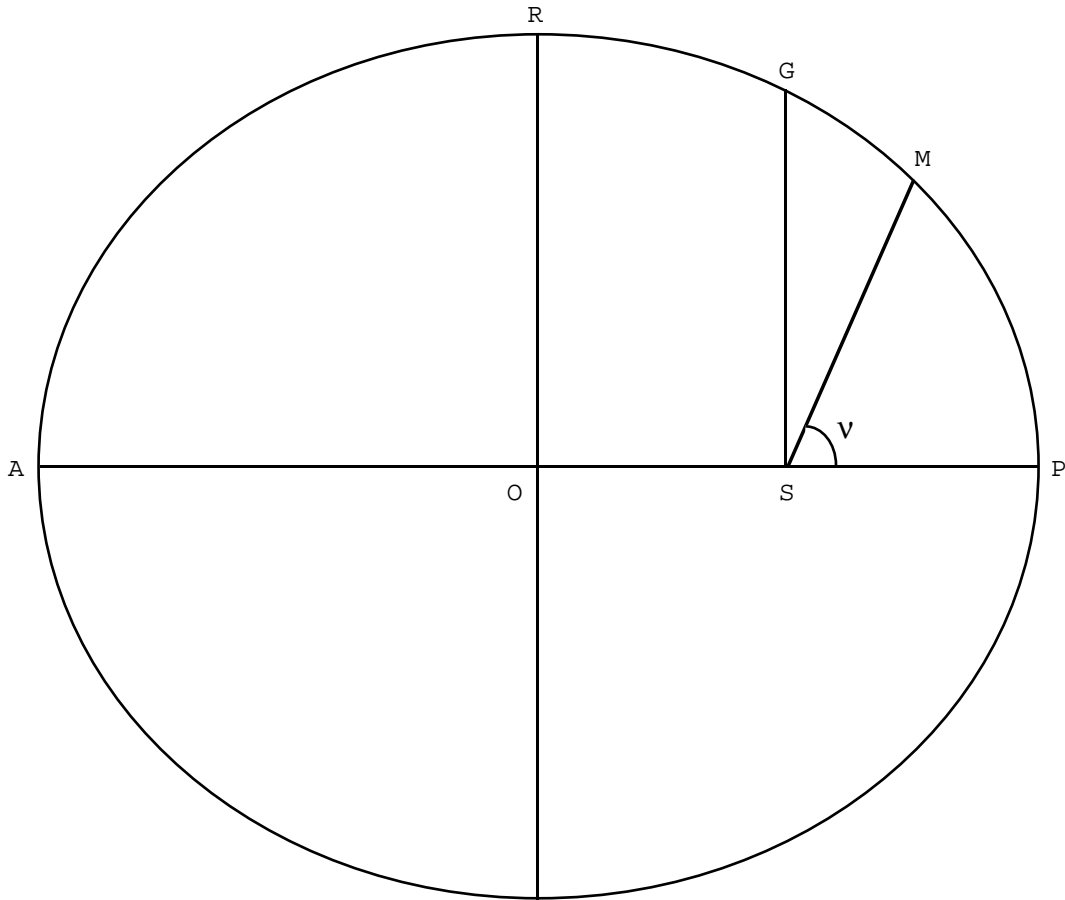


Figure 9. Ellipse (eccentricity = 0.5)

Table 4. Eccentricities of Conic Sections

Eccentricity	Shape
0	Circle
(0, 1)	Ellipse
1	Parabola
> 1	Hyperbola

With these preliminaries, we may now examine Kepler's data, shown in Table 5 in modern units, along with the actual values for the times indicated [KEP09, STA97].¹⁷

¹⁷ see files *Examples:Kepler_Mars.in* and *Examples:True_Mars.in* Note that these data do not comprise a single Martian year; they were observed during a sequence of oppositions.

Table 5. Orbital Data for Mars

Time (JD – 2300000) ¹⁸	Radius (AU)			True Anomaly (radians)	
	Kepler	Predicted	Actual	Kepler	Actual
–789.86	1.58852	1.58853	1.58910	2.13027	2.12831
–757.18	1.62104	1.62100	1.62165	2.39837	2.39653
–753.19	1.62443	1.62443	1.62509	2.43037	2.42854
–726.27	1.64421	1.64423	1.64491	2.64327	2.64152
–30.95	1.64907	1.64907	1.64984	2.70815	2.70696
2.84	1.66210	1.66210	1.66288	2.96844	2.96732
13.74	1.66400	1.66396	1.66473	3.05176	3.05065
49.91	1.66170	1.66171	1.66241	3.32781	3.32675
734.17	1.66232	1.66233	1.66356	3.30693	3.30640
772.03	1.64737	1.64738	1.64812	3.59867	3.59818
777.95	1.64382	1.64383	1.64456	3.64488	3.64440
819.86	1.61027	1.61028	1.61083	3.97910	3.97865
1507.15	1.61000	1.61000	1.61059	3.98145	3.98157
1542.94	1.57141	1.57141	1.57186	4.28018	4.28035
1544.97	1.56900	1.56900	1.56944	4.29762	4.29779
1565.94	1.54326	1.54327	1.54210	4.48063	4.48083
2303.05	1.47891	1.47889	1.47886	4.94472	4.94565
2326.98	1.44981	1.44981	1.44969	5.18070	5.18169
2330.96	1.44526	1.44525	1.44512	5.22084	5.22184
2348.90	1.42608	1.42608	1.42589	5.40487	5.40591
3103.05	1.38376	1.38377	1.38332	6.12878	6.13067
3134.98	1.38463	1.38467	1.38431	0.198457	0.200358
3141.90	1.38682	1.38677	1.38643	0.274599	0.276497
3176.80	1.40697	1.40694	1.40676	0.653406	0.655260
3891.17	1.43222	1.43225	1.43206	0.940685	0.943206
3930.98	1.47890	1.47888	1.47896	1.33840	1.34079
3937.97	1.48773	1.48776	1.48789	1.40552	1.40788
3982.80	1.54539	1.54541	1.54583	1.81763	1.81986

Since Kepler intended these data to be illustrative of his law of ellipses, it is reasonable to ask whether they are, in fact, described by the model of Equation 14. The graph in Figure 10, where X is true anomaly and Y is radius, makes it qualitatively apparent that they are.¹⁹

This assessment may be quantified by **hypothesizing** that the information explained by this model and the residual error constitute independent sources of variation. If this is true, then the total variance of the dataset is the sum of the variances for each of these components

¹⁸ JD (Julian date) \equiv interval, in days, from Greenwich noon, 1 January 4713 B.C.

¹⁹ The ellipse, in this (optimized) model, has $A = 1.51043$ AU and $B = 0.0926388$.

and it would then be a trivial matter to compare the fraction of the variance/information explained by this model to the total variance in the dataset. A good model is one that leaves relatively little unexplained.

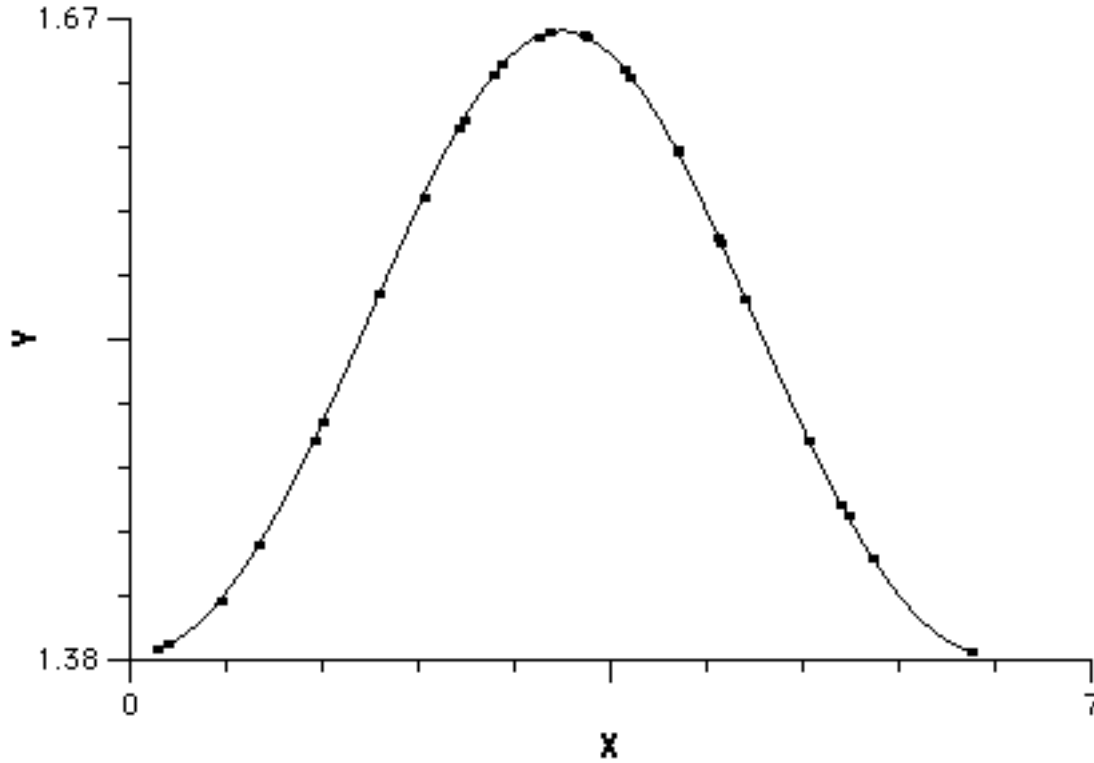


Figure 10. Kepler's Data and Model

Since the number of points is constant throughout this example, we may substitute sums of squares for the respective variances. The information not explained by the model, plus the variance due to any errors, is then equal to the *error-sum-of-squares*, ESS, often called the *residual-sum-of-squares*, defined in Equation 15.

$$\text{Error-sum-of-squares} \equiv \text{ESS} = \sum_{i=1}^N (\hat{y}_i - y_i)^2 \quad 15.$$

where \hat{y} is the value predicted by the model.

Therefore, the fraction of the total variance in the dataset that is explained by a given model is equal to the statistic *R-squared*, defined in Equation 16.

$$\text{R-squared} = 1 - \frac{\text{ESS}}{\text{TSS}} \quad 16.$$

Kepler's data, plus the predictions of the optimum model (Table 5, column 3), produce the following results (computed to six significant figures):

$$\begin{aligned}\text{TSS} &= 0.273586 \text{ AU}^2 \\ \text{ESS} &= 1.257\text{e-}08 \text{ AU}^2 \\ \text{R-squared} &= 1.00000\end{aligned}$$

In other words, by this statistic/criterion, the model described in Equation 14 and Figure 10 explains all of the deterministic information contained in Kepler's dataset to a precision of one part in 100,000.

This result is not quite perfection. Since $\text{ESS} > 0$, there remains something, true deviations or error, yet to be explained.²⁰ Also, the 28 radii listed by Kepler are not really as accurate as his values indicate. Nevertheless, this degree of success is impressive. There are thousands of sociologists, psychologists, and economists who would be more than delighted with results of such precision.

We shall see later how Kepler's model was optimized to find parameters yielding this value for R-squared.

Example D3—World Track Records

As a measure of information and/or error, the variance statistic reigns supreme in most of contemporary analysis of deterministic data. However, intuition suggests that the “best” deterministic model is the one that minimizes the *average deviation*, defined in Equation 17. When graphed, this model gives the curve that is as close as possible to all of the data points simultaneously, using a statistic with the same units as the dependent variable.

$$\text{Average Deviation} = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i| \quad 17.$$

where \hat{y} is the value predicted by the model.

To illustrate this metric, consider the data in Table 6.²¹ This table lists 14 distances for which world track records (for men) are recognized. The corresponding average speed attained during each record run is listed as well [YOU97]. The task is to model this speed as a function of distance.

Not surprisingly, the average speed decreases sharply, at first, but then seems to level off somewhat. Many analytical forms would be suitable but we shall examine the power law given in Equation 18 (where y is speed and x is distance).

$$y = A x^B + C \quad 18.$$

²⁰ That this “something” is relatively small does not imply that it is spurious or insignificant.

²¹ see file *Examples:Track.in*

Table 6. World Track Records (Men)

Speed (m s ⁻¹)	Distance (m)
10.152	100
10.173	200
9.240	400
7.864	800
7.565	1,000
7.233	1,500
7.172	1,609.344 (mile)
6.947	2,000
6.740	3,000
6.449	5,000
6.236	10,000
5.855	20,000
5.636	25,000
5.598	30,000

The model of this form with the minimum average deviation has the parameter values given below. Its graph is shown in Figure 11.

$$\begin{aligned}A &= 18.35 \\B &= -0.2380 \\C &= 4.020\end{aligned}$$

The value of the average deviation resulting from this model is 0.163 meters per second; the worst deviation is -0.954 meters per second (point #2). According to the R-squared metric, this model explains 94.691 percent of the total information in the dataset.

Had we, instead, used the R-squared criterion to optimize the model, we would then have obtained the following parameters:

$$\begin{aligned}A &= 20.36 \\B &= -0.2369 \\C &= 3.785\end{aligned}$$

The latter model would have explained 96.634 percent of the information in the dataset but with an average deviation of 0.203 meters per second and a worst deviation of -0.582 meters per second (point #2). Evidently, just as there is more than one conceivable model for these data, there is also more than one way to optimize that model.

The minimum deviation is often used with data thought to contain one or more *outliers*, data points with unusually large deviations from the putative model. Minimizing the average deviation is more *robust* than minimizing ESS, signifying, among other things, that the computed model is not as sensitive to the presence of extreme values as it is when apparent deviations are squared. The literature on robust estimation is quite large and several good techniques have been developed over the years.

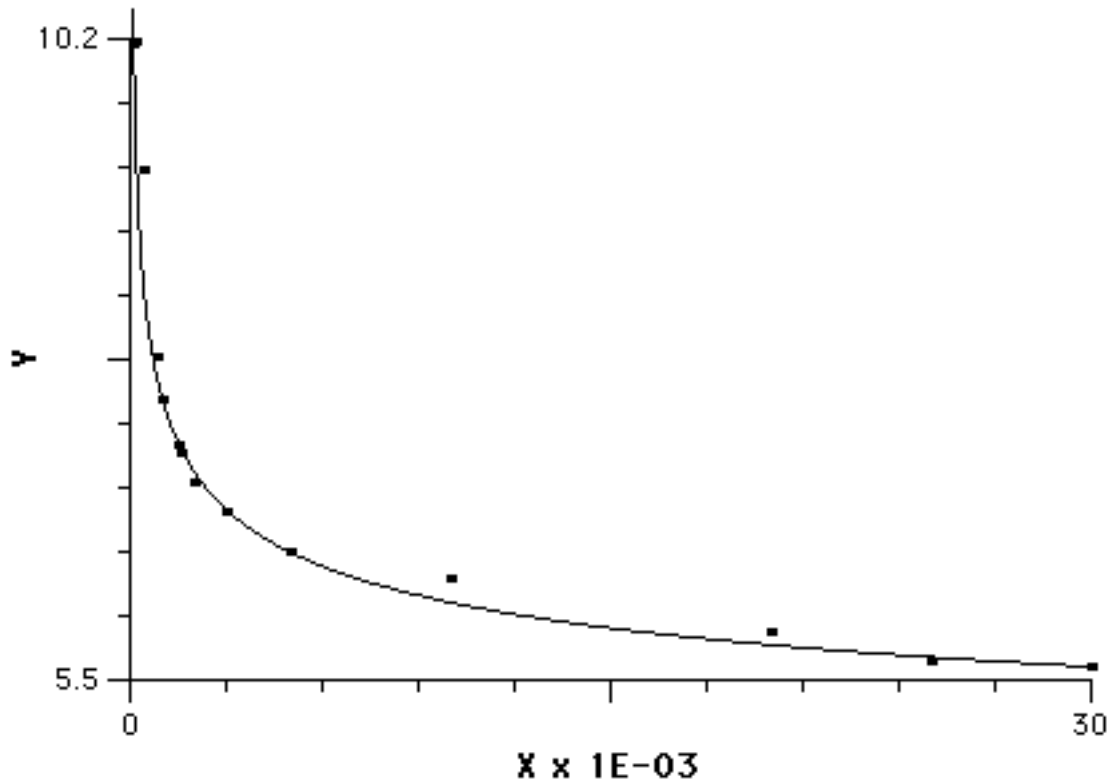


Figure 11. World Track Records

Deterministic Information—Summary

In this section, we have described two metrics for deterministic modeling, giving different answers. Is the choice a matter of indifference or is one better than the other?

To anyone trying to understand the real world, there can be only one “best” answer because there is only one Universe. Nature does not use one set of rules on weekdays and another on weekends. To be credible, a model must eschew arbitrariness to the greatest extent possible. Moreover, it must be completely consistent with whatever is already known about the data it describes, including even aspects of the data that are not the subject of the model.

As noted earlier, any deterministic model should be optimized utilizing whatever is known about the errors in the data. In Examples D2 and D3, this concern was not addressed. So far, we have considered only the fraction of information explained and average deviation (error), without considering specific properties of the residual variance. Finally, in discussing these two models, we did not relate much of what we said about stochastic modeling, especially the powerful concept of maximum likelihood, to the fact that errors in data are largely random.

In the next section we shall put everything together. Our efforts will be rewarded with a set of valid, intuitive, and well-defined criteria for computing optimum model parameters.

FINDING OPTIMUM PARAMETERS

Data, as we have seen, come in two varieties: stochastic and deterministic. Each of these may be modeled and that model characterized by one of a number of metrics (statistics) relating something about the quality of the model. Quality implies descriptive or explanatory power, proof of which is manifested in predictive capability. All of this presupposes, of course, that models exhibiting these desiderata can actually be found.

This section clarifies some relationships among the five metrics described earlier. In doing so, a general technique for finding optimum parameters will be cited. It will be apparent that this technique may be readily implemented as a computational algorithm.

Five Criteria

The five metrics/statistics that we shall utilize as modeling criteria are listed in Table 7. The two primary criteria are shown in **bold** and their preeminence will be discussed.

Table 7. Modeling Criteria

Criterion	Applicability
Maximum likelihood	All random variates
Minimum K-S statistic	Continuous random variates
Minimum chi-square statistic	Discrete random variates
Minimum ESS (<i>least squares</i>)	Deterministic data
Minimum average deviation	Deterministic data

Some of the reasons why these five are desirable criteria were described in the last section. It was taken for granted there that, given an appropriate analytical form for the model, finding parameters that would optimize the chosen metric was practicable. Indeed, most of the values for model parameters given in the text are illustrative.

The methodology described here takes a modeling criterion as a kind of meta-parameter. Partial justification for this approach derives from the fact that these five criteria are more interrelated than they might appear. In particular, as first demonstrated by Carl Friedrich Gauss (1777-1855) and Adrien Marie Legendre (1752-1833), there are deep, fundamental connections linking the criteria for stochastic and deterministic modeling. These relationships are due to the randomness of error.

Maximum Likelihood Redux

In the course of any description of modeling and statistics, it is easy to get lost in the almost inevitable deluge of symbology and equations and, in the confusion, lose sight of the original objective. One is dealing, presumably, with real data from the real world. Thus, a model is not just some computational gimmick for reproducing a matrix of numbers. A good model says something genuine about the Universe. Its analytical form is discovered, not invented; its parameters are properties of Nature, not the whim of an analyst.

The ML criterion acknowledges this role, as well as the supremacy of data, in the most direct fashion. Given the form for a stochastic model, ML parameters maximize the probability that the observed data came from a parent population described by the model. Deterministic modeling, on the other hand, attempts to maximize the explanatory power of the model or, occasionally, to minimize the average deviation. It turns out that there is a profound connection between least squares (also, minimum deviation) and maximum likelihood.

We shall start with least squares. This is undoubtedly the most widely used of all modeling criteria. It is usually presented simply as a technique for generating a curve that is closest to all of the data points. Of course, it isn't. The minimum-deviation criterion is the one that yields that particular curve. As we have seen, the least-squares technique actually minimizes the error **variance** and, hence, maximizes the fraction of information that is explained by the model. If it did nothing else, one could hardly complain. However, it does much more.

In real life, errors are almost never simple manifestations of a single mechanism. Nearly always, the total error in an observation is the net effect of several different errors. As a result, the Central Limit Theorem takes over and stipulates that such errors have a strong tendency to be $\sim \text{Normal}(A, B)$. This theorem was illustrated in Example S3.

In devising a deterministic model, suppose that this theorem and the proliferation of error modes are taken at face value. Then one might ask, "What set of parameter values maximizes the likelihood of the residuals if they are normally distributed? Also, is this vector unique?"

Thus, assume that the residuals for the data points, $\epsilon_k \equiv \hat{y}_k - y_k$, are $\sim \text{Normal}(A_k, B_k)$. In addition, assume that these residuals are all unbiased so that A_k is always zero. Then, the likelihood is given by Equation 19 (cf. Eq. 10 and pg. A-85).

$$\text{Likelihood} \equiv \prod_{k=1}^N f(\epsilon_k) = \prod_{k=1}^N \left[\frac{1}{B_k \sqrt{2\pi}} \exp \left(-\frac{1}{2} \left(\frac{\epsilon_k}{B_k} \right)^2 \right) \right] \quad 19.$$

Equation 19 can be simplified. The $\log()$ function is monotonic, so any set of parameters that maximizes $\log(\text{likelihood})$ will necessarily maximize the likelihood. Taking (natural) logs of both sides of Equation 19 produces Equation 20.

$$\text{Log-likelihood} = -\sum_{k=1}^N \log(B_k \sqrt{2\pi}) - \frac{1}{2} \sum_{k=1}^N \left(\frac{\epsilon_k}{B_k} \right)^2 \quad 20.$$

Given a set of measurement errors, the first term in Equation 20 is a constant and has no bearing on the maximization. The likelihood will be a maximum if and only if the expression in Implication 21 is a minimum.

$$\text{Maximum likelihood} \Rightarrow \min \left[\sum_{k=1}^N \left(\frac{\epsilon_k}{B_k} \right)^2 \right] \quad 21.$$

If the errors for all of the points come from the **same** normal distribution, then all B_k are equal and, after canceling the units, this constant can be factored out giving Implication 22.

$$\text{Maximum likelihood} \Rightarrow \frac{1}{B} \min \left[\sum_{k=1}^N \varepsilon_k^2 \right] \Rightarrow \min \left[\sum_{k=1}^N (\hat{y}_k - y_k)^2 \right] \quad 22.$$

However, the term on the far right is just ESS! In other words,

if all errors are $\sim \text{Normal}(0, B)$, then the parameters that minimize ESS are the very same parameters that maximize the likelihood of the residuals.

