Sample Sections

Collected here, in two parts, are the preface, the table of contents, and a number of chapter sections that will show you the breadth and depth of the third, 2012 edition of my (copyrighted) *Advanced Excel for scientific data analysis*. They tend to emphasize some of the newer material rather than the nitty-gritty of statistics, data analysis, and VBA coding that are also covered in this book, but it may at least give you an idea of its level and style. Occasionally the page layout had to be adjusted from that used in the book.

Preface (AE3 pp. vii-xii)

Even more than its earlier editions, the present volume will take its readers far beyond the standard spreadsheet fare provided by Microsoft. While its contents have been expanded by the inclusion of a number of new features, its general organization is unchanged from that of the second edition. After the introduction you will find three chapters on least squares analysis, followed by two on Fourier transformation and related methods, and one on digital simulation. The next chapter describes in some detail how to use VBA to write user-designed functions and macros to expand the reach of the spreadsheet and make it solve *your* problems. The three final chapters illustrate how user-designed functions and macros can improve some standard numerical methods, facilitate the uses of matrix algebra, and can greatly enhance the accuracy of the spreadsheet.

The most visible change in this third edition is its significantly enlarged page format, because the earlier edition had reached its practical thickness limit. The text has undergone a number of useful updates, with more emphasis on visual representations such as error surfaces, and the inclusion of several entirely new sections: chapter 1 sections 2.4 through 2.7 and 5.2; chapter 4 sections 21, 23, 24, and 26; chapter 5 section 11; chapter 7 sections 4 and 7; chapter 8 sections 3, 16, and 22; chapter 9 sections 2.10, 2.11, 4.4, and 6.3; chapter 10 sections 11, 12, 16, 17, 19 and 20, as well as much of chapter 11. Chapter 4 now includes a brief discussion of a Levenberg-Marquardt routine that consistently outperforms Solver on the NIST standard reference tests, and chapter 9 contains a novel algorithm for the numerical differentiation of mathematical functions that, for reasons outside of my control, did not make it into the second edition, and caused me to change publishers. Moreover, the recent, quite substantial extension of Volpi's Xnumbers.xla by John Beyers to XN.xla and XN.xlam has been incorporated throughout this book, because it can be used throughout Excel (on the spreadsheet, in functions, and in subroutines) with all recent pc versions of Excel, from 2000 through 2010. All the needed add-ins are freely downloadable from my website, http://www.bowdoin.edu/~rdelevie/excellaneous, which now doubles as a convenient transfer station for the many contributions by Volpi, Beyers, and others that are described in this book.

*****

Science and engineering progress through a mixture of three essential ingredients: ideas, tools, and experiments. Excel is a ubiquitous and rather intuitive computational program that, through the selfless efforts of people like Leonardo Volpi and John Beyers, has now become a serious numerical analysis tool. As such, it is a means to an end, a distinction that was obviously lost on a reviewer of an earlier edition of this book who advised readers not to use Excel "for anything serious like curing cancer". No software will cure cancer, but it can surely assist in that and other worthwhile scientific and engineering efforts, if only because of its low psychological entrance barrier, open architecture, wide distribution, and computational prowess.

This book is intended for those who are already familiar with the Excel spreadsheet, and who want to explore what it can offer them in the physical sciences and engineering. It is not intended as an introduction to Excel, nor is it meant for those who are already comfortable with high-performance general-purpose numerical software platforms such as Fortran or C, with programs that can do symbolic math such as Mathematica or Maple, with more specialized numerical packages such as Matlab or statistical

ones like SPSS, SAS or R, and therefore may not find it worthwhile to investigate what Excel can do for them. All such programs can perform most any task that can be done on Excel, and as a consequence of Volpi's work, Excel can now also perform many tasks performed by those high-performance numerical analysis packages, even though the extra computational overhead of its user-friendliness makes the spreadsheet unsuitable for really large problems, and an extensive library of statistical applications still remains to be written. However, for most small and medium-sized scientific and engineering computations, the choice of platform often depends primarily on its availability as well as on one's prior familiarity with that particular software, and both of these criteria often tend to favor the spreadsheet.