It turns out that this vector of parameters is almost always unique. The least-squares parameters have many other useful properties as well but these lie beyond our present scope.

While this is a wonderful result, it falls short of perfection for at least two reasons. First, the fact that the parameters are the ML parameters says nothing about the analytical form of the model. If the model itself is inappropriate, its parameter values are academic. Second, there is no guarantee that the errors are independent or normally distributed. These hypotheses must be proven for the least-squares method to yield the ML parameters.

However, the strong assumption that all B_k are equal is unnecessary **provided that their true values are known**. Minimizing Implication 21, instead of Implication 22, is referred to as the *weighted least-squares* technique.²²

Finally, what about the minimum-deviation criterion? Can that be related to the maximum-likelihood function as well?

Yes, it can. The second most common distribution for experimental errors is the Laplace distribution (see pg. A-63). Substituting the Laplace density for the Normal density, the derivation above produces Implication 23 instead of Implication 21.

$$\text{Maximum likelihood} \Rightarrow \min \left[\sum_{k=1}^N \frac{|\hat{y}_k - y_k|}{B_k} \right] \quad 23.$$

Implication 23 is equivalent to minimizing the (weighted) average deviation, analogous to weighted least squares. Of course, the same caveats apply here as in the case of normal errors.

Alternate Criteria

The two criteria that do not relate directly to maximum likelihood are the minimum K-S criterion and the minimum chi-square criterion. Nevertheless, the first of these is justified because it is the most common goodness-of-fit test for continuous random variates, as well as one of the most powerful. The chi-square criterion is the discrete counterpart of the K-S criterion and any discussion of discrete variates would be very incomplete without it.

²² Some texts discuss only the latter since setting all B_k equal to one is always a possibility.

From this point on, we shall take it as given that the criteria listed in Table 7 are not only valid and appropriate but, in many respects, optimal. The next task is, therefore, to compute the parameters that realize the selected criterion.

Searching for Optima

As noted earlier, a single computational technique will suffice regardless of the selected criterion. All optimizations of this sort are essentially a search in a multidimensional parameter space for values which optimize the *objective function* specified by this criterion. There are many generic algorithms from which to choose. One of the most robust is the *simplex method* of Nelder and Mead.²³

The simplex algorithm traverses the parameter space and locates the **local** optimum nearest the starting point. Depending upon the complexity of the parameter space, which increases rapidly with dimensionality, this local optimum may or may not be the **global** optimum sought. It is, as always, up to the analyst to confirm that the computer output is, in fact, the desired solution.

The simplex is a common algorithm and source code is readily available [CAC84, PRE92].

Summary

Three of the five modeling criteria described in the first section have been shown to be closely linked to the concept of maximum likelihood. Consequently, they are eminently suitable as modeling criteria. They are intuitive, valid, and appropriate. Moreover, they almost always provide unique parameter vectors for any model with which they are employed.

Two other criteria were also listed. Both of these are best known as goodness-of-fit criteria for random variates with respect to some putative population density. However, they may also be employed to determine the parameters for this density, not just to test the final result.

The simplex algorithm will prove suitable, in all cases, to compute parameters satisfying the chosen criterion in any modeling task that we shall address. The truth of this assertion has not yet been demonstrated but it will become evident in the sections to follow. Some of the figures shown above provide examples.

To this point, we have seen, or at least stated, how models may be specified, optimized, and computed. The stage is now set to evaluate the results of any such endeavor. We shall discuss the quality of models in general and of their parameters in particular. Is the model any good? Are the parameters credible? Does the combination of model and parameters make sense in the light of what we know about this experiment and about the Universe in general? In other words, are we doing real science/engineering/analysis or are we just playing with data?

²³ not to be confused with the linear programming technique of the same name

IS THE MODEL ANY GOOD?

The “goodness” of a model depends not on how well it might serve our purposes but on the degree to which it tells the truth. Beauty, in this case, is not entirely in the eye of the beholder. If a model is based upon observed data, especially physical data about the real world, then the model must be equally real. Several criteria have been established, in earlier sections, that measure the validity (i.e., the reality) of a model. In this section, we shall focus mainly on the statistical dimension of this reality; the physical dimension is beyond our scope.

Deterministic data and stochastic data will be treated separately because the former case is very easy but the latter is not. However, as we have seen, it is the modeling of error as a random variable that lends credence to the aforementioned deterministic criteria. Thus, the validity of stochastic models is fundamental to the validity of modeling in general.

In this section, we encounter a new technique—the *bootstrap*. The bootstrap may be *parametric* or *nonparametric*. In the discussions to follow, both forms will be employed. The utility of these techniques should be self-evident.

The Bootstrap

Since there is ample literature on bootstrap methodology [EFR93], the descriptions here will be somewhat abbreviated. While the name is fairly new, some of what is now termed “bootstrap” methodology is not. In particular, the parametric bootstrap includes much that was formerly considered simply a kind of Monte Carlo simulation. However, the nonparametric bootstrap, outlined in the next section, is quite recent.

The bootstrap technique was designed, primarily, to establish *confidence intervals* for a given statistic. A confidence interval is a contiguous range of values within which the “true” value of the statistic will be found with some predetermined probability. Thus, the two-sided, 95-percent confidence interval for a given K-S statistic is that range of values within which the true K-S statistic will be found in 95 percent of all samples drawn from the given population, being found 2.5 percent of the time in each tail outside the interval. Other things being equal, the width of this interval decreases with increasing sample size.

Sometimes, a confidence interval may be computed from theory alone. For instance, means of large, random samples tend to be unbiased and normally distributed. Therefore, the 95-percent confidence interval for any such mean is just $\mu \pm 1.96 \text{ SE}$, where μ is the observed mean and SE is the *standard error* of the mean, given by Equation 24.

$$\text{SE} = \frac{\sigma}{\sqrt{N}} \quad 24.$$

where σ is the population standard deviation for the data and N is the sample size.

As an example, consider again the data exhibited in Figure 7. Theory says that these 1,000 means should, themselves, have an (unbiased) mean of 18 and a standard deviation (SE) of 0.30. Hence, 95 percent of them should lie in the interval $18 \pm 1.96 \cdot 0.30 = [17.41, 18.59]$, with the remaining five percent divided equally between the two tails. In fact, if you sort the

file *Examples:Rolls3avg.in*, you will find that these 1,000 means exhibit a 95-percent central confidence interval of [17.40, 18.55]. The latter interval is, of course, just a sample of size one of all similar confidence intervals for similar data. As such, it is expected to deviate a little bit from theory. If you sense a little recursion going on here, you are quite right.

What we have just done, without saying so, was a form of parametric bootstrap. The file of 1,000 means was obtained by simulating the original experiment 1,000 times. Each trial was terminated when an event with a probability of **1/6** was realized **3** times. The simulation used these two **parameters** to synthesize 1,000 *bootstrap samples*, each sample containing 1,000 trials, with only the sample mean recorded. The explicit use of input parameters makes this a parametric bootstrap.

Figures 7 and 8, and the accompanying discussion, illustrate not only the Central Limit Theorem but also the validity of the bootstrap technique. This technique has received a great deal of scrutiny over the past decade, with very positive results. We shall put it to good use. Almost all of our remaining discussion will derive from the now well-established fact that

bootstrap samples are valid samples.

Whether they are created from theory (parametric bootstrap) or experiment (nonparametric bootstrap), bootstrap samples constitute **additional** data. These data may be characterized by one or more statistics and the ensuing distribution of these statistics utilized to estimate any desired confidence intervals. By comparing optimum values, for any model, to the confidence intervals for these values, the probability of the former may be estimated. In this section, this probability will be related to goodness-of-fit; in the following section, it will be employed to estimate parameter uncertainty.

Stochastic Models

Table 7 listed three criteria for optimizing a stochastic model. Unlike their deterministic counterparts, none of these criteria are self-explanatory. In general, it is not easy to tell, from the value of the given statistic, whether or not that value is probable under the hypothesis that the optimum model is valid.

For example, the K-S statistic for the best-fit Gumbel distribution to the Batting-Average data is 0.057892. This number represents the maximum (unsigned) discrepancy between the theoretical and empirical CDFs. So much is very clear. What is not at all clear, however, is whether or not **a random sample (N = 97) from a genuine population described by the same distribution** would be likely to exhibit a K-S statistic having the same value. It could be that the value 0.057892 is too large to be credible.

How could you tell? The traditional approach would be to look up a table of K-S critical points in a reference and compare the observed value to some table entry. The problem with this approach is that nearly all such tables list asymptotic results, i.e., with $N = \text{Infinity}$. Even those that have entries for specific sample sizes are not always appropriate in every case. There might be hidden dependencies on model form or parameters that could invalidate the recorded critical points. Ideally, what one would really like is a made-to-order table, conditioned on the sample size, the model form, and the model parameters. Before the advent of fast digital computers, such a table was a rare luxury. Today, it is fairly easy.

To construct such a customized table, we start with the putative parent population. In this case, we start with the Gumbel distribution given in Equation 25.

$$\text{PDF} = \frac{1}{0.01984} \exp\left(\frac{0.3555 - y}{0.01984}\right) \exp\left(-\exp\left(\frac{0.3555 - y}{0.01984}\right)\right) \quad 25.$$

From this distribution, which purports to be a good description of our data, draw a random sample ($N = 97$) a very large number of times (at least 1,000 times). With each of these samples, **carry out the same operations that were performed on the original data**. Here, we assume that each sample is $\sim \text{Gumbel}(A, B)$ and we optimize the model, for each sample, using our original criterion. When all of this is done, we have a vector of 1,000 **bootstrapped** K-S statistics, all of which a) genuinely reflect our candidate population, by construction, and b) were obtained exactly as was the original K-S statistic.

There are now several ways to proceed but, here, we shall adopt the simplest approach. First, sort the 1,000 bootstrapped statistics in ascending order. The result will look much like the histogram in Figure 12 (where $Y = \text{K-S statistic}$).

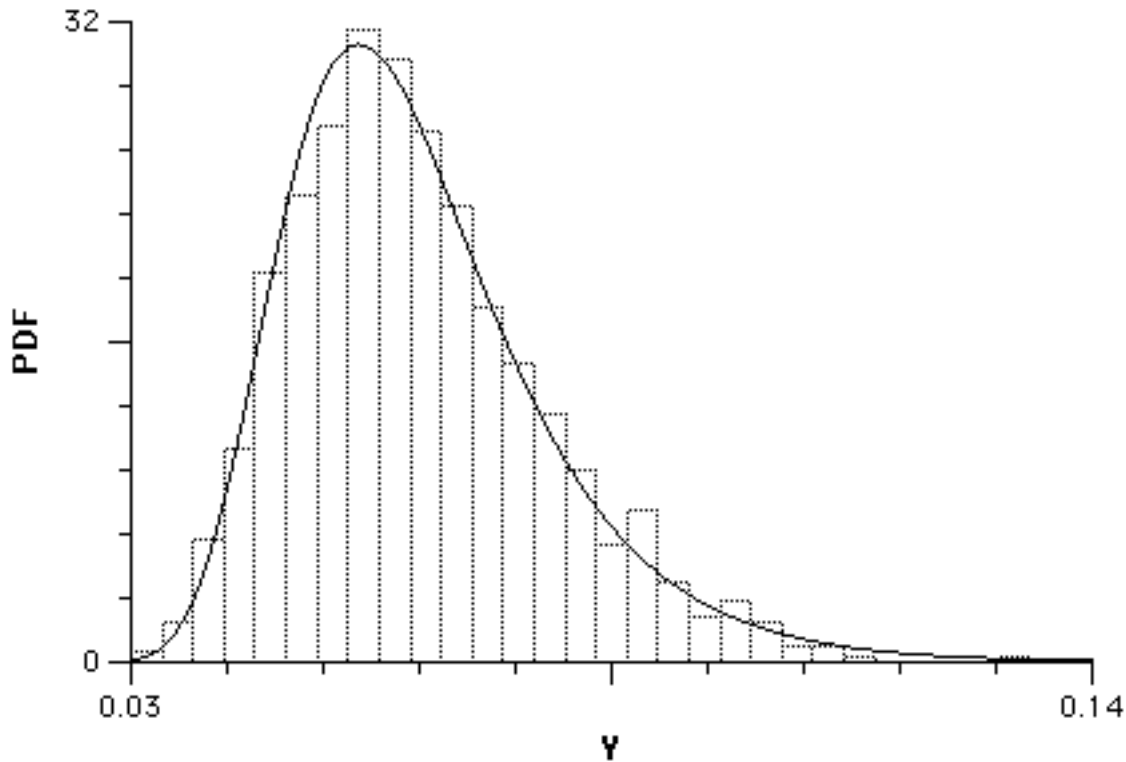


Figure 12. Distribution of K-S Statistic from Equation 25

The bootstrapped K-S statistic is, itself, a random variable and we could try to model it. Since each of these continuous variates represents the maximum of 2×97 comparisons,²⁴ they might even be $\sim \text{Gumbel}(A, B)$!²⁵ However, we shall instead use just the raw values. It turns out that our observed value of K-S is found in the 41st *percentile* of this specific K-S distribution. In other words, if you draw a sample ($N = 97$) from the distribution given by Equation 25, model it as $\sim \text{Gumbel}(A, B)$, and recompute the optimum A and B using the ML criterion then, 59 percent of the time, the K-S statistic you will get will be larger (**worse**) than the one we obtained with the Batting-Average data.

It is customary to require the 95th percentile before rejecting an hypothesis which means that we **cannot** reject the null hypothesis that our data are modeled by Eq. 25. While not rejecting an hypothesis is not quite the same thing as accepting it, in practice, this is the usual interpretation. Therefore, we conclude that

according to the K-S test, Eq. 25 is an ACCEPTABLE description of our data.

We could go through the same procedure with the 1,000 Log-likelihood statistics that resulted from this bootstrap process, assessing the observed value in the same fashion. Then, we would have two measures of acceptability. The prudent course would be to require both of them to be acceptable before declaring that the model is good. In this example, the observed value falls in the 47th percentile of its (bootstrapped) distribution so it, too, is acceptable.

With discrete variates, the usual goodness-of-fit statistic is Chi-square. The same procedure would apply there as well. In fact, the bootstrap technique is a very general method for assessing goodness-of-fit, no matter what statistic/criterion is used.

There is also a simple rule-of-thumb when using the Chi-square statistic as described. Its expected value is equal to the number of degrees of freedom.²⁶ Therefore, if the Chi-square value found is less than the range of y-values, it will always be acceptable.

Deterministic Models

Since the bootstrap technique is generally valid, it could be used for assessing goodness-of-fit with deterministic models as well. However, R-squared, the usual statistic in such cases, is self-explanatory. It is equal to the fraction of the variance explained by the model. There is no need to ask about the probability of such a value with a given dataset/model combination.

The minimum-deviation statistic is not quite as transparent. Still, this statistic has the same units as the data and it is usually easy to tell if a given minimum deviation is small enough for the intended purposes.

²⁴ Each end of each “step” in the empirical CDF must be examined.

²⁵ Indeed they are, $\text{Gumbel}(0.05604, 0.01192)$ [Fig. 12, solid line]. In fact, this model is an even better fit than it was with the Batting-Average data. See also, *Examples:KStest.in*.

²⁶ see pg. A-42

All of the above is true whether the deterministic modeling (*regression*) is weighted or unweighted. With weighted regressions, the weighted residuals take the place of the usual residuals since the former are now normalized with respect to their own standard errors. However, the goodness-of-fit metrics, and their meaning, remain the same.

Summary

This section has outlined a general technique for establishing the validity of a mathematical model. If a model is deemed acceptable, it signifies that the equation (or density function) may substitute for the original data. Moreover, the model may now be queried, perhaps in ways that would be impossible with the actual dataset. Deterministic models may likewise be used to interpolate or extrapolate from the observed values. In short, if a model is acceptable, it is equivalent to any and all past (or future) observations.

With stochastic models, acceptability is assessed by comparing one or more goodness-of-fit statistics (random variables) to their respective distributions, asking whether the value observed is too improbable to credit. With deterministic models, the goodness-of-fit statistics are sufficiently perspicuous that we need not go to so much trouble.

Once again, it must be emphasized that none of this is meant as some kind of mathematical gimmick. The number of potential models is infinite and, with enough fortitude, any dataset could be adequately fit to one or more of them. The real goal is to find the model that Nature uses by examining a sample of observations. The techniques described herein are immensely powerful but they are blind. The necessary vision must be supplied by the analyst.

HOW PRECISE ARE THE PARAMETERS?

Our goal has been to show how to obtain a validated mathematical model for any dataset. This we have done. Yet the story is not quite complete. Certainly, an acceptable model of the proper form, with optimum, ML parameters cannot be significantly improved but there remain two loose ends.

One loose end is that the procedure described above for determining ML parameters is just a search of the parameter space for the best parameter vector. In general, there is no guarantee either that there is only one best solution or, if there is, that you will find it. Depending upon where you start looking, you might converge on a local optimum instead of the desired global optimum. Also, the latter might be degenerate, i.e., there could be more than one of them.

For instance, the ML fit of a sine wave to the Daytime data (Table 1) is shown in Figure 13 (cf. Figure 1). This fit (Eq. 26) has $R\text{-squared} = 0.99908$ so this model is good to one part in a thousand but simple, high-school trigonometry reveals that the phase term, -1.391 , may be incremented by any integer multiple of 2π (360 degrees) without changing the daytime value, y . Therefore, the number of ML solutions is actually infinite.

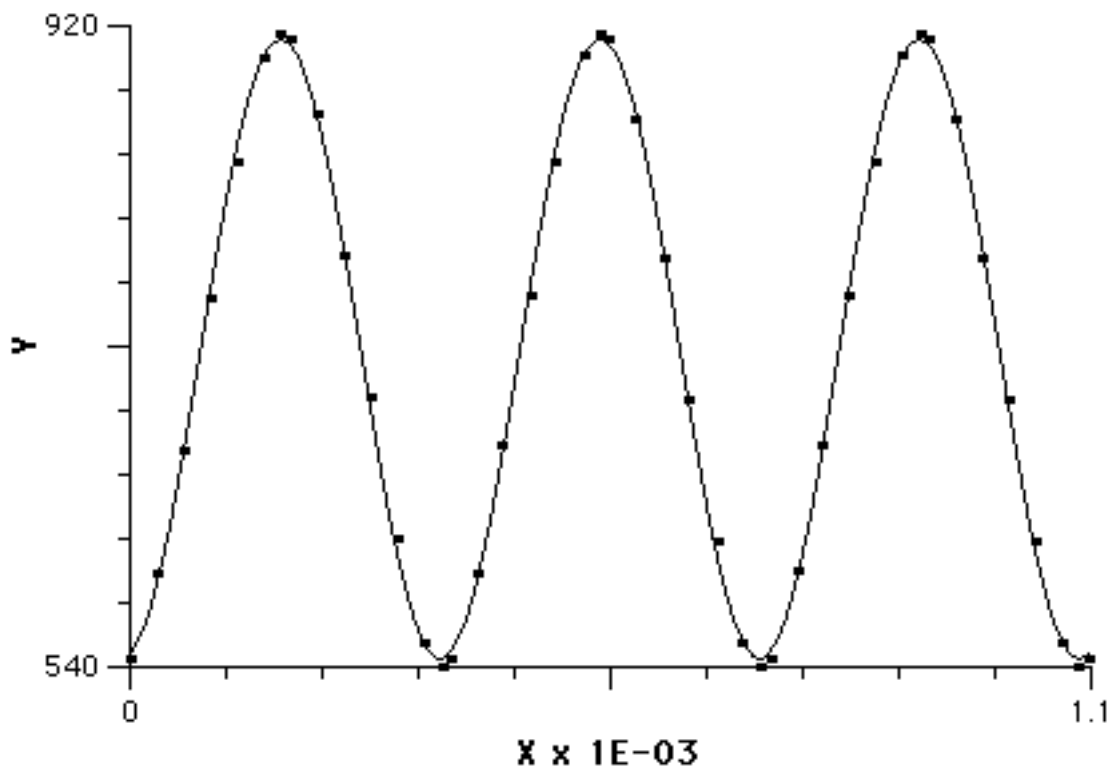


Figure 13. Least-squares Sine Fit to Daytime Data

$$y = 183.3 \sin ((2 \pi) 0.002736 x - 1.391) + 728.4$$

26.

The other loose end stems from the definition of maximum likelihood. ML parameters are, by definition, more probable than any others—but how probable is that? Suppose you propose to flip a coin 100 times and bet on the outcome. The wise bet would be 50 Heads and 50 Tails because that is the ML result; every other result is less likely. Suppose, however, that some bystanders are watching all of this and decide to bet, amongst themselves, whether or not you will win your bet. The wise choice in this case would be to bet that you would lose. Even though, with a fair coin, a 50:50 outcome is the most probable, its probability is still less than eight percent. There are just too many other possibilities for any one of them to be at all likely. So your chances of losing are, at best, more than 92 percent.

The probability that the ML parameters in a model are equal to the true parameters is even worse. Unless theory insists that a given parameter must be a rational number, then you can be absolutely certain that the optimum value you found for it will always be incorrect. The reason, noted earlier, is that selecting a specific number from the real domain has a probability of zero. You can never do it exactly.²⁷

But you can get as close as you like. All it takes is a lot of effort and a **very** large sample.

Sampling is the key. The model for the radius of the orbit of Mars (Table 5, col. 3) as a function of true anomaly has two ML parameters: $A = 1.51043$ AU and $B = 0.0926388$. The Keplerian model is an approximation, albeit a very good one, to the true solution. Nevertheless, even if we grant that the elliptical approximation is good enough, these two parameters are not perfect. Were new data found for exactly the same time frame, these parameters would be different **even though the orbit of Mars was constant**. Every dataset is a sample, not the entire population. Therefore, there is always some sampling error.

Nothing can be done to compensate for degenerate global optima except to examine the ML parameters for reasonableness. They must make sense! They must have the correct units and be consistent with what is known about the real world. This is essential even if the solution is not degenerate. Likewise, parameter uncertainties due to sampling error, or to deficiencies in the model, cannot be removed once the ML result has been obtained. However, they can be estimated.

The purpose of this section is to describe a procedure for estimating parameter uncertainty. In addition, parameter correlation is also discussed. The latter is a subtle kind of uncertainty which recognizes that parameters are not necessarily independent. We shall take it as given that the model is acceptable since a model that is not acceptable is of little interest and its parameters even less so.

To quantify uncertainty, we shall utilize confidence intervals. Every optimum parameter can be assigned a confidence interval within which the true value of the parameter will be found with some specified probability. The wider this confidence interval, the more uncertain the parameter value.

²⁷ assuming that you cannot go out and examine every member of the parent population

As we have already seen, confidence intervals may be estimated *via* the bootstrap. For this application, however, the parametric bootstrap is not suitable. We are not asking now about a theoretical population from which we can select random variates. Rather, we are asking about one particular dataset, the one that gave rise to the model. Hence, we need a **nonparametric** bootstrap that somehow uses this dataset to generate further bootstrap samples. The requisite methodology is outlined below. As we have done throughout this tutorial, we concentrate on examples instead of theory, first with stochastic models, then with deterministic models.

The Nonparametric Bootstrap

A nonparametric bootstrap sample is created by **resampling the original dataset, with replacement**. Imagine that the original data are in a paper bag. Close your eyes and select one point/variante from the bag, record its value, then put it back. Repeat this selection until you have a sample of the same size as the original. Such a sample is nonparametric; it comes from the actual data, not from a parametric model of that data. Reusing the data in this way might seem like cheating²⁸ but it is quite valid, as long as proper corrections are applied.

In the previous section, when we discussed the parametric bootstrap, a simple approach was used to analyze the bootstrap distributions. For instance, consider the 1,000 K-S statistics in Figure 12. These 1,000 values constitute an empirical CDF for any K-S statistic computed for a sample ($N = 97$) that is $\sim \text{Gumbel}(0.3555, 0.01984)$. To determine a central, 95-percent confidence interval for K-S, using the **percentile** method, we throw away the 25 largest and 25 smallest of these 1,000 K-S values. The remaining $0.95 \times 1,000 = 950$ values comprise the desired confidence interval, conditioned on this model and the given sample size. Its range is $[0.0401747, 0.0974331]$. Its interpretation is that 95 percent of the K-S statistics you would obtain with a genuine sample ($N = 97$) from this Gumbel distribution would fall in this range. Furthermore, 2.5 percent would fall in the range $[0, 0.0401747]$ and the last 2.5 percent in the range $[0.0974331, 1]$.

Were the model or sample size different, the 95-percent confidence interval would be different. The confidence interval would also grow or shrink with the need for increased or decreased confidence, respectively. Thus, the 90-percent confidence interval in this example is $[0.0430050, 0.0901146]$ and the 99-percent confidence interval is $[0.0364837, 0.107393]$.

With the nonparametric bootstrap, it is better **not** to use such a simple approach to find a confidence interval. When the original data are resampled, the resulting bootstrap distribution is biased and skewed. The corrections referred to above compensate for these defects. Instead of using the 25th and 975th elements of the sorted CDF to demarcate the tails of the bootstrap distribution, the *BCa technique* specifies alternate indices.²⁹ Once these have been identified, however, the interpretation of the confidence interval remains the same.

We shall now illustrate the use of these nonparametric confidence intervals to determine the precision (confidence) that we may ascribe to the ML parameters of any model.

²⁸ or, at least, “pulling oneself up by one’s bootstraps”

²⁹ Computing the latter is well understood but rather complicated [see *Technical Details*].

Stochastic Models

Repeat the following procedure 50 times:

- Step 1**
Let X and Y be random Cartesian coordinates $\sim \text{Normal}(0, 3)$.
- Step 2**
Select X and Y, then draw the radius, R, from the origin to the point (X, Y).
- Step 3**
Record the length of R.

At the end of this experiment, we have a sample of 50 values for R.³⁰ Although we shall not prove it, theory says that R is $\sim \text{Rayleigh}(3)$ which is the same as $\sim \text{Chi}(3, 2)$.³¹ However, let us pretend that we know that the distribution should be Rayleigh but have forgotten how to compute the theoretical scale parameter, A. Instead, we shall just determine the ML parameter for a Rayleigh(A) model (Equation 27), based upon this small sample.

$$\text{PDF} = \frac{y}{A^2} \exp\left(-\frac{1}{2} \left(\frac{y}{A}\right)^2\right) \quad 27.$$

The result (see Figure 14, with Y representing R) is quite acceptable, but with $A = 2.847$ instead of 3. What happened? Did we make a mistake? Is the theory wrong? Was our pseudo-random number generator faulty?

The answer is D, none of the above. Our 50 variates are just one sample out of an infinite number of similar samples that this experiment could have produced. It is extremely unlikely that the ML parameter of the $\sim \text{Rayleigh}$ model for any such sample would equal the theoretical prediction to four significant figures. Had the sample size been 1,000,000,000, then perhaps four significant figures might have been anticipated, but not with $N = 50$.

We can be certain of this because, using 2,000 nonparametric bootstrap samples, we find that the estimated 90-percent central confidence interval for A is [2.648, 3.062]. Since even this somewhat conservative (narrow) confidence interval includes the value 3,

we cannot reject the null hypothesis that $A = 3$.

Given the manner in which the 50 radii were created, they necessarily constitute a **genuine** sample ($N = 50$), $\sim \text{Rayleigh}(3)$. The confidence interval given above, therefore, reflects the intrinsic variability of this parameter. Were another sample of 50 radii constructed, its ML parameter would be different from that found for the first sample but the confidence interval would be much the same. The variability of confidence intervals decreases as the number of bootstrap samples increases. A set of 1,000 bootstrap samples is the minimum for experiments such as these.

³⁰ see file *Examples:Radii50.in*

³¹ see pg. A-21

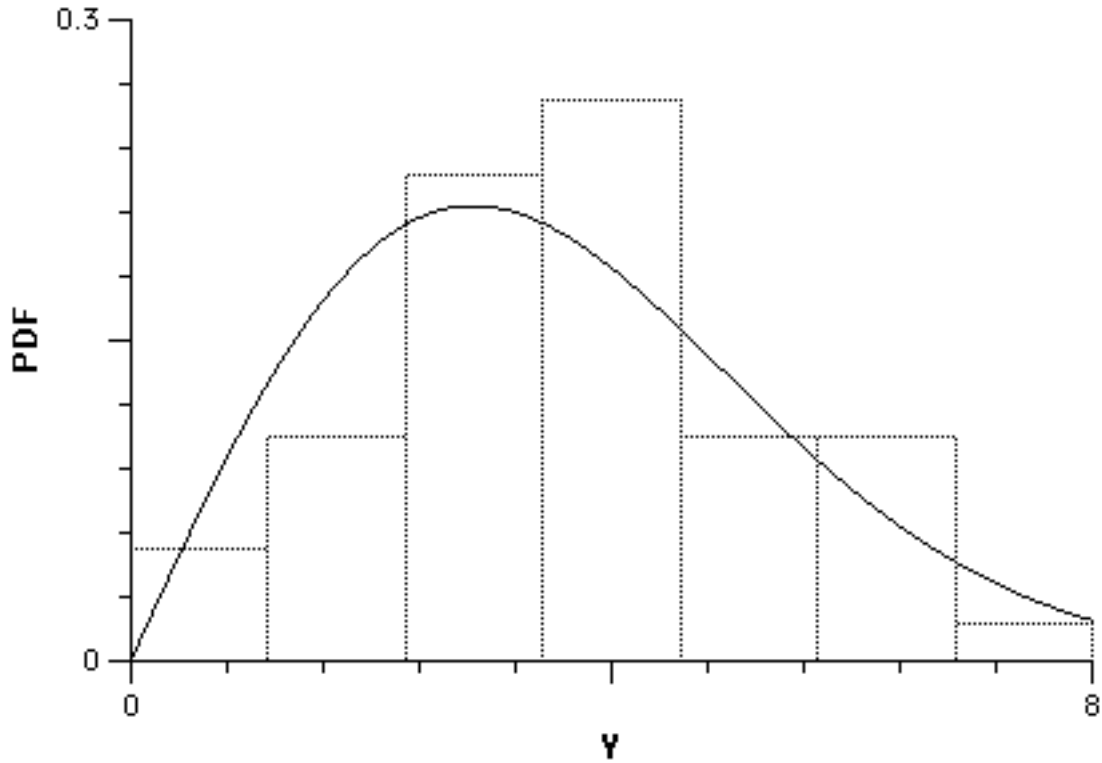


Figure 14. ML Fit of 50 Radii to a Chi Distribution

If we repeat this experiment with 5,000 nonparametric bootstrap samples instead of 2,000, in order to achieve greater accuracy, we obtain the following results for parameter A:

90% --> [2.649, 3.072]
 95% --> [2.597, 3.138]
 99% --> [2.507, 3.264]

At the same time, the corresponding parametric confidence intervals, determined from the model and the percentile method, instead of by resampling the data and the BCa method, are as follows:

90% --> [2.527, 3.186]
 95% --> [2.460, 3.246]
 99% --> [2.327, 3.355]

Finally, it is interesting to compare our sample of 50 radii to its theoretical distribution. If we insist that $A = 3$, we get the theoretical PDF shown in Figure 15 (with Y representing R) along with the empirical histogram representing the data. Using the ML and K-S criteria established in the previous section, plus 2,000 parametric bootstrap samples, it turns out that this fit is likewise acceptable, with likelihood and K-S values falling in the 9th and 35th percentiles, respectively.

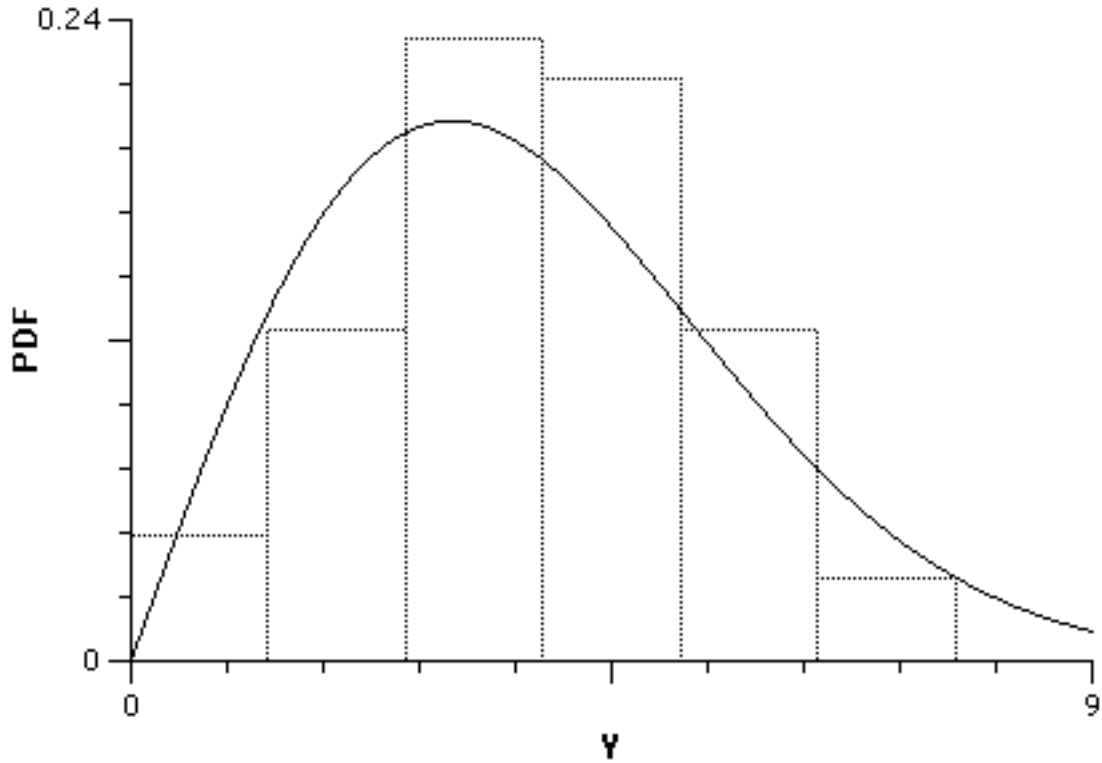


Figure 15. Comparison of 50 Radii to their Theoretical Rayleigh Distribution

Deterministic Models

Consider the data and model shown in Figure 13 and Equation 26. These daytime values were modeled with a sine function having four parameters. The value of R-squared, 0.99908, is excellent and we would have no qualms about accepting such a fit. Still, by using the least-squares technique, we implicitly assumed that the residuals were $\sim \text{Normal}(0, B)$. We did not prove it. Now that we know how to assess a distribution of random errors, let us do so.

Table 8 shows again the daytime data of Table 1, along with the daytime values predicted by our supposedly optimum model. A brief glance at either this table or at Figure 13 discloses that the fit is quite good. The residuals are listed in column 4. The worst one, shown in bold, is only -7.2 minutes. The parameters given in Equation 26 are obviously very good. However, they are the ML parameters if and only if we can prove that the vector of residuals shown here is $\sim \text{Normal}(0, B)$. Otherwise, as good as the fit is, there are better parameters yet to be found, perhaps even a better model.

We shall not repeat earlier discussion. As Sherlock Holmes would say, “You know my methods.” We find that the residuals are best described as $\sim \text{Normal}(0, 4.280)$. In this case, the mean was held constant at zero, not estimated from the data. The likelihood and the K-S statistics fall in the 49th and 82nd percentile, respectively, based upon 1,000 (parametric) bootstrap samples. The unbiased, Gaussian nature of these residuals is, therefore, accepted.

Table 8. Daytime Data, Observed and Modeled

Observed		Modeled	
Daytime (min.)	Day	Daytime (est.)	Residual
545	1	548.7	3.7
595	32	591.8	-3.2
669	60	664.0	-5.0
758	91	760.1	2.1
839	121	845.0	6.0
901	152	900.7	-0.3
915	172	911.7	-3.3
912	182	909.2	-2.8
867	213	868.6	1.6
784	244	789.2	5.2
700	274	696.0	-4.0
616	305	608.8	-7.2
555	335	555.9	0.9
540	356	545.1	5.1
544	366	548.4	4.4
595	397	590.8	-4.2
671	426	665.5	-5.5
760	457	761.7	1.7
840	487	846.3	6.3
902	518	901.3	-0.7
915	538	911.7	-3.3
912	548	908.9	-3.1
865	579	867.6	2.6
782	610	787.7	5.7
698	640	694.4	-3.6
614	671	607.6	-6.4
554	701	555.3	1.3
540	721	545.1	5.1
545	732	548.7	3.7
597	763	591.9	-5.1
671	791	664.0	-7.0
760	822	760.1	0.1
839	852	845.1	6.1
902	883	900.7	-1.3
915	903	911.7	-3.3
912	913	909.2	-2.8
865	944	868.6	3.6
783	975	789.1	6.1
699	1005	695.9	-3.1
615	1036	608.8	-6.2
554	1066	555.8	1.8
540	1086	545.1	5.1
545	1097	548.4	3.4

In saying this, we imply only that the residuals, collectively, could easily have come from a zero-mean Normal distribution. “Collectively” signifies that we do not consider the **order** of the residuals with respect to each other. If you look carefully at Table 8, column 4, you will notice a suspicious alternation of positive and negative residuals. Such a periodicity would **not** be observed in a random sample from any Gaussian distribution. It is, rather, a **systematic error** in our model. It appears that we have not, after all, completely characterized the motions of the Earth and Sun with this simple equation. Kepler could have told us that!

We forego perfection for the moment and take up the matter of uncertainty in the least-squares parameters. To do this, we shall again require a large number of nonparametric bootstrap samples. There are two common methods for obtaining such samples. We could simply resample the data points as we did with the 50 radii of the previous example. However, it is more common to use all of the data points and resample the residuals instead. If the latter are independent (hence, uncorrelated), then any residual might have been found associated with any data point. Here, there is some small amount of correlation but, for the sake of this illustration, we shall ignore it and proceed in the usual fashion, using 1,000 bootstrap samples.

Doing so, we obtain simultaneous confidence intervals for all of the parameters, as follows:

- | | | |
|-------|----------------------------------|-----------------------|
| A | 90% --> [182.247, 184.318] | (cf. Eq. 26, page 37) |
| | 95% --> [181.944, 184.594] | |
| | 99% --> [181.354, 184.991] | |
|
B | 90% --> [0.00273247, 0.00273958] | |
| | 95% --> [0.00273152, 0.00274036] | |
| | 99% --> [0.00272935, 0.00274223] | |
|
C | 90% --> [-1.40549, -1.37757] | |
| | 95% --> [-1.40857, -1.37274] | |
| | 99% --> [-1.41479, -1.36706] | |
|
D | 90% --> [727.679, 729.264] | |
| | 95% --> [727.465, 729.569] | |
| | 99% --> [727.030, 729.939] | |

Note that the widths of these intervals, relative to parameter magnitude, are much narrower (i.e., better) than those seen in the example of the 50 random radii in spite of the fact that the latter were a genuine sample from a known parent population while the sine model here is known to be deficient.

It is difficult to predict the variability of parameters. It was not obvious, for instance, that the uncertainty in the period of these data would be less than the uncertainty in the amplitude. Unless confidence intervals can be computed from theory, only a large bootstrap sample will yield this sort of information.

The interpretation of the confidence intervals is the same with these deterministic data as it was with the stochastic data in the previous example. Also, the accuracy of the intervals will improve, *ceteris paribus*, with bootstrap sample size.

Parameter Correlation

There is one, final comment to make concerning confidence intervals and that concerns their appearance. A confidence interval is just a range of numbers and it is very easy to fall into the trap of believing that all of the numbers in that range are equally probable or, if there is more than one such range for a given model, that any parameter may have any value in the stated range regardless of the values of the other parameters. Neither of these beliefs is valid.

As an illustration, let us examine the nonparametric bootstrap results for the daytime data more closely. Given 1,000 bootstrap samples, we may compute the mean of all the bootstrap parameters and, also, the covariance matrix.

The means are as follows:

A = 183.383
B = 0.00273601
C = -1.39028
D = 728.435

Note, first of all, that these means are not quite equal to the ML values of the parameters. Bootstrap distributions are often unsymmetrical, with the mode and mean unequal.

Here is the upper triangle³² of the covariance matrix, with indices from A to D:

6.82581e-01	2.12136e-08	1.06586e-03	5.41600e-02
	8.03571e-12	-2.29840e-08	4.54181e-07
		1.19938e-04	-6.43826e-04
			4.14830e-01

The ij^{th} entry in this matrix is given by Equation 28:

$$\text{Cov}(P_{ij}) \equiv \sigma_{ij}^2 = \left\langle \left(P_i - \langle P_i \rangle \right) \left(P_j - \langle P_j \rangle \right) \right\rangle \quad 28.$$

where P is a parameter.

When $i = j$, this is just the variance of parameter[j]; when $i \neq j$, it is the *covariance*. As we have seen before, the variance measures the intrinsic variability of P . The covariance measures the variability of one parameter with respect to another. Looking at this matrix, we see that parameter B, the period, has the smallest variance. No matter what the other parameters might have been, for any of the 1,000 bootstrap samples, the period stayed almost constant.

The sign of the covariance is an indication of the sense of correlation. Covariances that are positive indicate that the two parameters in question increased and decreased together, on average. Negative covariances mean that, on average, one parameter increased when the other

³² All covariance matrices are symmetric about the main diagonal.

decreased. Here, for example, the amplitude of the sine wave, A , is positively correlated with all of the remaining parameters. However, the phase, C , is negatively correlated with D . To get an idea of how strong these relationships are, it is, perhaps, best simply to compare the covariances for any parameter to its own variance. Any statistics textbook will discuss these matters in great detail.

In any case, the implications with regard to confidence intervals is clear:

All combinations of parameters are not equally probable.

Summary

Our goal was to establish a quantitative measure of the variability of model parameters. We have succeeded by taking advantage of the power of the nonparametric bootstrap. The measure of variability obtained is a confidence interval which denotes the numerical range wherein the true value of the parameter is likely to be found with some specified probability. By using information from the covariance matrix, we gain additional insight into the manner in which parameters change with respect to each other when the dataset changes.

99.9999999999% PERFECT!

The Rydberg constant describes the colors of atoms. For a hypothetical atom of infinite mass, its limiting value, $R_\infty = 109,737.3156864 \text{ cm}^{-1}$, has an accuracy of more than twelve significant figures [UDE97]. Think about that for a moment.

That's like measuring the average distance between Earth and the Moon with an error less than the width of the period at the end of this sentence. Clearly, this is far beyond our current technology. How, then, can we measure R_∞ with such perfection?

Kepler, and his eponymous laws of planetary motion are partly responsible. Using the first and third of these, it is possible to derive the classical expression for the energy of an electron in an atom. If this expression is combined with the laws of Quantum Mechanics (QM) and its insistence upon the importance of integers, the result is that an atom cannot have just any color but only a limited (yet still infinite!) set of colors.