Long before the opening up of the internet, two innovative programs, word processing and spreadsheets, had already fueled the explosive development of desktop and laptop computers and their current, wide distribution makes them uniquely suitable for a close examination of their capabilities. At present, whether we like it or not, Excel is by far the most widely distributed numerical software platform in the world. In my own experience, as a teacher charged with introducing incoming graduate chemistry students to numerical data analysis, I found that most students preferred the low barrier to learning, the open structure, and the visual and numerical immediacy of the spreadsheet to more formal languages such as Fortran or C, even though the latter at that time were certainly much more powerful, and already came with impressive subroutine libraries. Its user-friendliness makes the spreadsheet an excellent teaching tool, even if its openness and great flexibility can also be its Achilles heel, as described in chapter 11, and an accountant's headache. Developing VBA and incorporating it into Excel was Microsoft's masterstroke that made it overtake its main competitors, Lotus 1-2-3 and QuattroPro, and this book fully exploits that aspect. Unfortunately, the unwillingness of Microsoft to promptly correct sub-optimal software has turned off many potential users, especially engineers and statisticians. Although Microsoft has not changed its habits, the work of Volpi and Beyers goes a long way towards overcoming that problem.

*****

The introductory chapter surveys some of the standard features of Excel, and therefore can serve as a brief refresher. It also contains some more advanced material that needed to be introduced there in order to be available in subsequent chapters. However, the novice user of Excel is urged first to consult a manual, or an introductory book, as can usually be found on the shelves of a local library or bookstore.

This book has two main parts. Chapters 2 through 7 deal with some of the main analysis methods used in science and engineering, specifically least squares analysis, Fourier transformation, and rudimentary digital simulation, as applied to the Excel spreadsheet. The emphasis is not so much on their mechanics, but on their proper use. The next four chapters, 8 through 11, focus more strongly on implementing these methods with custom functions and macros in those cases for which Excel does not provide the needed software.

In order to avoid possible confusion, we will adopt several definitions and distinctions that are not part of common English usage, and therefore need a brief explanation. In data analysis, one often deals with experimental uncertainty in the data. In this connection we will follow a convention adopted by most physical scientists, viz. to distinguish between their *accuracy*, i.e., how close these data are to their true values, and their *precision*, i.e., how reproducible they are under the experimental conditions used. This is a useful general distinction, even though precision and accuracy may sometimes get entangled in the data analysis.

Amongst numbers we will distinguish between measured *variables*, i.e., experimentally determined quantities such as distance, time, current, voltage, absorbance, wavelength, concentration, and pH, and computed *parameters*, the model-based constants that we extract from experimental data, such as speed, acceleration, resistance, molar absorptivity, equilibrium constant, etc. This distinction is not necessarily a physical one, e.g., acceleration can be determined directly, or derived from measuring distance as a function of time. Moreover, fitting parameters may be purely empirical, in which the "model" is some convenient mathematical function that appears to fit. But it is always useful in data analysis to separate the directly measured data from those derived from them during the analysis, and to consider how that data analysis itself can sometimes affect the results.

In working with graphs, Microsoft distinguishes between a *chart* and a *plot*. A chart contains the actual data, just as a canvas contains a painting, while the plot contains the chart plus its immediate surrounding, including axis labels, just as a frame surrounds a painting and, in a museum setting, may also include an explanatory label. Likewise, we will differentiate between an *array* (a set of data) and a *range*

(a set of spreadsheet cells) and, when discussing VBA procedures, between *functions* (self-updating code) and *subroutines* (code that only responds when called). *Macros* are a special type of subroutine, callable from the spreadsheet, just as public functions are a callable subset of all (public and private) functions.