Evidence for this conclusion can be illustrated using Hydrogen, the simplest atom, with the simplest set of colors. The primary colors of Hydrogen are listed in Table 9.³³

Table 9. Observed Transitions/Colors for Hydrogen (1H^1)

N_u	N_l Series	Wavelength (in Angstroms)					
		1 Lyman <i>Ultraviolet</i>	2 Balmer <i>Visible</i>	3 Paschen	4 Brackett <i>Infra-red</i>	5 Pfund	6 Humphries
2		1215.67	6562.72	18751.0	40511.6	74578.0	123685.
3		1025.72	4861.33	12818.1	26251.5	46525.1	
4		972.54	4340.47	10938.1	21655.3		
5		949.74	4101.74	10049.4			
6		937.80	3970.07	9546.0			
7		930.75	3889.05				
8		926.23	3835.38				
9		923.15	3797.90				

The Rydberg constant describes all of these colors *via* the model given in Equation 29.

$$\text{Transition energy} \sim R_H \left(\frac{1}{N_l^2} - \frac{1}{N_u^2} \right) \quad 29.$$

where N_l and N_u are positive integers and where \sim , in this case, means “is proportional to.”

According to the laws of QM, the color, i.e., the wavelength (λ) of the light emitted during any transition is inversely proportional to its energy. The final model, based on these laws, is

³³ see file *Examples:Rydberg.in* [WEA75]

given by Equation 30. Fitting the data in Table 9 to Equation 30, the (unweighted) least-squares value for R_H is $109,708 \text{ cm}^{-1}$.³⁴ The resulting graph is shown in Figure 16, with Y representing wavelength. To six significant figures, $R\text{-squared} = 1.00000$. When experiments along these lines are done very carefully, far more than six significant figures can be achieved. However this is still not enough to get the reported precision of R_∞ .

$$\lambda = \frac{x}{R_H} = \frac{1}{R_H} \frac{N_u^2 N_l^2}{N_u^2 - N_l^2} \quad 30.$$

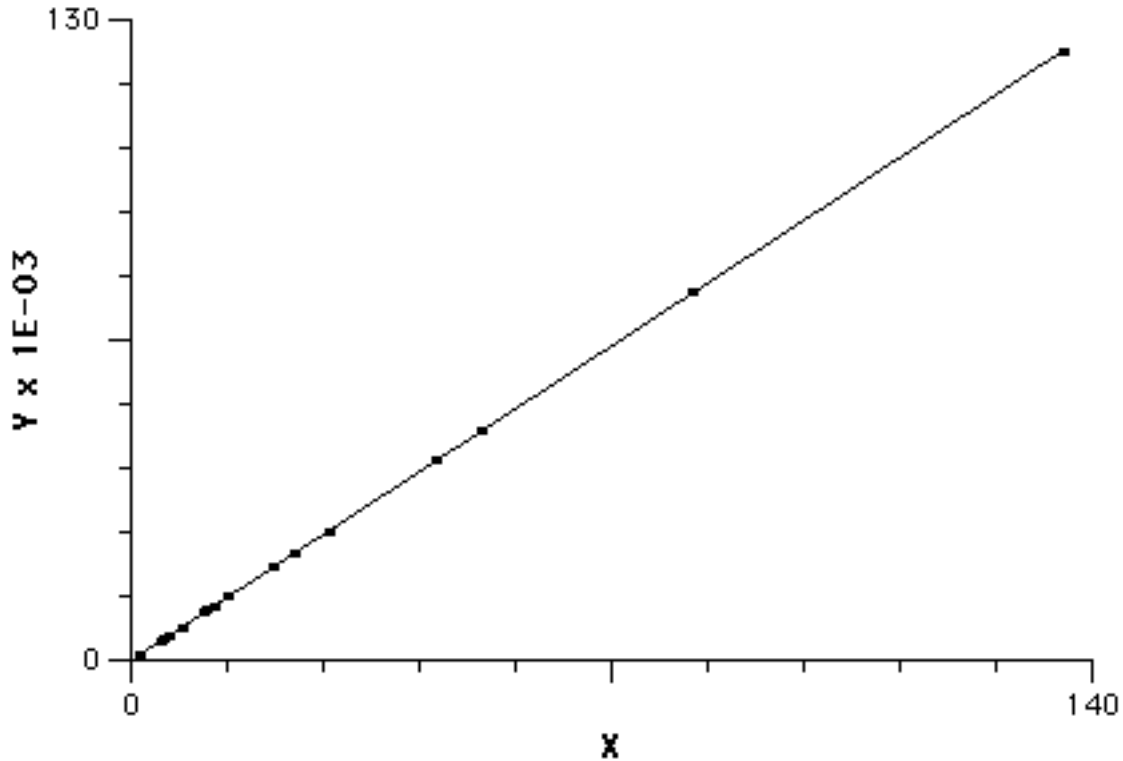


Figure 16. Rydberg Data for Hydrogen

The current accuracy of the Rydberg constant derives not from any single experiment or insight into the laws of Nature but from the unavoidable requirement that the Universe be consistent. This consistency must be mirrored in the physical sciences, specifically in the many scientific models that have been developed over the last four centuries. When all of these models are considered simultaneously, they must agree.

This requirement is manifested in relationships such as Equation 31.

³⁴ One centimeter = 10^8 Angstroms.

$$R_{\infty} = \frac{2 \pi^2 m e^4}{c h^3} \quad 31.$$

where m and e are the mass and charge of the electron, respectively, c is the speed of light, and h is Planck's constant, the proportionality constant connecting energy to wavelength.

Equation 30 relates a mass, a charge, the speed of light, and several other constants, to the observable colors of atoms. One can perform a variety of experiments to determine this mass, charge, or speed. In every case, there are still more factors, determinable in yet different ways. Each of these experiments depends upon a model and all of these models interlock. They are like words in a crossword puzzle. The total solution will break down if even one of them is incorrect.

The physical sciences, which seek this total solution, are not incorrect; they are mature and reliable. One need only turn on a television set or view a CAT scan to appreciate that the laws of the Universe are very well understood. Except at the frontiers of knowledge, the puzzle all holds together. It does not break down.

It is, in part, this internal consistency which affords the degree of accuracy that is seen in R_{∞} . Whenever any physical experiment is performed, it yields a quantitative answer with some precision. If sufficient attention has been paid to eliminating sources of error, the experiment will have high precision. When many such experiments are incorporated into a single analysis, it is possible to compute a vector of means for any quantities, such as those in Equation 30, that their answers might have in common. As we have seen, in Equation 24 for instance, these means will be more precise than any single experimental result. Simply put, the information from many experiments necessarily exceeds the information provided by just one of them.

In our efforts at modeling, we have sought to extract information from data. The examples described herein illustrate that contemporary methodology is very effective in this task. When the same methodology is applied to a collection of experiments, it is possible to achieve enough synergy to eliminate almost all of the residual error. It is possible to obtain 99.999999999 percent accuracy.

Twelve significant figures do not come about by accident. The techniques described in this tutorial are just the latest in a heritage that goes all the way back to Kepler and Galileo. That heritage is one which affirms the primacy of data as well as the power of mathematics and statistics. These affirmations are implemented in techniques that are based upon the concept of maximum likelihood, techniques that literally construct the best possible match between theory and experiment, between model and data.

In computing the optimum result, these same techniques explicitly acknowledge the role of measurement error and sampling error. Even the best measurements are not perfect. Even the best experiments must examine less than the entire Universe.

Error implies that the models we construct, and their parameters, must always be somewhat uncertain, however much we might like to believe otherwise. No model is complete until those uncertainties have been rigorously quantified. A good model knows its limitations...

...as does a good tutorial.

We began with data, but where do we end? Perhaps T. S. Eliot had the right idea:

“We shall not cease from exploration
And the end of all our exploring
Will be to arrive where we started
And know the place for the first time.”

And be ready to start again on a new adventure, older and wiser, and better prepared—
to experiment,
to analyze,
to play the game one more time.

Users' Guide

WELCOME to *Regress+*... and Congratulations on choosing the best software for mathematical modeling available anywhere! *Regress+* (“Regress_plus”) is a user-friendly software tool to facilitate the statistical analysis of quantitative data, construct (univariate) mathematical models, and quantify the uncertainties of any model and its parameters in a rigorous fashion. Models may be deterministic or stochastic, depending upon the nature of the data. In the first case, the model will be an equation, $y = f(x)$; in the second case, the model will be a distribution, $f(y)$.

Here is an abbreviated list of what you can expect *Regress+* to do:

- Read in some data, written by any application capable of producing a textfile
- Fit these data to any model you select, from a large menu of choices
- Let you type in your own equation if the one you want is unavailable
- Allow you to choose the optimization criterion
- Consider from one to ten parameters
- Start with your own estimates of the values for these parameters
- Permit you to stipulate that some of these values are definitely correct
- Quantify the goodness-of-fit of the model, using two independent metrics
- Determine confidence intervals, and the covariance matrix, for all parameters
- Generate a plot of the data and model (with one keystroke!)
- Save the results, including a PICT file of the plot
- Create a file of random variates from any distribution you choose
- Run unattended, if necessary
- Do all of this, even while other applications are running at the same time

Unless you are already an expert in this field, the chances are that *Regress+* will do one more thing. It will teach you more about modeling than you knew before. That's a plus, no matter how your models turn out.

INSTALLATION

This Users' Guide assumes that you already know Macintosh® basics—how to make an alias, use the Standard File Dialog, etc.

As with everything else in this software package, installation has been designed with ease-of-use in mind.

1. Drag the *Regress+* folder to the folder, on your hard drive, where you keep such applications.
2. For maximum convenience, make an alias of the *Regress+* application and place it on your desktop so that you can drag-and-drop any input file onto it.
3. If you have not already done so, READ THE TUTORIAL. This Guide tells you what buttons to push and when, but not why.

USING *REGRESS+*

There are many options available in *Regress+* and it would be tedious just to describe them one by one. Besides, not many people enjoy reading documentation, so we shall assume that you have just installed this software, have the data files in the *Examples* folder, and can't wait to try them out. Since it is the more familiar, this section will start with deterministic modeling, a.k.a. regression.

This section is intended only to get you started. For all the fine print, consult the *Reference* section.

Deterministic Modeling

Before we get too fancy, let's take a very simple example, one that you could do even with a calculator. There is a famous hypothesis in astronomy, known as Bode's Law, which is supposed to predict the distances of the planets from the Sun. It isn't actually a law, and it wasn't even discovered by anyone named Bode, but we'll try it anyway.

Bode's Law directs that you create a series by starting with zero and three, then doubling thereafter. By adding four to each of the numbers in this series, then dividing by ten, you get the independent variable, x , supposedly related to the average planetary distance, y (AU), by the simplest of linear models (Equation 32). This dataset is listed in Table 10 [ABE64].

$$y = A x + B \quad 32.$$

Table 10. Bode's Law of Planetary Distance

Planet	N	Distance (y)	x	y (est.)	Residual
Mercury	0	0.387	0.4	2.659	+2.272
Venus	3	0.723	0.7	2.822	+2.099
Earth	6	1.000	1.0	2.985	+1.985
Mars	12	1.524	1.6	3.312	+1.788
(Ceres)	24	2.767	2.8	3.965	+1.198
Jupiter	48	5.203	5.2	5.272	+0.069
Saturn	96	9.555	10.0	7.884	-1.671
Uranus	192	19.218	19.6	13.110	-6.108
Neptune	384	30.110	38.8	23.561	-6.549 *
Pluto	768	39.545	77.2	44.462	+4.917

Input File

Open the *Examples* folder and locate the file *Bode.in*. If you look at this file with any text editor, you will see that it is just a plain, tab-delimited ASCII file. The tabs are not required; they are used for convenience, so that you can open the file with a spreadsheet and the data will be lined up in columns. The semicolons appearing in the file introduce comments and full-line comments usually start with a semicolon. Column 1 **must** be the dependent variable, y , and column 2 **must** be the independent variable, x , the same order as in the formula $y = f(x)$. If there are weights for the dependent variable, then they must be included as a third column. The

extension *.in* (case-sensitive) tells *Regress+* that this is an input file. Without this extension, a textfile will not appear in the input file dialog.

Drag-and-drop this input file onto *Regress+* (or its alias) and you will immediately see the large **Setup** dialog, shown in Figure 17.³⁵

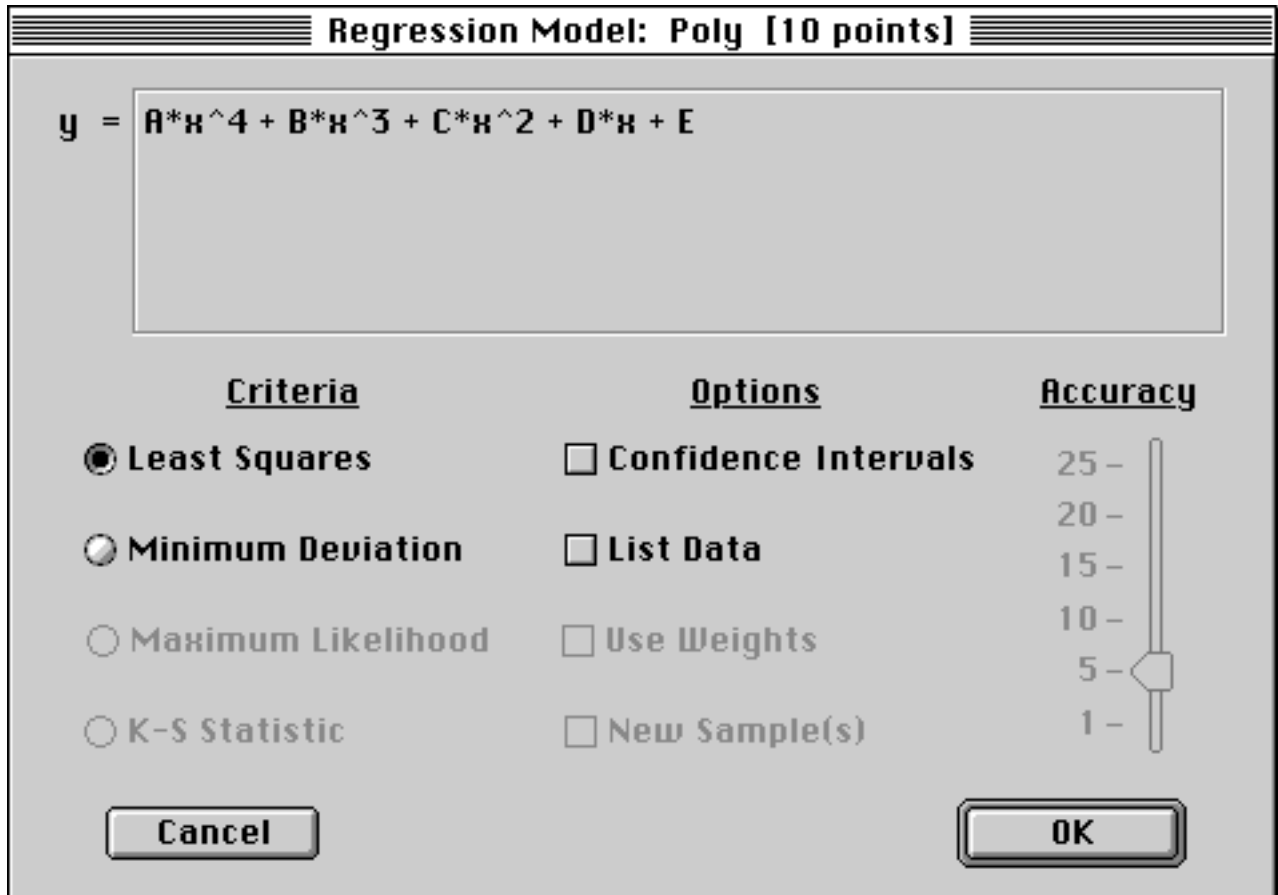


Figure 17. Setup Dialog

Setting up the Model

Let's go through the items in this dialog box.

The title of the dialog indicates that

- this is a regression model
- the (default) model is a polynomial equation (Poly)
- there are 10 points in the input file

³⁵ All screenshots depicted in this Guide were captured using 4-bit grayscale.

The displayed form of Poly is shown in the highlighted box at the top of the dialog. This model subsumes the one shown in Equation 31 as a special case. If $A = B = C = 0$, then we are left with the slope, D, and intercept, E, of a standard linear model. Obviously, parameter labels are arbitrary and D, E will do just as nicely as A, B. Forcing A-C to be zero is trivial, as we shall soon see.

Just below the model box, there are three columns of Setup variations. With deterministic modeling, they are as follows:

- **Criteria**

You have your choice of two optimization criteria. The most common, by far, is Least Squares (default). The alternative is Minimum Deviation. Here, as elsewhere, choices that are ghosted out are either not applicable or impermissible under the circumstances.

- **Options**

All Options are OFF by default.

Confidence Intervals (90-, 95-, and 99-percent) apply only to model parameters. List Data will generate a second output file, listing the input along with predicted y-values and residuals (predicted – observed), with an asterisk identifying the worst residual. Use Weights applies to regressions but is ghosted out in this example since the input file did not include a (third) column of weights.

- **Accuracy**

Accuracy is disabled by default.

Accuracy applies to the bootstrapped confidence intervals. The number of bootstrap samples equals $200 \times \text{Accuracy}$.³⁶

The two buttons, OK and Cancel, have their usual meanings. At times, a button may also be ghosted out, for the reasons cited above.

Not all of the Setup choices are in the dialog box. This dialog is also connected to the **Equations** menu. This menu lists all of the built-in regression models. For now, we shall stick with our linear model.

To get all of the output discussed in this example, check List Data.

Accepting the Setup

After you have chosen the desired model, options, and (if appropriate) Accuracy, click the OK button (or hit Return or Enter). The Setup dialog will be replaced by the Parameter dialog. In this example, it will look like Figure 18.³⁷

³⁶ unless the bootstrap phase is *Canceled* prematurely

³⁷ If the model has fewer than five parameters, there will be fewer entries in the list.

Starting Parameters

$$y = A \cdot x^4 + B \cdot x^3 + C \cdot x^2 + D \cdot x + E$$

Parameter	Constant
A = 1.00000E+00	<input type="checkbox"/>
B = 1.00000E+00	<input type="checkbox"/>
C = 1.00000E+00	<input type="checkbox"/>
D = 1.00000E+00	<input type="checkbox"/>
E = 1.00000E+00	<input type="checkbox"/>

Cancel OK

Figure 18. Parameter Dialog

Initializing the Parameters

The Parameter Dialog records your “guesses” for the parameter values. This parameter list is a “Tab group” and the Tab key will rotate among them, from top to bottom. You can cut-and-paste but do **not** hit Return before you have finished with all of them.

In general, you will be using your own data and a model that you have thought about. Therefore, you should have at least a rough idea of the correct parameter values. As described more fully in the section *Technical Details*, *Regress+* achieves an optimization by converging to a **local optimum** that is determined by the initial parameter values and the (unknown) basin of attraction. In fact, what you really seek is the **global optimum**. In the general case, there will be more than one optimum and the one you get may not be the one you want. In very simple cases, with only one or two parameters, there is usually only one optimum. If you know your data and your model, as you should, then your starting parameter estimates should be close enough to optimal to enable *Regress+* to converge on the global optimum very easily.

The default value for all parameters is 1.0. Since we want parameters A-C to remain zero, enter zero **and click Constant** for each of them. When a parameter is stipulated as Constant, it is the same thing as inserting a number into the equation instead of a parameter. By making A-C Constant, we are left with just two parameters, D and E. Allow D and E to keep their default initial values.

Accepting the Initial Parameter Values

Were you to click Cancel at this point, you would return to the Setup dialog. However, assume that you are satisfied with both the model and initial estimates of the parameter values and just click the OK button (or hit Return or Enter). The Parameter dialog will be replaced by the Display dialog, shown in Figure 19.

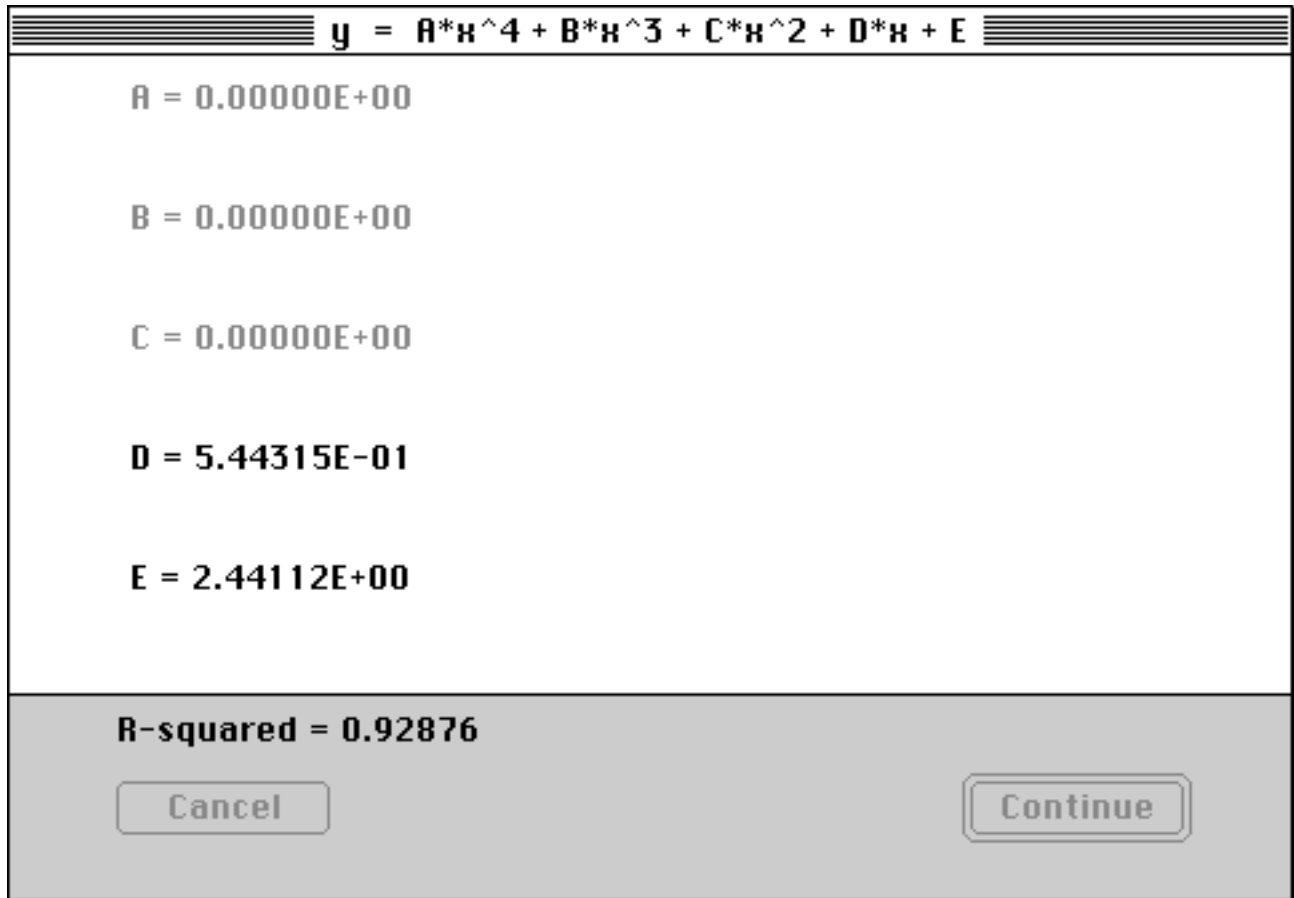


Figure 19. Display Dialog

Figure 19 shows parameters A-C ghosted out, indicating their Constant status. The values shown for D and E are the optimum values for those parameters, according to the least-squares criterion. At the bottom of the dialog is the value for R-squared. Finally, the two buttons are disabled because, given the Setup, there is nothing left to do from this dialog.