*****

The predecessor of this book, *How to use Excel in analytical chemistry and in general scientific data analysis* (Oxford Univ. Press 2001) focused on the responsible use of statistics in analytical chemistry, and to this end contained a number of add-in macros. The present book addresses a much more general readership, but continues the emphasis of its predecessor. The second edition added some of Leonardo Volpi's routines, especially for matrix functions (Matrix.xla) and for extended-precision macros (Xnumbers.dll). The latter add-in has become obsolete with Excel version 2007, and is replaced in the present edition by the more powerful and flexible XN.xla(m), as extended by John Beyers. As a consequence, there is some overlap between the various add-ins described in this book, and the user sometimes has several more or less equivalent options, for both functions and macros, even though other areas are still underrepresented, such as the statistical approaches chemists often describe as chemometric, or more sophisticated numerical simulations. I hope that some statistician(s) thoroughly familiar with those fields will step in to fill this void.

Most of the instructions in this book apply to versions of Excel for IBM-type personal computers starting with Excel 97, although some new features (such as cell comments) have only been available in more recent versions of Excel, and may therefore not work in the earliest versions. The specific instructions in this book were mostly tested in Excel 2000 and 2003, still my favorite versions. In 2007 Excel underwent a major cosmetic facelift, which affected neither its underlying computational engine nor VBA, but changed the names of many first-line commands, a real nuisance for old-timers.

Differences in Excel on an IBM-type personal computer or on a Macintosh are relatively minor, such as using Ctrl$\cup$click instead of right-click, and the Option key instead of the Alternate key. (The linking symbol $\cup$ is used here to indicate that the Ctrl key should remain depressed while clicking. We use it instead of the more common + because you will be less inclined to type it.) Appendix A.8 lists some other differences between pc and Mac commands. Excel versions 1 through 4 used a completely different macro language, and are therefore *not* suitable for use with this book, since none of the custom functions and macros described here will work in those early versions. If you have such an early version, it is high time to upgrade.

Idiosyncratic notation has been kept to a minimum, with three exceptions. The notation 3 (2) 9 is used as convenient shorthand for the arithmetic progression 3, 5, 7, 9 (i.e., starting at 3, with increment 2, ending at 9). The linking symbol $\cup$ is used to indicate when keys should be depressed simultaneously, as in Alt$\cup$F11 or Ctrl$\cup$Alt$\cup$Del. And the symbol $\varnothing$ will identify deconvolution, complementing the more usual symbol $\otimes$ for convolution. These symbols are based on the fact that, in the Fourier transform domain, deconvolution is associated with division (/) just as convolution is with multiplication ($\times$).

Especially in the first few chapters of this book an effort has been made to show the reader how to use Excel 2007 and 2010, in which the look and feel and, more importantly, the names of many of its first-line commands were changed to sell old wine in new bottles. Despite those changes, the material in this book is applicable to these recent versions of Excel.

Excel 2007 and 2010 will accept your earlier Excel spreadsheets, but are *not* fully backward compatible, and its use may therefore require you to make some changes. For scientific data analysis, there is little to recommend upgrading from Excel 2003 to its newer versions, and much to discourage it if you were already an experienced user before 2007. One can of course run Excel 2003 on the new Vista platform to gain its supposedly improved immunity to viruses and other unwanted intrusions.

Excel 2007 has largely done away with toolbars. Instead you will find a series of ribbons that display the most popular choices but were often renamed, take up more space for fewer options, and consequently require frequent ribbon-changing. Fortunately, many (though not all) of the old hotkey commands still work in Excel 2007/10. There was also a useful change in Excel 2007/10, its greatly increased spreadsheet size, but the corresponding instruction set was not given a corresponding update or expansion. (The "64-bit" versions of Excel 2010 offer 64-bit cell addressing, but unfortunately still no 64-bit computing.) The expansion of the spreadsheet area in Excel 2007 required extending the column designations to three letters. Therefore, before importing your pre-existing functions into Excel 2007/10, check whether it uses

names that will now become cell addresses and, if so, change them. We use the marker 07,10 to indicate, especially in the first chapters, where Excel 2007/10 requires an approach different from that of its earlier versions. For the sake of uniformity we will indicate the old hotkey commands (by underlining specific letters), most of which still apply, even though the instruction itself may be called by a different name in the most recent versions.