However, there are two actions left, if you choose. The first is to Save or SaveAs... the results. If you Save, then an output textfile will be placed in the default folder with the default filename *Bode.Ply.LS.out*. This filename is constructed from the input file prefix, the model, and the criterion. The suffix is always *.out*. If, instead, you SaveAs..., you will get to choose the filename and the default folder.³⁸ For this example, select Save from the File menu, or hit

³⁸ The default folder is determined, in part, by your Finder preferences.

Command-S, to save the output file. Since the List Data option was selected during Setup, a tab-delimited *.list* file will be saved at the same time.

The second action you might wish to take is to look at a plot of the data and model together. **While the Display dialog is showing**, the Displays menu will be enabled. Create a graph by selecting the Graph Show/Hide menu item (or hit Command-G). This menu item is a toggle and will alternate between showing and hiding the graph. In this case, the graph appears as shown in Figure 20.

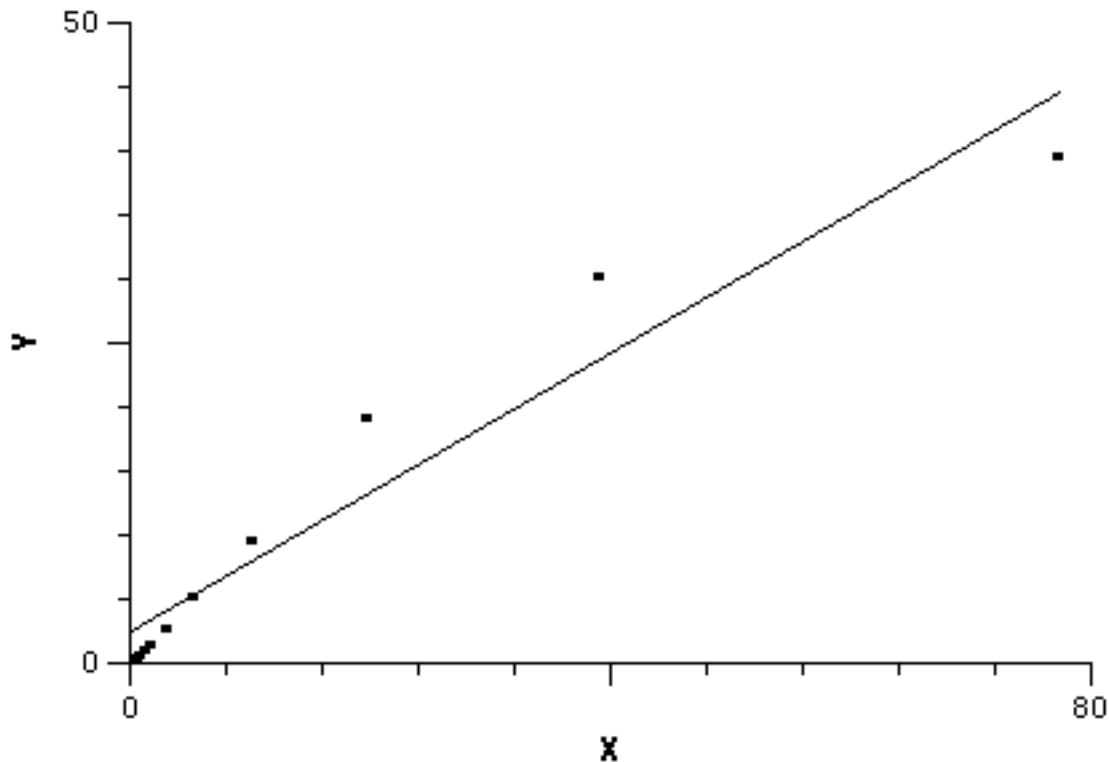


Figure 20. Graph of Bode Data and Linear Model

If you have read the Tutorial, then the format of this graph will look familiar. The abscissa is always labeled *X* and the ordinate *Y*. The scaling is automatic and you cannot change it. In return for this rigidity, you get a graph as easily as one could wish, just a single keystroke.

If the Graph window is frontmost (active), the Displays menus will enable the Save as PICT... menu item. Selecting this will bring up the SaveAs... dialog and enable you to save the graph to a black-and-white PICT file. This PICT may then be cut-and-pasted into any document using the global Clipboard. This is exactly what was done with Figure 20.

Whenever a PICT is pasted into a document, e.g., with a word processor, you may **crop** the whitespace around the edges without damaging its appearance. However, if you **resize** it, the fonts will be degraded. *Regress+* PICTs are sized to fit on any standard page with standard margins, as shown.

Output Files

Output files are tab-delimited textfiles and can be viewed using any text editor, such as *SimpleText*TM. They appear best when displayed/printed in a monospaced font of size 10, e.g., Courier 10.

Look at file *Bode.Ply.LS.out*. Much of what appears is simply a repetition of the input. The parameters are again listed, in this case with the word “constant” after A-C. The Summary Statistics should also look familiar except for the last one. The standard-error-of-estimate, which measures the average error of the model, is defined in Equation 33.

$$\text{Std. Error of Estimate} = \sqrt{\frac{\text{ESS}}{(\# \text{ points} - \# \text{ parameters})}} \quad 33.$$

where ESS is the Error-sum-of-squares and # parameters does **not** include any held Constant.

Here, the standard-error-of-estimate is about four which, looking at Figure 20, appears to be a reasonable value for a “typical” prediction error.

The file *Bode.Ply.LS.list* contains the data that were used to create Table 10. Table 10, column 5 lists the values for y that were predicted (estimated) by this optimized model. Table 10, column 6 lists the signed residuals. The worst prediction, for Neptune, is marked by an asterisk.

Before we leave this example, note that Figure 20 suggests strongly that a quadratic model might be a lot better than a linear model. Adding the extra parameter guarantees, of course, that it will not be any worse. [The optimal quadratic could always make the quadratic term equal to zero and thus reproduce the linear model.] You should now have enough information to try out this quadratic on your own.

One thing that *Regress+* does not do is explicit *stepwise regression*. That is, it does not carry out all of the statistical tests necessary to determine whether or not an additional parameter gives **significantly** better results. This is usually easy to do only with a generalized linear model, such as a polynomial.

Where *Regress+* really shines is with *nonlinear* models. Let us try one of those.

A Nonlinear Model

In the context of regression, the term “nonlinear” refers to the unknowns in the problem. Once an analytical form for the model has been chosen, the only unknowns are its parameters. As an example of a nonlinear model, consider again the Daytime data discussed in the Tutorial. Drop the file *Daytime.in* onto *Regress+*.

Select the sine (Sin) model from the Equations menu and check Confidence Intervals in the Setup dialog. Hit Return.

In this nonlinear example, the parameters appear in places that no finite polynomial could reproduce exactly. A sine function is transcendental and only an infinite number of terms will suffice. Consequently, there is no simple matrix computation that will optimize this model. Not only that but, due to the periodic nature of trigonometric equations, there are a number of spurious optima in this problem and you must make good starting estimates for the parameters if you want to converge to the best solution. However, this is easy to do, assuming that you understand the data and the model.

Parameter A

This is the *amplitude* (half-height) of the sine wave. Looking at Table 1 (pg. 3), at the Daytimes for 6/21 and 12/22, a good guess would be 187.5 minutes.

Parameter B

This is the *frequency*, the reciprocal of the period, which should be about 1/365 days = 0.0027 per day.

Parameter C

This one is tricky. It is the *phase* of the sine wave, indicating the offset along the abscissa. A sine wave begins by rising from its average. With these data, that occurs at the vernal equinox (about 3/23). This dataset, on the other hand, starts at 1/1 which is about a quarter cycle **before** the start of the sine wave. Therefore, the phase should be about $-\pi/2 = -1.6$. The phase is dimensionless.

Parameter D

Finally, parameter D represents the average Daytime which is roughly the length of the day at an equinox. Using the average of solstice values, this is approximately 727 minutes.

Accept these initial values (hit Return).

The optimum values are those given in Equation 26 (pg. 37). As R-squared indicates, it is an excellent fit. Check out the graph.

Since we opted for Confidence Intervals during Setup, the Continue button is now enabled, indicating that the modeling is not finished yet. Click Continue.

The first thing you see is a message that *Regress+* is Initializing the process for computing confidence intervals. The latter involves a nonparametric bootstrap and it takes a bit of setting up. A progress bar indicates the amount of time remaining in this phase. In general, initialization time is proportional to the number of parameters and the number of points.

After this initialization phase, the bootstrap sampling will begin. Since we did not change the default Accuracy, this quantity will be equal to five, meaning that there will be 1,000 nonparametric bootstrap samples in this analysis. After a few minutes, depending on the speed of your computer, the Display dialog will look something like Figure 21.

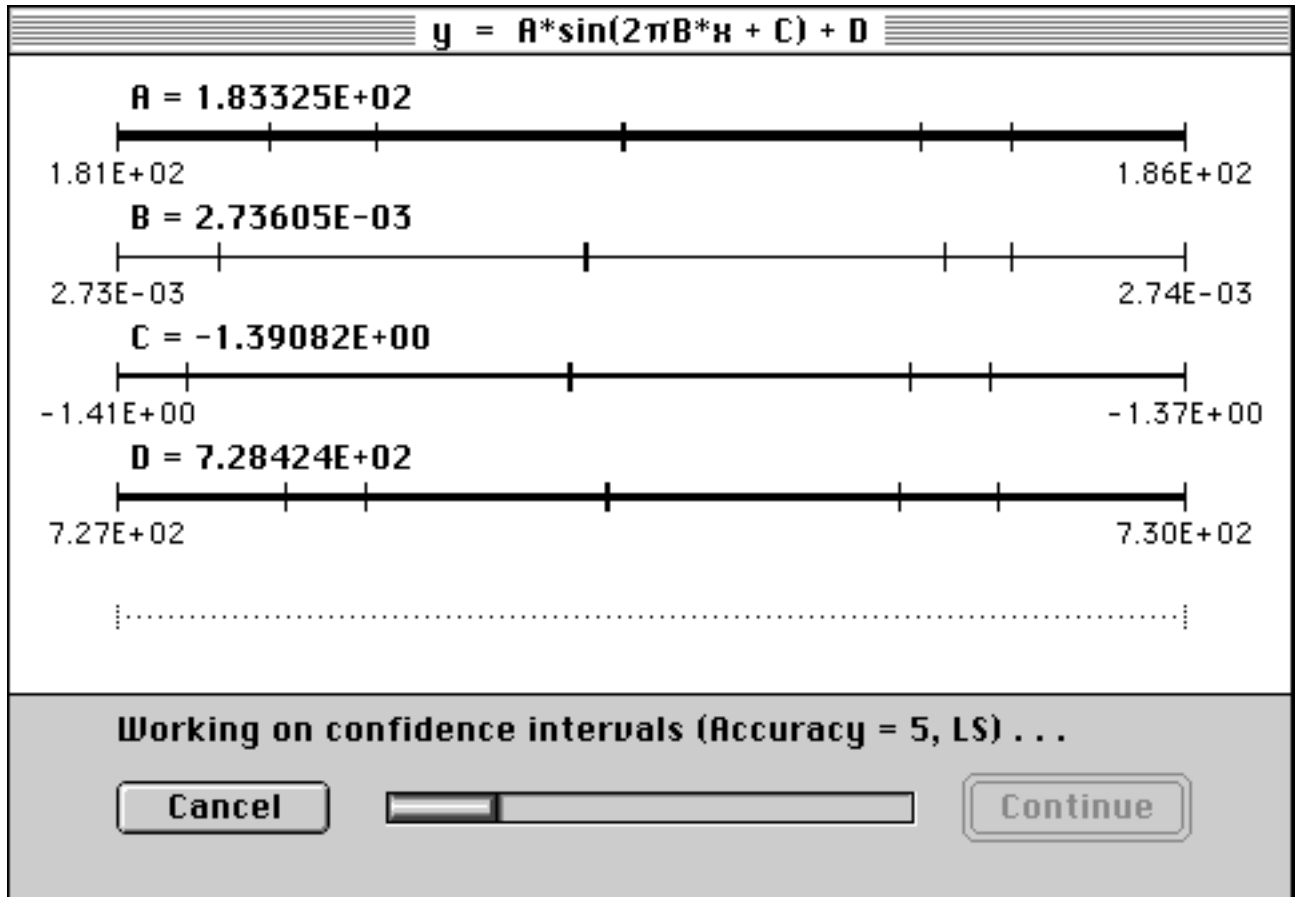


Figure 21. Working on Confidence Intervals

It is not likely that your run will ever look **exactly** like Figure 21 because the bootstrap is a stochastic process and seldom proceeds twice in precisely the same way.

The items in this figure deserve some comment. First, the optimum parameters are shown in bold. Just below each, there is a bar of varying thickness. Near the middle of a bar, a thick vertical marker shows the position of the optimum. Usually, its thin counterparts, on either side, will eventually occur in pairs, with the inner pair denoting the 90-percent confidence interval, the middle pair the 95th, and the outer pair the 99th.

Often, there will be fewer than six confidence limits showing. For instance, Figure 21 displays estimates for all three confidence intervals for parameters A and D but not for parameters B and C. Both of these are missing the lower limit for their 99-percent confidence intervals. This means that the number of bootstrap samples examined so far is insufficient to reach such an extreme value. If the analysis ends without the confidence interval you require, your only recourse is to Restart (File menu) and use a larger Accuracy. Occasionally, the bootstrap distribution is so skewed that even 5,000 bootstrap samples (the maximum) is not enough to obtain all three confidence intervals for all parameters. However, this seldom happens when the model is acceptable.

When these horizontal bars appear, they will always be bounded by numerical limits. If any of these limits is uncertain, due to an insufficient number of bootstrap samples, then it will appear in *italics* in the Display dialog. Any confidence limit that is still uncertain when the analysis is finished will appear in the output file next to a paren instead of a square bracket. The value reported, in such cases, is just the appropriate extremum in the empirical bootstrap distribution. It should not be trusted as a genuine confidence limit.

Since the bootstrap is stochastic, the confidence intervals it estimates are not perfectly reproducible. However, they get better with increasing Accuracy. Table 11 shows how the confidence intervals for the amplitude, A, vary with Accuracy. You could perform the same experiment, with similar results.

Table 11. Daytime Model—Amplitude Confidence Intervals

Accuracy	% Confidence	Lower Limit	Upper Limit
1	90	182.069	184.423
	95	181.749	184.842
	99	181.307	185.261
5	90	182.247	184.318
	95	181.944	184.594
	99	181.354	184.991
10	90	182.233	184.361
	95	181.931	184.630
	99	181.228	185.347
15	90	182.149	184.325
	95	181.858	184.637
	99	181.281	185.230
20	90	182.262	184.485
	95	181.956	184.802
	99	181.353	185.389
25	90	182.258	184.412
	95	181.960	184.754
	99	181.432	185.274

The amplitude was chosen for this experiment because it is the most variable of the four parameters. This is indicated in Figure 21 by the thickness of the horizontal bar associated with each parameter. The thickest bar is the most variable and the thinnest is the least variable. Note that this is just an order metric; the variability is not proportional to thickness.

Finally, the default Accuracy is five. This is the smallest Accuracy that you should use if you are going to report the results. The Setup dialog permits a value of one but this is useful only for a cursory examination of the results. You should never use it for any serious work.

Transformations?

Regress+ takes great pride in achieving all of its results without using any transformations, or any approximations other than those normally associated with bootstrap sampling. On the other hand, in the general literature, a **lot** of use is made of transformations. The reason is that nonlinear models are not mathematically tractable and, most often, only a transformation to a linear model will yield any answer at all. Although this is common practice, it is usually a very bad idea. Once you transform the dependent variable in a nonlinear fashion, you do the same to any **errors** in that variable. Consequently, if you then carry out a least-squares analysis, for example, you are no longer minimizing ESS but some nonlinear function of ESS. Even though the original model and the transformed model may be algebraically equivalent, they will not be statistically equivalent **unless all the points fall exactly on the model curve!** In other words, the transformed model will give the ML result only when ESS is zero.

Let us look at a simple example. One of the most common transformations is to take logs of both sides of an equation. This is done by all pocket calculators, for instance, whenever they try to compute the least-squares result for an exponential model (Equation 34).

$$y = A \exp(B x) \quad 34.$$

Taking the logarithm of this equation gives a model expressing $\log(y)$ as a linear function of x . To test the equivalence, file *Transform_test.in* was created. This is a synthetic dataset. The recipe for its construction is given in Equation 35.

$$y = 15 \exp(-0.03 x) + \varepsilon \sim \text{Normal}(0, 0.1) \quad 35.$$

The random errors are genuine zero-mean Gaussian errors, so least-squares will give the true ML result in this case.

Drop this file onto *Regress+*. Use the exponential (Expo) model and make parameters B, C, and E constant at zero (so that they will not be used). You will get an excellent result, with the value of R-squared = 0.99887. The ML parameters are A = 14.9047 and D = -0.0296642. These are very close to the true answers. The graph is shown in Figure 22.

Good models make good predictions. In this case, the standard-error-of-estimate = 0.12.

For comparison, use the file *Transformed.in*. This is the same dataset, after taking natural logs of both sides. Repeat the previous steps but using the model shown in Equation 31.

This time R-squared = 0.98347, seemingly a **worse** fit, even though the data have not changed! The fitted parameters are D = B = -0.0274139 and E = $\log_e(A)$ = 2.59645.

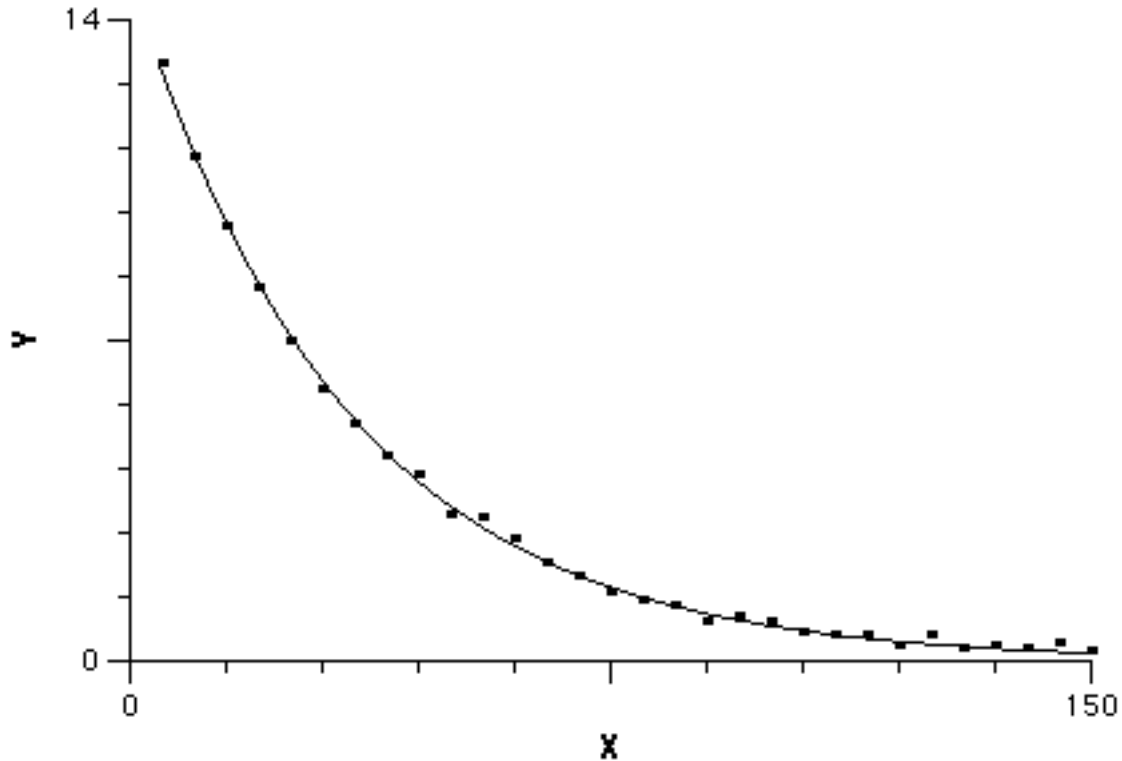


Figure 22. True Exponential Model

In the transformed model, the interpretation of B (D) is unchanged but parameter E is now the natural logarithm of our previous value for A. Thus, $A = \exp(E) = 13.4160$. As you can see, both A and B are worse estimates than before (cf. Eq. 35). Moreover, the transformed model gives a standard-error-of-estimate = 0.159205. However, this too is the logarithm of the actual standard error. Again, taking the antilog, we find that the actual standard error for this model is 1.2, ten times the error of the real model! The graph of this linearized model is shown in Figure 23.

Is it ever advisable to take logs of both sides, or some other nonlinear transformation? Of course, provided that you know exactly what you are doing and can interpret the transformed results correctly. For instance, many exponential models are used with datasets that span a large number of orders of magnitude. Taking logs of both sides might be appropriate in such a case. Still, the log function will compress large values more than small ones and will skew the model accordingly, just as Figure 23 suggests.

Other kinds of transformations may have different effects on the errors and their impact on the model is not always easy to predict, nor is it easy to remove the effects of a transformation after the fact.

In case you were wondering, **linear** transforms don't matter. A linear transformation of an equation consists in multiplying both sides by a constant and/or adding a constant to both sides. This amounts to nothing more than a change of units (e.g., Centigrade to Fahrenheit)

and will have no effect on the model, although it will change the parameters. Remember, units are **very** important.

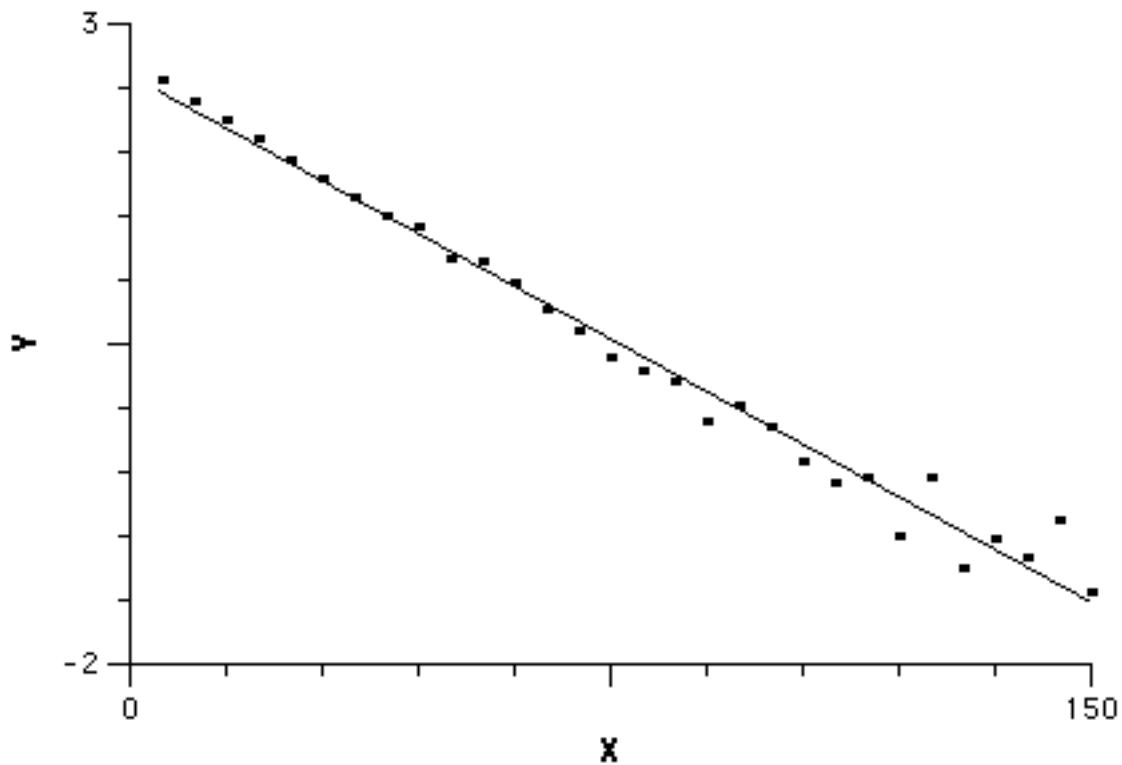


Figure 23. Log-transformed Model

Weighted Regression

If you are fortunate enough to know the standard deviation of your measurements at each value of x , and know that they are not all the same, then you will get a much more accurate model by doing a **weighted** regression. In *Regress+*, the weights are input as a third column vector. They must, of course, all be greater than zero, although it is not essential that they represent one sigma; any constant multiple (> 0) of one sigma will suffice (see Equation 20, pg. 29).

An example of such variable precision was depicted in Figure 2 (pg. 6). To see how much difference knowing the actual weights can make, let us first pretend that the precision at all of these seven points is the same and do a non-weighted regression. The data can be found in the file *Hale_Bopp.CN.in*. Use the model given in Equation 34 with the List Data option. Use Least Squares **without** weights (now an option). The results are as follows:

$$\begin{aligned} A &= 2,763.39 \\ B &= -0.978253 \\ R\text{-squared} &= 0.82936 \end{aligned}$$

Presumably, this exponential model explains only 83 percent of the information in the data. For scientific data, this is not a very impressive result.

The graph of this unweighted model is shown in Figure 24.

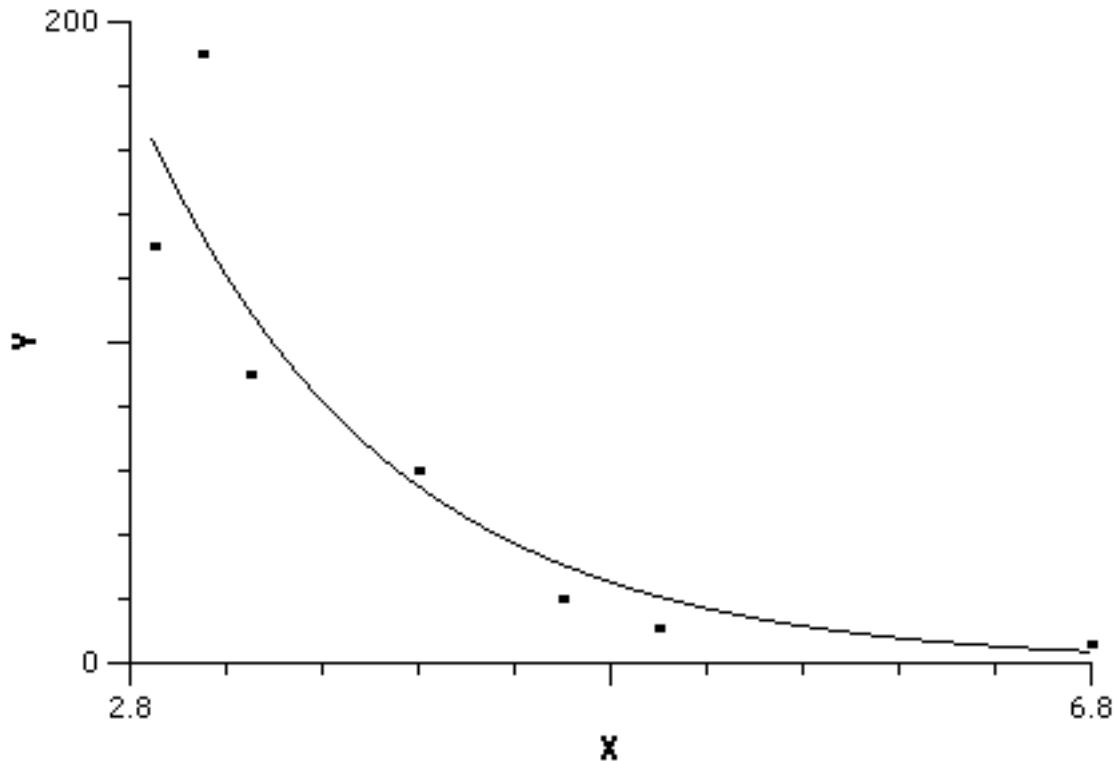


Figure 24. Hale_Bopp Data -- Without Weights

Restart (see File menu) and Use Weights this time. The new results are as follows:

$$A = 2,926.11$$

$$B = -1.04642$$

$$R\text{-squared} = 0.99354$$

The latter parameters are similar to the former set but the value for R-squared is now much better than before. Given this R-squared value, we would have no hesitation in stating that the correct model is exponential. With the unweighted regression, we would be much less certain about making such a claim.

The graph of this weighted model is shown in Figure 25. The error bars are an option, found in the Displays menu.

Notice, in Figure 25, that the model does not try very hard to reproduce the y-value at the point $x = 3.1$ AU. This point has the largest uncertainty and, therefore, the least credibility.

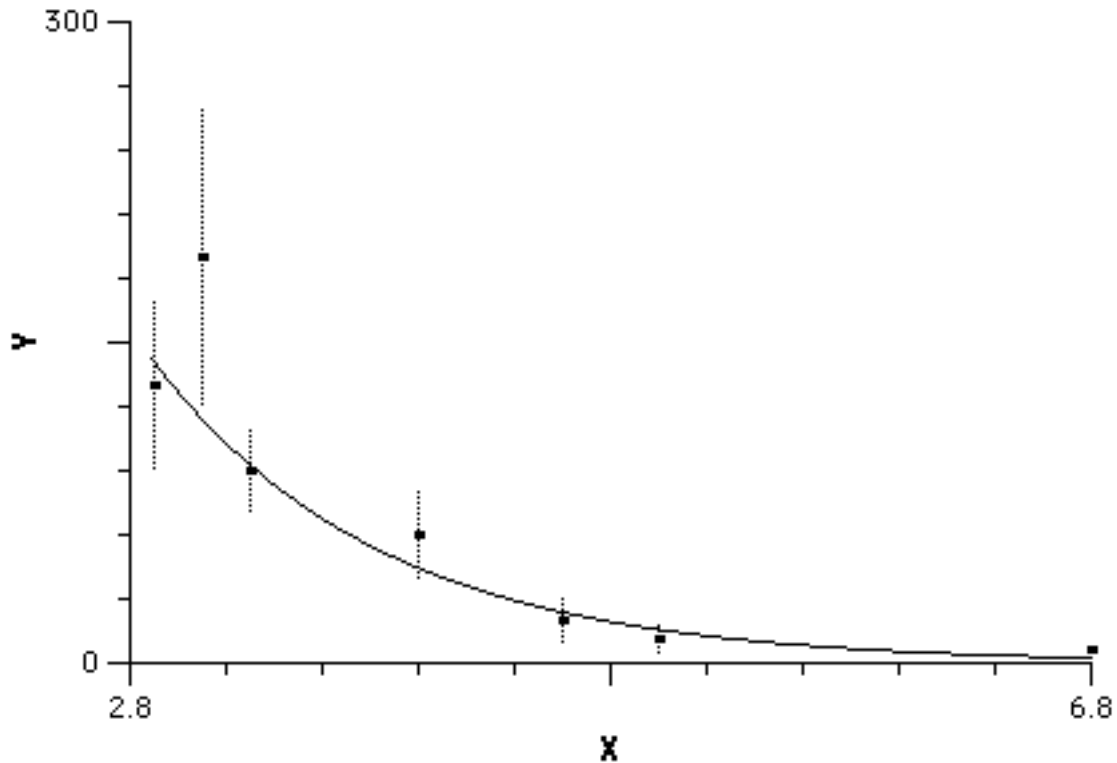


Figure 25. Hale_Bopp Data -- With Weights

Another indication of the value of the weights can be seen in the residuals. In the unweighted regression, the worst residual is at 3.1 AU; in the weighted regression, it is now at 6.8 AU. Also, in the weighted case, the worst residual is 9 times larger than the best while, in the unweighted case, it is 23 times larger. Clearly, the use of weights had a moderating effect on the variability of the model predictions.

Care must be taken when interpreting weighted results. The weighted residuals have been divided (normalized) by their respective weights. For this reason, the prediction error at 3.1 AU appearing in Figure 25 is much less than it appears to be. Similarly, the R-squared value that is computed for a weighted regression uses the weighted total-sum-of-squares.

User Model

Even though *Regress+* offers a large variety of models in its Equations menu, there may come a time when the model you desire cannot be fabricated from any of the built-in set. If this happens, then you have the option of typing in the equation yourself. To illustrate, we shall use the track data described earlier (pg. 25).

The model selected for these data, shown in Equation 18, is one that could be constructed from the general power model (Pow) at the cost of eliminating one parameter and redefining some others. However, just to show that it can be done, we shall define the model exactly as it appears in Equation 18.

Drop the file *Track.in* onto *Regress+*. Select the User-defined model from the Equations menu (Command-U). The Setup dialog will be replaced with a new dialog (see Figure 26).

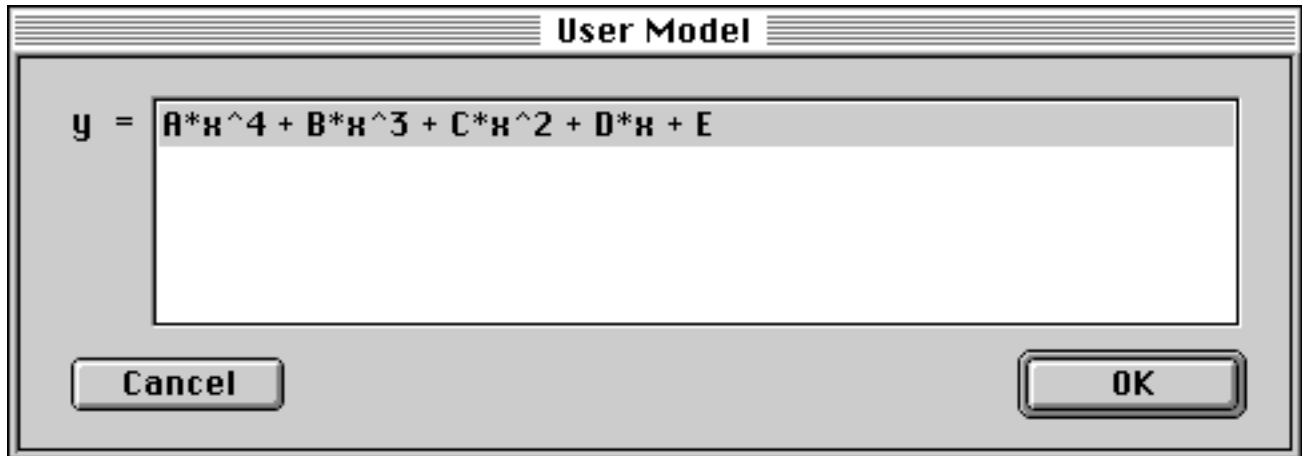


Figure 26. User-Model Dialog

The white box contains a highlighted copy of the current model which you may edit as you wish. There are only a few caveats:

- You must have at least one parameter and, at most, ten.
- The parameters must be labeled A-J and x must appear at least once.
- The dependent variable, y, may not appear on the RHS of the equation.
- The RHS may have at most 250 characters.
- You do not need to use all possible parameters, but you may not skip any.
- The operators and functions allowed are limited to those listed in Table 12.
- Everything is case-sensitive.
- Whitespace characters in the expression are ignored.
- Very little error-checking is done to validate the model. You are on your own!

These operators and functions have their usual priorities but a liberal use of parentheses is recommended, if only for legibility. Apart from the exponentiation operator, the User-model syntax is very similar to the C-language syntax. With Table 12, the following restrictions apply:

- The constant, Pi, is recognized and has approximately 18 significant figures.
- The “minus” symbol must be a hyphen, not the Option-hyphen.
- Unary minus is allowed **only** when it is the first character on the RHS or immediately following a left paren.
- Unary minus takes precedence over exponentiation, i.e., $-x^2$ is a **positive** number.
- Implied multiplication and unary plus are not supported.
- All trigonometric functions are in radian measure.
- The `log()` and `exp()` functions are the natural variants, not their base-10 counterparts.

This set of operators and functions is likely all you will need for a mathematical model of real data. When combined with the allowable length of 250 characters, even complicated expressions may be constructed.

Table 12. User Model—Valid Operators and Functions

Symbol	Interpretation
(left paren
)	right paren
+	addition
−	subtraction
−	unary minus
*	multiplication
/	division
^	exponentiation
abs	absolute value
acos	arc cosine
acosh	hyperbolic arc cosine
asin	arc sine
asinh	hyperbolic arc sine
atan	arc tangent
atanh	hyperbolic arc tangent
ceil	ceiling
cos	cosine
cosh	hyperbolic cosine
exp	exponential
floor	floor (or trunc)
log	logarithm
sin	sine
sinh	hyperbolic sine
sqrt	square root
tan	tangent
tanh	hyperbolic tangent

Overwrite the polynomial model in the dialog with the model in Equation 18. It should appear as shown in Equation 36.

$$y = A*x^B + C \quad 36.$$

with only the RHS written in the edit box.

Hit Return. This will restore the Setup dialog just as though you had picked this model from the Equations menu.

Proceed as usual from this point. The optimum values are those given on page 26. These values can be found even using the default initial parameter values but convergence will be more rapid if you make better initial guesses.

If you try the Minimum Deviation criterion as well, then you will probably observe that convergence is slower with this metric. This is usually the case. In addition, the confidence intervals for the parameters are usually wider than with least-squares.

Stochastic Modeling

Using *Regress+*, the procedure for finding the optimum model for a set of random variates is almost exactly the same as with deterministic input. Most of the dialogs and menus behave just as you would expect. The only changes are as follows:

- The input file has only a single column.
- The Setup dialog has different options enabled.
- The Distributions menu is enabled instead of the Equations menu.
- There is no User Model.
- With discrete variates, a Chi-square criterion appears in place of the K-S criterion.
- The Accuracy indicator is always enabled.

As we go through a few examples, the reasons for these changes will become apparent.

Let us begin by considering an example in which we know all of the answers beforehand. The file *Normal_2_3.in* is a sample of 30 variates that are genuinely $\sim \text{Normal}(2, 3)$. How such a known sample may be obtained, using *Regress+*, will be discussed later. Drop this file onto *Regress+*, check Confidence Intervals, set Accuracy = 10, and accept all other defaults in both the Setup and Parameter dialogs.

Unlike a regression, the goodness-of-fit metrics for stochastic modeling are not obviously good or bad. Usually, the only way to tell is to do a Monte Carlo simulation, conditioned on the model and sample size. For this reason, Accuracy is never disabled with stochastic data.

After the optimum, ML (or alternate) parameters have been obtained and the Display dialog appears, select the Continue button (or hit Return). This will initiate the goodness-of-fit test. The amount of time that it will take is roughly proportional to the number of data points, the number of parameters, and the Accuracy. Once the progress bar indicates that the test is over, the fit will be evaluated on the basis of the **worse** of the two criteria listed in the Setup dialog. This evaluation will be reported in the Display dialog and in the title of the Graph window.

The evaluation is estimated by a parametric bootstrap, as described in the Tutorial, and is based on a **one-sided** test of the percentile of the criterion value, computed using its empirical distribution, as shown in Table 13.

Table 13. Stochastic Model—Goodness-of-fit Assessments

Assessment	Percentile (P)
ACCEPTABLE	$P < 90$
MARGINALLY UNACCEPTABLE	$90 \leq P < 95$
UNACCEPTABLE	$95 \leq P < 99$
VERY UNACCEPTABLE	$P \geq 99$

Thus, a criterion value is VERY UNACCEPTABLE if it is so large that it falls in the upper one percent of the metric's CDF. Most often, if a model is truly appropriate, the percentiles for both metrics will fall in the range [40, 60].

In this example, the ML estimate for the mean (A) is 1.49844 and the bias-corrected estimate for the population standard deviation (B) is 2.48715. The real answers are, of course, 2 and 3, respectively. However, *Regress+* has no way of knowing this, and neither would you, had the source of the data not been disclosed above.

In the *.out* file, the observed ML and K-S metrics are reported to be in the 52nd and 2nd percentile, respectively. These percentiles are not perfectly reproducible since, as noted earlier, a bootstrap is, itself, a stochastic procedure. With the worse metric in the 52nd percentile, this fit is clearly ACCEPTABLE, even though the optimum parameters are not exactly the same as those of the parent population.

One reason for picking this example is that, given the estimated model, it is easy to find confidence intervals for its parameters from theory alone. Let's do so.

Recalling the discussion on page 32, the central, 95-percent confidence interval of the mean should extend 1.96 standard errors on either side of the mean, assuming that a sample of size $N = 30$ is big enough to use Normal statistics. Since the estimated standard error is 0.454, this 95-percent confidence interval should therefore be [0.608, 2.39]. The **parametric** bootstrap procedure gives [0.635, 2.41].³⁹

The latter interval assumes that the true parent population is the one estimated from the data. Also reported in the *.out* file is the corresponding confidence interval from a **nonparametric** bootstrap, computed because you checked Confidence Intervals. This estimate is based on a resampling of the data, not on sampling the model. It gives an interval of [0.789, 2.24]. The fact that these two intervals are not very different is another indication that the model is a good one. Which of these two confidence intervals is to be preferred depends entirely on the goals of your analysis.

Moving on to the standard deviation, we note the following theoretical result:

$$\frac{(N-1)s^2}{\sigma^2} \sim \chi^2(N-1) = \text{Gamma}\left(2, \frac{N-1}{2}\right) \quad 37.$$

where s^2 is the variance estimated from the sample and σ^2 is the true population variance.

With $N = 30$, the appropriate percentage points from the Chi-square distribution, using any table, are 16.0 and 45.7. Making the indicated substitutions, we find that the theoretical 95-percent confidence interval for the standard deviation of the estimated model is [1.85, 3.12]. The parametric and nonparametric bootstraps estimates were [1.87, 3.14] and [2.15, 2.94], respectively. This suggests that our particular sample was not as variable as one might have expected.

³⁹ Once again, your results may differ slightly.