*****

The focus of this book is on the *numerical analysis of experimental data*, such as are usually encountered in the physical sciences and in engineering. Much of such data analysis is nowadays performed with one of two approaches, least squares analysis or Fourier transformation, which therefore form the first major subject areas of this book. But the emphasis is neither on numerical analysis as an abstract mathematical subject, nor on its specific computer implementations, but on analyzing experimental data and extracting the best possible information from them, and on explaining the basic principles involved, primarily by example. We therefore relegate most of the numerical manipulations to functions and macros, and in this book focus instead on how best to *use* these tools. With such understanding, with the tools described, and with knowing how to make your own Excel tools when none are already available, this book aims to make the spreadsheet do *your* bidding, not so much by prettying up its display, as by exploiting its considerable computational capabilities to the fullest. And in case Excel's built-in facilities and its many add-ons don't provide what you need, this book describes how to make Excel routines to fill *your* specific needs, shows you many specific examples (and pilferable parts) in the associated MacroBundle, and has a number of MacroMorsels specifically illuminating aspects of VBA that might otherwise cause you some coding difficulties.

In the current edition we have incorporated many more graphical aids, especially those that can visualize central concepts such as (the logarithm of) SSR, the ***s***um of ***s***quares of the ***r***esiduals, i.e., the quantity usually minimized in least squares routines. The shapes of such plots can often alert us to potential difficulties with the associated data analysis.

If you want to use the rectangular spreadsheet array for linear algebra (they are clearly made for each other) but feel constrained by the limited set of matrix instructions provided by Excel, you will find many additional tools in chapter 10. If you require scientific, engineering, and statistical functions that Microsoft does not have, need higher accuracy than Excel can provide, or precision higher than the IEEE 754 standard "double precision", read chapter 11. In short, this book empowers you to do much more with Excel than you might otherwise have thought possible.

As much as feasible, the material is illustrated with practical examples taken from the scientific literature, and specific instructions are provided so that you can reproduce the analyses, and can then modify them to suit you own needs. Real examples have the advantage that they often highlight realistic applications, but there is a second reason to favour real measurements over made-up examples, viz. the fundamental difference between "pure" mathematics on the one hand, and numerical computation and the physical sciences on the other. In pure math, numbers are well-defined, hard objects, while in numerical computation and in the physical sciences they are usually somewhat fuzzier and softer. This is readily seen for criteria such as $x < 0$, $x = 0$, and $x > 0$, which are absolute in math, but can become vague due to numerical round-off errors in computation and/or to experimental errors in scientific data.

Since this book deals with the application of numerical software to scientific data, this softness is often emphasized, and methods to minimize its effects are stressed. Least squares methods, e.g., tend to minimize the effects of random experimental errors, deconvolution those of systematic ones, while careful choice of algorithms and/or use of extended numberlength can reduce computational errors. Fourier transformation and singular value decomposition allow us to filter out less significant signals from those of interest, and hybrids between these various methods can often combine some of their best aspects. Such tools can and should play a role in data analysis, i.e., in converting experimental numbers into useful information.

*****

Almost all of the special routines described in this book can be downloaded as open-access functions, macros and subroutines. Where necessary, short introductions are given to the topics being introduced, but you will not find any conjectures, propositions, lemmas or theorems here, but instead simple explanations of the basic principles involved, often accompanied by a few crucial equations. You will not encounter any screenshots of Excel dialog boxes either, or any templates to fill in, but plenty of illustrations

of actual spreadsheets, often accompanied by a listing of the explicit instructions used. This makes this text useful for individual study as well as for an introductory course in applied numerical analysis in the physical sciences and engineering, especially when its worked examples are combined with projects of the student's own choosing. I have had my students in courses using this book work on a wide variety of topics, from the ups and downs of the stock market and the tides (some contrast!) to an analysis of the supposedly equivalent (but evolutionarily maintained, and therefore almost certainly significant) synonyms in the four-letter amino acid code, where the Word-related VBA commands operating on letters and strings came in handy.