Looking at the parametric bootstrap estimates for the confidence intervals of the parameters of our estimated model, we observe that neither the true mean nor the true standard deviation of the parent population may be rejected with 95-percent confidence, based on this one sample. Conversely, had we begun with the true parameters, we would have found that the estimated parameters cannot be rejected either, at the same confidence level. Finally, we note what we wanted to demonstrate once again, namely, that these bootstrap techniques are good estimators. This is especially important since, in the vast majority of analyses, theoretical computations are rarely feasible.

Discrete variates may also be considered. The Discrete submenu was disabled in the previous example because *Regress+* knew that it was inappropriate. Discrete data are signaled in one of two ways. The first is to write the input vector as a set of (positive) integers.

Unless at least one variate contains one or more decimal places, *Regress+* will consider the sample discrete.

The second method is to input grouped data, as follows:

```
@value1 frequency1
@value2 frequency2
@value3 frequency3
etc.
```

In this format, the @ symbol **must** be the first character in each record, else the record will be treated as a blank line. Values with a frequency of zero need not be entered.

The files *Rolls3.in* and *Rolls3G.in* illustrate these respective formats. This example was discussed on page 14. To reproduce the results shown in the Tutorial, use a NegativeBinomial model and a Constant value of 3 for parameter B. Actually, in part because this is such a large sample, *Regress+* makes good initial guesses all by itself.⁴⁰

With one parameter left, the ML estimate is 0.169090. The theoretical value (see Tutorial) is 1/6. As you can see by experimenting, it makes no difference whether the input is grouped or not, although the grouped file is a lot smaller.

Since this is a discrete model, the alternate criterion is Chi-square. This is, in fact, the usual goodness-of-fit metric for discrete data and tests the correspondence between the theoretical and empirical histograms, which can be observed in the graph of the result. Clearly, it is very good. How good can be determined from the Chi-square value.

The expected value of a Chi-square statistic is equal to the number of degrees of freedom. In practice, this is equal to the number of histogram bins examined during the test which, in this case, is approximately equal to the observed range, 54. Actually, *Regress+* has to extend this range so as not to truncate the tails significantly and, thereby, short-change the Chi-square statistic.

⁴⁰ Parameter B must be a Constant integer. Moreover, $B > 3$ is impossible (see pg. A-81).

The observed result is Chi-square = 65. How this relates to the expectation is not easy to predict. In fact, as the output file reports, the value 65 is in the 41st percentile. Moreover, the file discloses that the 90-percent confidence interval for Chi-square is [45.2, 176], which contains the value 54 but which is extremely skewed to the right. This is just the sort of thing that the bootstrap was invented to determine.

If you desire confidence intervals, they can be determined in the manner already described. However, with such a large file, it may take a while, depending on the speed of your computer. In the Displays menu, there is a Timeout item. This item is enabled only during a goodness-of-fit test, and only if Confidence Intervals have been checked. When Timeout is selected, *Regress+* will pause for one second after the goodness-of-fit test has finished, then continue with the remainder of the analysis, **without your having to click Continue**. This is done so that you won't have to wake up in the middle of the night to hit Return!

Generating Random Samples

Although it is not the primary purpose of this software, *Regress+* includes the capability to generate a (tab-delimited) textfile of random variates, using any allowable stochastic model. This feature exists because *Regress+* must create such samples in order to perform a goodness-of-fit test. The sample file may contain from 1 to 9,999 rows, having from 1 to 999 columns. By default, this file will have a *.samp* extension. Since this facility allows you to create a file of as many as 9,989,001 doubles, your destination volume will first be checked to ascertain that it has room for such a file.

To create the file, proceed as follows:

1. Start with an input file which could, itself, be a sample from your desired parent population.
2. Select the model as usual.
3. In the Setup dialog, check New Sample(s).
4. In the Parameter dialog, enter your desired parameter values, make them all Constant (so that *Regress+* will converge immediately), then click OK.
5. You will now see a Sample dialog similar to the one shown in Figure 27 but initialized to your current sample size.
6. Fill in the left box with the number of desired **columns** and the right box with the number of **rows**. Click OK.
7. Fill in the SaveAs... dialog as usual.⁴¹
8. The file is finished as soon as the watch cursor goes away.⁴²

⁴¹ As noted earlier, this may change your default volume.

⁴² If the variates are discrete, they will not be grouped.

Creating random variables from a known distribution is not an easy task. Often, it takes longer than you might expect. One of the reasons that some of the model parameters described in Appendix A are constrained to a range smaller than that described in standard references is due to the fact that permitting them to get much larger would mean that random samples would take far too long to generate. The Beta distribution is an obvious example as you will discover if you try to make some. These synthetic samples are of unusually high quality because the underlying pseudo-random number generator is state-of-the-art (see *Technical Details*).

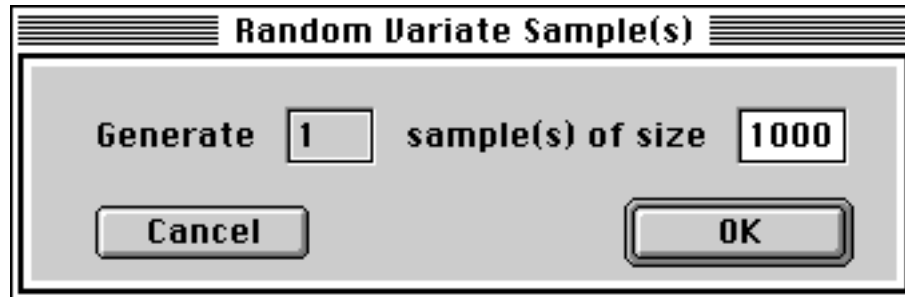


Figure 27. Sample Dialog

Summary

This completes our excursion into the world of *Regress+*. The examples discussed above illustrate all of its principal features. There remain some details found in the output files, some warning messages, for instance. However, most of these are self-explanatory. Just in case they are not, the section *Regress+ Reference* describes all of the menu and dialog items. Error and warning messages are listed in Appendix B.

By this time, you should be feeling pretty confident about your chances of finding an optimum model for your data. Before you get too confident, however, you should first read the next section. Life isn't always that easy!

WHAT CAN POSSIBLY GO WRONG?

Regress+ is about as robust as possible, considering what it does. However, nonlinear optimization in a multidimensional parameter space is a very hard task and, sometimes, things can go wrong. This section describes six major faults that might affect *Regress+* performance. Minor faults, explicable in just a few sentences, are described either in the *Reference* section, which follows, or in Appendix B.

Failure to Converge

Regress+ reports all optimum parameter values to six significant figures. Therefore, it considers an optimization to have converged when all of the parameters achieve this precision. It gives itself a lot of chances but, occasionally, this allowance is not sufficient. In that case, you will get an alert to the effect that convergence was not achieved.

At this point, you have the option of Restarting (File menu). There is no limit on the number of Restarts.

Regress+ will Restart from where it left off. When you see the Parameter dialog, you will probably note that the parameter values are now different from their previous values. It is possible that simply giving *Regress+* more time will result in eventual success.

If you try several times, convergence might still prove elusive. This **usually** means that the sample size is too small and/or the model form is not very good. As a rule-of-thumb, you should try to have at least 15 points per (non-Constant) parameter. In the case of regression, these should be unique points (unique values of x).

There is one, final trick you can try if you lose patience with Restarting. You can always make some of the parameters temporarily Constant and allow the others to converge. *Regress+* will nearly always converge with just two variable parameters. It is possible, in this fashion, to “creep up” on the desired optimum.

Once in a while, you may find that convergence is impossible when all of the parameters are allowed to “float,” i.e., to vary. You may be forced to make one of them Constant. This condition is rare and is due to the shape of the parameter space, given the model and data. Should this happen, there is really nothing that can be done to alleviate the problem completely except to use a different, preferably larger, sample.

Convergence to an Incorrect Solution

It was noted earlier that a nonlinear optimization may have multiple solutions. It may even have multiple global optima (see pg. 37). Usually, however, the spurious solution is only a local optimum, meaning that there is a better solution that lies elsewhere in the parameter space.

For this reason, it is crucial that i) you understand your data and desired model and ii) you examine the “optimum” you get with *Regress+* to ascertain that it makes sense. If the parameters are not what you expected, this is a strong hint that something may be wrong, either with your understanding or with the result.

One way to test the result for optimality is to Restart and use the alternate optimization criterion. If the model is good, both criteria should yield similar results. This test is not perfect, however, and is certainly not guaranteed to work every time. In a mathematically nonlinear world, there is very little that works every time.

On rare occasions, it is possible that there may be a spurious solution that is not very much different from the global optimum. If this happens, you are unlikely to notice the fault. Once again, the only real remedy is to repeat the optimization with a much larger sample.

Flat Response

Given some data, a model form, and an optimization criterion, the shape of the response surface, for the criterion as a function of the parameters, is fixed. Occasionally, this surface will be very flat in the neighborhood of the optimum. As a result, a change in a parameter, by an amount that *Regress+* considers significant, might not yield a significant change in the value of the criterion. Consequently, the value output for that parameter might not be as precise as it appears. That is, it might have fewer than six significant figures.

There is little that can be done to fix this fault except, perhaps, to add more data to the input file.

Systematic Error

When doing a regression, certain assumptions are made concerning the distribution of the residuals of the ML fit (see pg. 30). *Regress+* does not assess the acceptability of the empirical distribution. To do so would require an entire bootstrap analysis. However, a simple test for an undue amount of systematic error is carried out.

The way this test works can be seen by looking at the residual column from a *.list* file (e.g., Table 8, pg. 43). If you scan down this column, you will see a sequence of positive and negative values, indicating that the prediction is greater than or less than the observation, respectively. If the residuals are random, as assumed, then the **sign** of the residual must be random as well. In fact, there should be a probability of 0.5 that any residual is negative. Furthermore, given the total number of plus and minus signs, the distribution of the **number of runs** is predictable.

Regress+ assesses systematic error by comparing the number of observed runs to the theoretical prediction. For instance, in Table 8, there are 19 runs. Theory predicts 22.5 runs, but it also predicts that the probability of observing at most 19 is 14 percent. *Regress+* ignores systematic error, as defined here, unless the number of runs is so small that its probability is **less than one percent**.

Consequently, if you see an alert saying that there is an “unacceptable amount of systematic error” then you can be sure that *Regress+* is at least 99-percent certain of this assessment. In fact, it is a conservative estimate. Passing a *runs test* on the sign of the residual is a necessary, but not sufficient, criterion for the absence of systematic error. Also, if the sample size is very small, it is possible that *Regress+* might not be able to attain 99-percent certainty even when there is only one run. Should this happen, no alert will be displayed.

Very Slow Progress

Finding the optimum parameters for a chosen model form is the easiest thing that *Regress+* does. It is also the quickest. Most of the time required by this software is devoted to the bootstrap phase of the analysis, or the preparation for it. Without indulging in approximations, which *Regress+* refuses to do, the time spent is unavoidable even though it may try your patience.⁴³

As a little experimentation will disclose, the two bootstrap phases have a computation time that is roughly proportional to the sample size, the number of (non-Constant) parameters, and the Accuracy. The phase labeled “Initializing confidence intervals” is a *jackknife* computation requiring a time roughly proportional to the sample size and the number of (non-Constant) parameters.

In all cases, the form of the model and the choice of optimization criterion are important as well. Some models are simply harder to optimize than others and the default criterion usually converges faster than the alternate.

When creating a sample of random variates, the computation time is obviously proportional to the number of variates required. However, it is strongly dependent on the nature of the distribution. For instance, it is far easier to generate Uniform variates than it is to generate Beta variates.

Wishful Thinking

Finally, there is you. It is always possible that, good intentions notwithstanding, when it comes to modeling, your level of expertise might be insufficient to assure success.

Regress+ makes a lot of difficult things easy. For example, entire books have been written on methods for finding optimum parameters for a Weibull distribution yet, with a few mouse clicks, you can not only find these parameters but assess their variability and the goodness-of-fit of the model as well. This is, of course, the purpose of *Regress+*. However, there are hidden dangers in so facile a facility.

The first is that one can use this capability unthinkingly. In previous sections, it has been emphasized more than once that an analyst must be intimately familiar with his/her data and the likely form that a valid model should take. However, when sample sizes are small, the statistical power of all tests decreases and, in a given instance, you may find any number of acceptable models solely because there is so little information available in your dataset that even coarse distinctions are not feasible. Alternatively, it may be that you do not know what model form to choose because none suggests itself *a priori*.

A second, related danger arises from the plethora of models available in *Regress+*. It is very tempting to keep trying one model after another in an attempt to obtain a slightly better goodness-of-fit value, or to flip from one criterion to another for the same reason, or to make the even more elementary error of disregarding the number of parameters. *Regress+* does not

⁴³ Whenever you think that *Regress+* may take a long time, be sure to disable any background tasks, e.g., a screensaver.

perform comparative modeling explicitly. That is, it does not enable you to determine the goodness-of-fit of different models with, possibly, different numbers of parameters and return a metric telling you whether one model is **significantly** better than another. This you must do for yourself.

Lastly, there is the all-too-common error of ignoring the context of the task and confusing what is significant with what is meaningful. If, for example, you have a dataset and histogram of 1,000 variates which, given their origin, should be Gaussian and look like they are, and yet *Regress+* reports that a Gaussian model is “very unacceptable,” then this result must not be overinterpreted. In such a case, the software is saying only that there are **real** factors contributing to the observations that have not been modeled; it is **not** saying that these contributions are large or important. Indeed, it is more likely that they are of no consequence whatever. Otherwise, you would already have been aware of them. A sample of 1,000 independent points contains enough information to compass very fine distinctions, often distinctions that you may discount with impunity.

Whatever model you choose, you must be prepared to defend it. More often than not, there will be those who have conflicting ideas. If you declare that some errors are Laplacian, not Gaussian, then, eventually, you may have to provide an argument describing why this must be the case. Merely to reply that *Regress+*, or some other software package, says so will not prove a sufficient rebuttal for an expert audience.

Beware of wishful thinking. Points that appear more or less linear are not necessarily so. Likewise, a histogram that is vaguely symmetrical, with a hump in the middle, is not necessarily Gaussian, even if your textbook talks about root-sum-squares and nothing else. There is a real Universe out there, with real answers. A good analyst will try to find them.

REGRESS+ REFERENCE

This Reference lists all of the elements and options of *Regress+* and briefly describes their purposes and limitations. This section will not repeat all of the description presented earlier in this Guide. Its primary purpose is to enumerate the options and fill in any remaining gaps.

The subsections below include Input/Output, Menus, Dialogs, and Miscellaneous. Error messages are listed in Appendix B.

Input/Output

There is one kind of input file and four different kinds of output files that you may use. They are distinguished by their default extensions. All of the output files may be given any filename, including extension, as described below (see the **SaveAs...** menu item). All of the various file types, including options and comments, are illustrated in the *Examples* folder.

Note that *Regress+* does nothing with its own files. If the program is ready for input and you drag-and-drop any of its default output files onto the application, this file will not be opened. Instead, *Regress+* will display the standard input file dialog.

Regress+ always checks to make sure that there is sufficient space available on the target volume for any output files that might be large.

.in file

This must be a textfile, in Macintosh® format. It must have the (case-sensitive) extension shown and must also have TEXT as its filetype. It may be created by any kind of application that is capable of generating such a file, such as a text editor, a spreadsheet program, a database program, etc.

The format for the different kinds of input is discussed starting on page 53. Any whitespace delimiters are acceptable. The input records need not be sorted. There must be at least 7 data points. In a regression, there must be at least 7 **unique** values of x. With random variates, the standard deviation of the input must be greater than zero. The maximum number of points, given sufficient RAM, is 2,147,483,647.

One caution: A vector of random variates entered as integers will be considered **discrete**. To force *Regress+* to consider them real, use a decimal point with, at least, one decimal place in at least one entry, e.g., 1000.0 or 1.0e3.

This Guide and the sample datafiles use a semicolon to introduce comments. However, this is just a convention and a freer format is permissible. The input file will be scanned a line at a time, looking for from one to three numbers. If this fails, the second choice is an '@' character in column one, indicating grouped, discrete variates. If this fails as well, the line is ignored.

.out file

The default filename is a combination of the input filename, the model, the criterion, and, if appropriate, the Accuracy, e.g., *Radii50.Chi.ML.25.out*.

Nearly all of the contents of this file have been described earlier. The only elements not discussed are a few, self-explanatory warning messages.

.list file

This file is an option in deterministic modeling. Its filename is the same as for the .out file, but without the Accuracy, e.g., *Daytime.Sin.LS.list*.

Its contents have also been described earlier. These contents vary, depending upon whether or not the regression was weighted. This file is tab-delimited so that it may be dropped onto a spreadsheet program, giving columns immediately.

.pict file

This file is always an option. Its filename is the same as for the .list file, e.g., *BattingAvg.Gmb.ML.pict*.

A picture is saved in this file as a *bitmapped PICT*. This means that it is **not** PostScript™ and will appear, in a document, just as it did on the screen, regardless of the printer used. The size of the picture has been chosen to fit into the normal margins of an 8.5 x 11 inch page without using landscape mode. As a picture, it may be cropped and even resized but the latter operation will usually corrupt the fonts. Although some lines may appear gray, this is a black-and-white PICT.

There is no way, in *Regress+*, to change the format of the picture. It may, of course, be opened in a drawing program and modified extensively.

.samp file

This file is an option in stochastic modeling. Its filename is a combination of the input filename and the model, e.g., *BattingAvg.Gmb.samp*.

This file contains a sample of random variates, created as described on page 73. It is tab-delimited, with multiple samples, if any, in separate columns.

Menus

There are at least six menus. The first three: **Apple**, **File**, and **Edit**, are standard; the remaining three: **Equations**, **Distributions**, and **Displays**, are specific to *Regress+*. In general, menu items are disabled when they are inappropriate, either because they are untimely or because the input data are incompatible. Standard menu items have their standard Command-key equivalents, if any.

If your configuration includes System 8 or higher, there will be a seventh **Help** menu. The current version of *Regress+* does not use this menu.

Apple

Contains only the **About Regress+...** item, which presents a standard About box, including the copyright notice. If you choose this item, you **must** click inside the box to dismiss it, else the program will appear frozen.

File

The **Restart** item is specific to *Regress+*; all others are standard.

Open

Brings up the standard input file dialog. This dialog has a filter that permits it to show only folders and textfiles with a *.in* extension. Note that it is simpler to drag-and-drop an input file onto the application (or its alias).

The input file dialog may come up unexpectedly when a chosen input file is invalid for some reason.

Restart

Returns you to the Setup dialog (see below).

Close

Tells *Regress+* that you are finished with this input file but that you do not want to **Quit**.

Save

Saves the *.out* file and, if requested, the *.list* file, using their default filenames. Note that any file in the default folder with the same name will be overwritten.⁴⁴

SaveAs...

Use this command when you want to change the name of the *.out* and *.list* files. If you change the destination folder for these files, your selection becomes the default folder.

Quit

Terminates the program immediately. However, if the output would have changed, and you have not yet saved it, you will be prompted to that effect.

Edit

Contains the standard **Cut**, **Copy**, and **Paste** actions. You may use these actions for entering text whenever appropriate.

⁴⁴ The default folder is determined, in part, by your system preferences. It is the folder that you see first when choosing **SaveAs...**

Equations

Presents you with a list of the 23 built-in functions for deterministic (regression) modeling plus the User Model, in case you want a function not listed.

Distributions

Presents you with one of five lists, depending upon whether the input is continuous or discrete.

Continuous A-G

Continuous H-Z

List 30 continuous distributions.

Continuous Mixtures

Lists 17 binary mixtures of selected continuous distributions.

Discrete

Lists 5 distributions. If you see this menu when you do not expect it, then you have provided continuous input expressed solely as integers (see pg. 79).

Discrete Mixtures

Lists 4 binary mixtures of selected discrete distributions.

Displays

Allows you to see a graph of the results and save it. This menu also contains the Display timeout item.

Graph Show/Hide

Toggle for the selected graph.

Weights Show/Hide

Toggle for including the error bars in the graph. By default, these are not shown although the Y-axis has been scaled to include them.

Save As PICT...

Presents a standard file dialog with a default filename.

CDF/PDF

Toggle to alter a graph of random variates from the default PDF to the CDF, and *vice versa*.

Flip Display

Toggle between the display for A-E and that for F-J, if the latter exists.

Timeout

This toggle is enabled only if a goodness-of-fit analysis is currently being carried out and you have also opted to estimate confidence intervals. It is OFF by default. When turned ON, *Regress+* will pause for one second when the goodness-of-fit test has finished, then continue on with the remainder of the analysis without your having to hit Continue in the Display dialog. When the Confidence Intervals analysis is complete, the program halts as usual. It does not save any files automatically.

This Timeout is very useful when, for example, you want to run a long analysis overnight.

Dialogs

In addition to the standard file dialogs, which we shall not describe, there are five dialogs and one window, as follows:

Setup Dialog

Used to select the model, the criterion, the options, and the Accuracy (see pg. 55). All of these choices have been fully described previously. Canceling from the Setup means that you are finished with this input file.

Note that the New Sample(s) option is not persistent and will be unchecked after being used once.

Parameter Dialog

Used to input values for all parameters and to flag those intended to be Constant, if any (see pg. 56). Hitting OK launches the optimization. Canceling returns you to the Setup dialog. When Restarting, some parameter values of bounded distributions may be shown as slightly different from their optimum values. This is done to offset rounding error in the subsequent validity check.

Upon hitting OK, *Regress+* does a quick check of the validity of these values. If any of these is found to be invalid for any reason, the dialog will be shown again with the (first) offending value selected.

Display Dialog

Results are first presented in this dialog (see pg. 57). It also includes a status line, a progress bar (when appropriate), and a Timeout indicator while the Timeout is ON.

You must Cancel from this dialog in order to **Restart** or **Close**. However, you may **Quit** anytime.

User-Model Dialog

This is where you enter a deterministic model of your own (see pg. 68), using the operators and functions listed in Table 12, pg. 69.

Upon hitting OK, a validity check is done. If the expression is found to be invalid for any reason, the dialog will be shown again along with an error message. As noted earlier, a liberal use of parentheses is recommended.

Sample Dialog

This dialog appears after the Parameter dialog when you have opted to create a sample of random variates (see pg. 74). The model and parameters used will be those specified in the Setup and Parameter dialogs. Each sample created will appear in a separate column in the *.samp* file. All of the variates are completely independent from each other.

Graph Window

This window shows a graph of the raw data together with the model found. Stochastic data are displayed as a histogram (see pp. 11 and 13); deterministic data are displayed as X, Y points, with or without error bars (see pp. 66 and 67). The format of this graph is fixed, with the only alternatives being those listed in the **Displays** menu.

The title of this window doubles as a status message.

Miscellaneous

If you have not already done so, be sure to read, *What Can Possibly Go Wrong?* (pg. 75). There are hidden pitfalls in all mathematical modeling, especially with nonlinear models. Some of these pitfalls are described there.

Still, experience is always the best teacher. There is a brief learning curve even with a user-friendly application like *Regress+*. If you utilize the *Examples* provided in the manner intended, then you should surmount this obstacle with little effort. The figure below outlines the main features of *Regress+* functionality.

Good Luck!

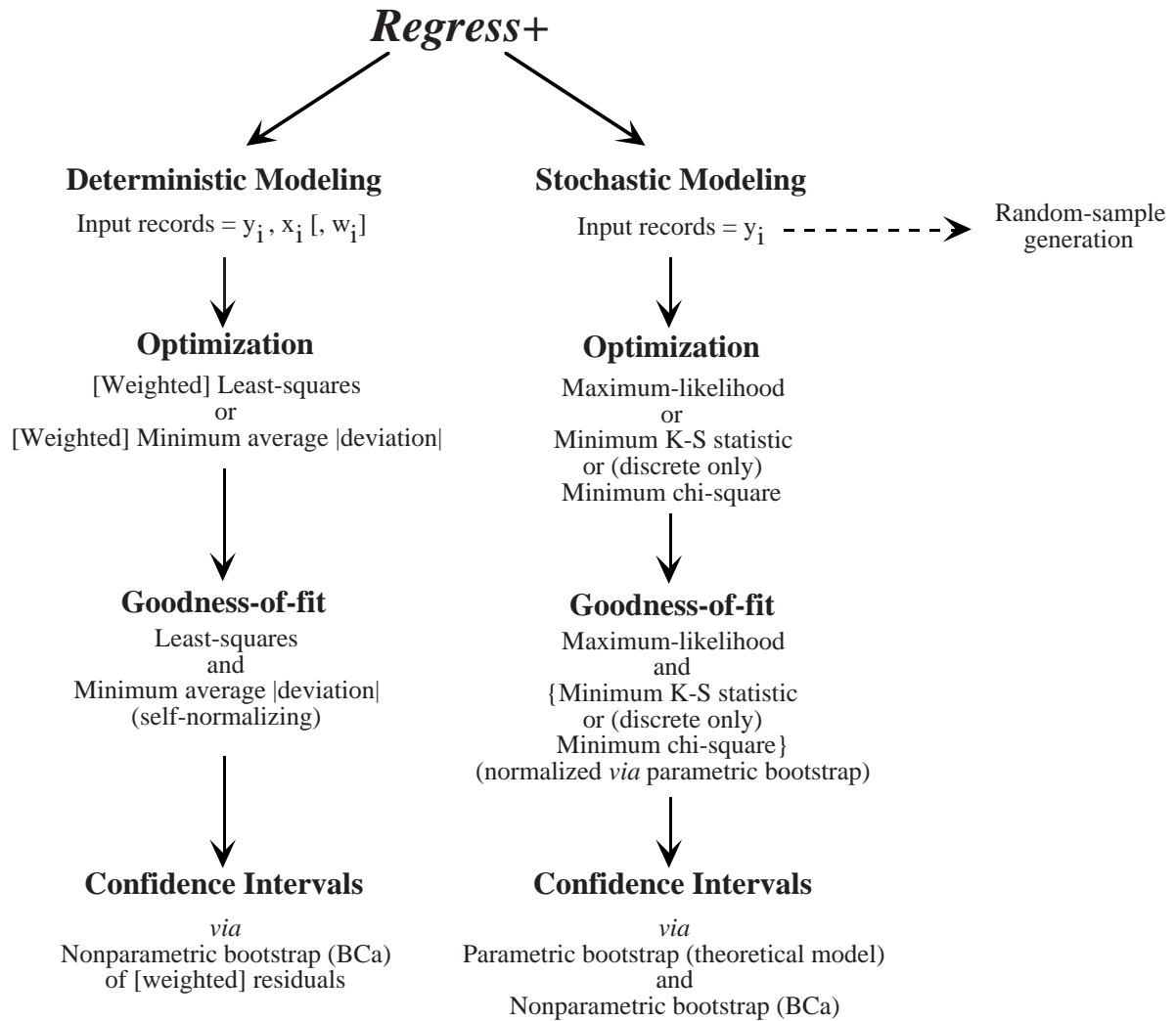


Figure 28. An Outline of *Regress+* Functionality

Technical Details

THIS section presents some details regarding the techniques and algorithms used in *Regress+*. The purpose is to describe enough of the implementation to afford professionals in this area the opportunity to determine whether or not these are algorithms in which they may have some confidence. Descriptions and references are presented for the optimization technique, the pseudo-random number generator, the generation of random variates, and the bootstrap methodology, both parametric (percentile) and nonparametric (BCa).

The descriptions in this section will not be sufficient to reproduce all of the code contained in the mathematical portions of *Regress+*. As always, there remain the numerous low-level details required to convert a general approach into a workable application. In this case, most of these are directed towards assuring that the software is robust. With nonlinear problems, convergence and singularities are persistent concerns. Speed of execution is also important. None of these issues will be discussed here.

OPTIMIZATION

The basic computation in *Regress+* is the minimization of a (usually) nonlinear objective function, given a starting vector of N parameters. This is done in a very straightforward fashion *via* the Simplex technique [CAC84, PRE92].

This well-known technique finds a **local** minimum, without the use of derivatives, by repeatedly deforming a *simplex* of $N+1$ dimensions using various geometrical transformations. This algorithm is a “greedy” algorithm and finds the nearest local minimum, “nearest” being defined, in part, by the shape of the local basin of attraction. One strong feature of this method is that it cannot diverge unless the asymptote is itself a minimum. Since the latter occurs frequently with many models, special consideration is given to this possibility. This explains, in part, why ranges for the parameters of the models described in Appendix A often have less than their theoretical span.

In *Regress+*, the simplex is initialized partly at random, by perturbing the values read from the Parameter Dialog, and the path to a minimum is almost never exactly the same in repeated trials. During the initial optimization and jackknife procedures, convergence is considered valid if and only if *Regress+* finds the same optimum vector twice consecutively.

With stochastic models, the optimization is constrained by continual validity checks on the parameters. The fact that this can be done with impunity was one of the factors considered in adopting the simplex technique.

RANDOM NUMBER GENERATOR

Nearly every analysis performed by *Regress+* requires random numbers of some sort. Hence, the pseudo-random number generator used must be highly random, very fast, and with an extremely long period.

The generator chosen for *Regress+* is an implementation of the *Ultra* generator published by Marsaglia and co-workers [MAR91]. This generator produces uniform random variates by repeated operations on a pair of 37-digit numbers in base 32. This result is combined with another, independent, multiplicative congruential pseudo-random number generator to yield an extremely random sequence with a period exceeding 10^{366} . In contrast to the usual algorithms, these uniform variates retain full precision regardless of the position of the decimal point.

The implementation, *URandomLib* [MPM98], is very fast because the basic generator is conditionally coded, depending on the CPU, in assembly. The source code for this library has been placed in the public domain. It may be accessed, on the Internet, at its web page:

<http://www.geocities.com/~mikemclaughlin/software/URandomLib.html>

RANDOM VARIATES

In order to assess the goodness-of-fit of a stochastic model, a parametric bootstrap (see below) is carried out. This procedure requires N random variates, y_i , selected from a specified parametric model. The basic generator, described above, produces a sequence of random variates that are $\sim \text{Uniform}(0,1)$. From the latter, all other variates may be generated but the method is often laborious and not at all obvious.

This section lists the kinds of techniques that are employed, in *Regress+*, for each of the 56 stochastic models available. There are five general approaches, which may be used in combination. The list below gives the method used for each distribution.

Inverse CDF

Often, a cumulative distribution function has an analytical form, $\text{CDF} = F(y)$, such that it is invertible, yielding the desired variate, $y = g(\text{CDF})$, in closed form. Starting with a CDF value, v , that is $\sim \text{Uniform}(0, 1)$, $g(v)$ then produces a variate that is described by the desired distribution.

The models in this category are the following:

Bradford	Gumbel
Burr	Hyperbolic Secant
Cauchy	Laplace
DoubleWeibull	Logistic
Exponential	Pareto
ExtremeLB	Reciprocal
Fisk	Triangular
Generalized Logistic	Uniform
Geometric	Weibull

Numerical Inverse

Even when a CDF is not invertible, as described above, a root can always be found numerically. The basic technique involves the Newton-Raphson algorithm which, in this case, is just a combination of the CDF and PDF. With appropriate checks for the validity of the result, this “method of last resort” is surprisingly competitive, especially when compared to rejection algorithms.

The models in this category are the following:

Beta	Nakagami
Cosine	Semicircular

Exact Transformation

Often, a density function may be expressed exactly as a function of some other density function(s). If generating variates from the latter can be easily done, this transformation becomes an appropriate approach for the desired distribution.

The models in this category are the following [PRE92, DEV86]:

Chi	Logarithmic
FoldedNormal	LogNormal
HalfNormal	Normal
InverseNormal	Student’s t

Rejection

This method employs one or more distributions that, together, comprise an envelope to the desired distribution. The former distributions are chosen such that generating variates from them can be done easily. This, along with their relationship(s) to the desired distribution constitute the rejection approach.

The models in this category are the following [PRE92]:

DoubleGamma	Poisson
Gamma	

Simulation

Sometimes, the simplest way to produce a variate from a distribution is to find a stochastic process that yields a result distributed as desired.

All of the binary mixture distributions fall in this category because all mimic, exactly, a pooled population composed of two sub-populations. Therefore, the mixture can be simulated by making a random selection from a sub-population with a probability equal to the weight of that component. In addition, the following distributions are simulated [DEV86, RUB81]:

Binomial	NegativeBinomial
----------	------------------

BOOTSTRAP

Regress+ makes considerable use of bootstrap techniques in testing goodness-of-fit and in estimating confidence intervals for parameters.

Assessing the goodness-of-fit of a stochastic model involves comparing its empirical goodness-of-fit metrics to their distributions. The only way to do this robustly, without making asymptotic, or other, approximations, so common in available tables of critical points, is to perform a Monte Carlo simulation, a kind of parametric bootstrap.

Nonparametric bootstrap techniques are used extensively to assess both deterministic as well as stochastic models. In this case, the data themselves are used in place of a parametric model.

It must be emphasized that bootstrap techniques, especially nonparametric techniques, are an active area of research. The combination of methods described below constitutes merely one of many useful slices that may be made through the space of bootstrap methodology. It will be a long time, if ever, before the final word is said on this subject.

Parametric Bootstrap

After *Regress+* computes an optimum model for a set of random variates, the next phase of analysis is to determine whether or not the model is acceptable. This is done using one-sided tests for the two metrics listed in the Setup dialog. The null hypothesis, in each case, is that the observed metric is not too large. That is, it would not fall in a high **percentile** of its distribution, conditioned on both the model and the sample size.

The approach here is the intuitive one:

1. Select N random variates, where N is the original sample size, from the putative parent density to give one (parametric) bootstrap sample.
2. Subject this bootstrap sample to a new optimization, exactly as was done with the original sample.
3. Compute two new goodness-of-fit metrics in the usual way and save them.
4. Repeat steps 1-3 for a total of at least 1,000 times, producing two arrays of that size, one for each metric (as well as arrays for the parameters).
5. Sort the two arrays to give bootstrap estimates of the empirical distributions for these goodness-of-fit metrics.
6. Compare the original, observed values for these metrics to their respective bootstrapped distributions and find the percentiles for each.

Regress+ makes an assessment based on the **worse** of the two metrics (see pg. 70).

Nonparametric Bootstrap

This description is an extremely brief synopsis of part of the excellent treatise referenced earlier [EFR93]. No attempt will be made to present material sufficient for implementation. Instead, the reader is urged to consult the cited reference.

The basic idea behind the nonparametric bootstrap is that, if a sample is truly representative of its parent population then, given suitable precautions, **sampling from the sample is tantamount to sampling from the population**. The term “bootstrap” derives, of course, from this observation.

A nonparametric bootstrap analysis proceeds as follows:

1. From an observed sample of size N , select N elements at random, with replacement, to give a (nonparametric) bootstrap sample.
2. Treat this bootstrap sample just as though it were an observed sample.
3. Repeat steps 1-2 for a total of at least 1,000 times, producing arrays of that size for whatever statistics (or parameters) might be of interest.
4. Sort the arrays to give bootstrap estimates of the empirical distributions for these statistics/parameters.

It is from these nonparametric distributions that any conclusions may be drawn.

How to take “suitable precautions” is crucial. It is something that has only recently been discovered and, indeed, is still under investigation. It is known, for instance, that a nonparametric bootstrap distribution, created as described above, is both biased and skewed with respect to the true distributions for the same quantities. One method for correcting for these faults is the *bias-correction with acceleration* (BCa) technique. This technique is the one employed by *Regress+*.

Since the cited reference devotes a large number of pages to the BCa technique, we shall note here only the final conclusions. It is proven that the corrections made by this method are quite valid and estimates, especially confidence intervals, are significantly improved when BCa corrections are applied.

In addition, BCa confidence intervals have two very important features. First, they are transformation-respecting. BCa confidence limits transform correctly if an estimated limit, w , is transformed into some function, $f(w)$. Thus, a confidence interval for variance could be easily transformed into one for standard deviation.

The second feature concerns the coverage of a BCa confidence interval. A central confidence interval of P percent should be bounded by two tails, each exhibiting an integral of $(1 - P)/200$. BCa intervals can be shown to have this property.

Although BCa estimates have some very excellent properties, they can sometimes take a long time to compute. Approximate variants of the method have been developed. However, as

noted more than once, *Regress+* refuses to make any unnecessary approximations and, therefore, it absorbs the time penalty usually associated with this technique. There are ample alternatives available to anyone with Internet access and a search engine.

Unlike the parametric bootstrap, the BCa technique requires two parameters of its own. One of these, related to the bias, can be estimated in closed form. The other, related to skewness (acceleration), is estimated using a jackknife analysis (see pg. 77). In this analysis, one point at a time is deleted from the data and a model estimated from this truncated dataset. For each statistic/parameter of interest, an acceleration parameter is then estimated from the skewness of the corresponding, jackknifed distribution.

In *Regress+*, nonparametric bootstrap methods are used to compute confidence intervals for model parameters. With stochastic models, the original data vector is sampled. With deterministic models, what is sampled is the vector of (weighted) residuals resulting from the optimum model. A random (re-weighted) residual is added to each observed y-value in turn, producing a new dataset with the same x-values.

REFERENCES

- Abell, G., 1964, *Exploration of the Universe*, Bantam Books, Holt, Rinehart and Winston, New York, NY. Note: This is just one of several sources for the complete datafile.
- Caceci, M. S. and W. P. Cacheris, 1984, "Fitting Curves to Data", *Byte*, **9** (5), pg. 340.
- Devroye, L., 1986, *Non-Uniform Random Variate Generation*, Springer-Verlag, New York, NY.
- Efron B. and R. J. Tibshirani, 1993, *An Introduction to the Bootstrap*, Chapman & Hall, New York, NY.
- Famighetti, R. (ed.), 1995-8, *The World Almanac and Book of Facts*, World Almanac Books, Mahwah, NJ.
- Famighetti, R. (ed.), 1998, op. cit., pg. 940.
- Johnson, N. L., Kotz, S. and A. W. Kemp, 1992, *Univariate Discrete Distributions*, 2nd ed., John Wiley & Sons, New York, NY.
- Johnson, N. L., Kotz, S. and N. Balakrishnan, 1994-5, *Continuous Univariate Distributions*, vols. 1-2, John Wiley & Sons, New York, NY.
- Kepler, J., 1609, *New Astronomy*, (W. H. Donahue, 1992, trans.), Cambridge University Press, Cambridge, UK, pg. 537.
- Kotz, S. and N. L. Johnson (eds.), 1982-9, *Encyclopedia of Statistical Sciences*, John Wiley & Sons, New York, NY.
- Marsaglia, G. and A. Zaman, 1991, "A New Class of Random Number Generators", *The Annals of Applied Probability*, **1** (3), pg. 462.
- McLaughlin, M.P., 1998, "URandomLib: The Ultimate Macintosh Random-Number Generator", *MacTech*, **14** (10), pg. 38.
- Press, W. H., et al., 1992, *Numerical Recipes in C*, 2nd ed., Cambridge University Press, Cambridge, UK.
- Rauer, H., et al., 1997, "Optical Observations of Comet Hale-Bopp (C/1995 O1) at Large Heliocentric Distances Before Perihelion", *Science*, **275**, pg. 1909.
- Rubinstein, R. Y., 1981, *Simulation and the Monte Carlo Method*, John Wiley & Sons, New York, NY, pg. 82.
- Standish, E. M., et al., 1997, *JPL Planetary and Lunar Ephemerides: DE406*, The Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA.

Udem, Th., et al., 1997, "Phase-Coherent Measurement of the Hydrogen $1S$ - $2S$ Transition Frequency with an Optical Frequency Interval Divider Chain", *Physical Review Letters*, **79** (14), pg. 2646.

Weast, R. C., 1975 (ed.), *Handbook of Chemistry and Physics*, 56th ed., CRC Press, Cleveland, OH. Note: This is just one of several sources used to compile the hydrogen spectrum.

Young, M. C., 1997 (ed.), *The Guinness Book of Records*, 1997, Bantam Books, New York, NY.

APPENDIX A

A COMPENDIUM OF COMMON PROBABILITY DISTRIBUTIONS

<Due to its large size (120 pp.), Appendix A is now published separately.>

APPENDIX B

ERROR MESSAGES

There are four kinds of error messages. *Fatal* errors terminate the program. *Warnings* tell you that something is not going to work. *Notes* allow the program to continue without error, but not in the manner intended. *User-model errors* indicate that something is wrong with the model that was typed in. The various messages in each category are listed here in alphabetical order.

FATAL ERRORS

Out of Memory!

Regress+ has run out of heap space. After the program terminates, select its icon, then Get Info, from the File menu (or hit Command-I). Increase heap space (allocated memory) in the indicated place at the bottom of the resulting dialog. It is OK to use more than the 1000K suggested.

This platform is too old for Regress+.

Regress+ checks the hardware platform and the installed operating system. If the hardware is outdated, or the System less than System 7, the program terminates.

WARNINGS

Insufficient space on default volume.

The amount of space available for an output file appears to be insufficient for the estimated size of that file. You are unlikely to see this message unless you are trying to create a very large file of random variates. You will have to save such a file to a volume with more free space.

Output file error.

This is a general warning that an output file could not be created. It may be due to a locked volume, insufficient RAM left to make a picture, or to other causes.

Output not saved!

You will see this message whenever you Close or Quit without saving the standard *.out* file. If you save the output, then take any action that would cause it to change, this message may be issued again.

There is an unacceptable amount of systematic error with this model.

Your model failed the Runs test (see pg. 76). Thus, there is a real factor evident in the data that your model does not take into consideration. Note, if your sample size is large, this may be a very small factor.

There is something wrong with this dataset.

This indicates that the format of your *.in* file is not in accordance with the rules described on pg. 79, and elsewhere.

This model failed to converge.

Regress+ does not always succeed in achieving convergence in its allotted quota of attempts (see pg. 75). You may try again, any number of times, by Restarting (Command-R) and accepting all subsequent defaults.

Usually, the reason for lack of convergence is obvious (e.g., poor initial guesses) but not always. For instance, a parameter might fail to converge repeatedly because, given the precision of the input, the digits trying to converge are not significant and so the optimum is not reproducible. In this case, just make that parameter Constant.

NOTES**Since all parameters are constant, Confidence Intervals are disabled.**

Confidence intervals make no sense if the parameters are not allowed to vary. They are then exact, by definition.

Since there are duplicate points, weights are disabled.

When there are multiple values of y for a given x_k , these *duplicate points* determine an implicit value for the standard deviation of $y(x_k)$. Since there is nothing to prevent this value from contradicting the value inserted into the input file as a weight (w_k), the former takes precedence over the latter. If you wish to override this message, you must add very tiny deltas (e.g., $1e-10$) to each duplicate x -value. *Regress+* checks only for strict equality.

Whenever there are duplicate points, the *.out* file will state that R-squared is diminished. This is due to the fact that a portion of ESS is directly attributable to the scattering of multiple points at single values of x . Were there only one point at each x , this could not occur.

USER-MODEL ERRORS**Bad constant**

A constant appears to be either infinite or, more likely, *NaN* (“not a number”).

Bad expression

This means that *Regress+* could not make sense of your model. It is probably due to a typographical error (see also, Table 12, pg. 69).

Bad parameters

You must use A-E for the parameters, starting with A and not skipping any letters. Remember that everything is case-sensitive.

Mismatched parentheses

The parentheses are unbalanced.

Starting parameters invalid!

Regress+ does a minimal evaluation check on the model, using the default starting parameters (all equal to one). If the result is infinity or *NaN*, this message will be issued.

Too many parameters

When you employ the User-specified model with more than five parameters, the number of unique values of x must exceed the number of parameters by at least two.

Unknown symbol

A symbol is not in the list of recognized symbols.