As always I will be grateful for reader's comments, corrections, and suggestions. Please address these to rdelevie@ bowdoin.edu.

Acknowledgements

First I would like to express my indebtedness to Leonardo Volpi and his many coworkers, who built much of the material described in this book, and to John Beyers and his brother Steve, who have continued its development and further extension. It is the marvel of the Internet and of the spirit of open access software, that Excel has now been transformed from a competent business software package to a first-rate instrument for scientific data analysis, while fully retaining the ease of use and visual rather than command-line-based orientation of the spreadsheet. It is an ironic consequence of that same Internet that I have so far met in person none of those mentioned here, and have corresponded only with a few of them: Leonardo Volpi, John Beyers, David Heiser, and Simonluca Santoro. Here, then, listed alphabetically, are the names of others whom I have only met on paper as contributing to this book: Eric Braekevelt, R. C. Brewer, Ricardo Martinez Camacho, Wu Chinswei, Lieven Dossche, Berend Engelbrecht, Rodrigo Farinha, Mariano Felici, Arnoud de Grammont, Hans Gunter, David F. Haslam, Michael Hautus, André Hendriks, Richard Huxtable, Ton Jeursen, Jianming Jin, John Jones, Michael Kozluk, Giovanni Longo, Bruno Monastero, Javie Martin Montalban, Sebastián Naccas, Takuya Ooura, Kent Osband, Robert Pigeon, Simon de Pressinger, Roger Price, Luis Isaac Ramos Garcia, James R. Ramsden, Michael Richter, Iván Vega Rivera, Michael Ruder, Mirko Sartori, Gerald Schmidt, Gabriel Simmonds, Vidas Sukackas, David Sloan, Ken Thompson, Christopher Titus, Abel Torres, Alfredo Álvarez Valdivia, Franz Josef Vögel, Shaun Walker, Gregg B. Wells, P. J. Weng, Yong Yang, Vladimir Zakharov, Jakub Zalewski, Thomas Zeutschler, Shanjie Zhang and Oldgierd Zieba. In the name of the readers and users of this book: thank you all for your generous contributions.

Numerous friends, colleagues and students have contributed to this book, corrected some of its ambiguities, and made it more intelligible. I am especially grateful to Bill Craig for invaluable help on many occasions, to Whitney King, Panos Nikitas, Carl Salter, and Brian Tissue for their many helpful comments, especially on the chapters on least squares, to Peter Griffiths and Jim de Haseth for commenting on the chapter on Fourier transformation, to Peter Jansson for valuable comments on deconvolution, to Philip Barak for letting me use his elegant equidistant least squares macro, to Simonluca Santoro for letting me incorporate his beautiful contour diagrams, and to Harry Frank, Edwin Meyer, Caryn Sanford Seney, and Carl Salter, for sending me experimental data that are so much more realistic than simulations would have been. I gladly acknowledge the various copyright holders for permission to quote from their writings or to use their published data, and I am grateful to William T. Vetterling of Numerical Recipes Software for permission to incorporate some programs from *Numerical Recipes* in the sample macros.

I am very grateful for the many helpful comments and suggestions from colleagues and friends, especially from John Beyers, Stephen Bullen, Bosco Emmanuel, Steve Feldberg, David Heiser, Nemad Jeremic, Linde Koch, Ernest Lippert, Jan Myland, Keith Oldham, Hans Pottel, Simonluca Santoro, Mohammad Tajdari, Joel Tellinghuisen, and alphabetically last but helpwise first and foremost, Leonardo Volpi. My wife Jolanda was the great facilitator whose love, support, forbearance, proofreading, and help made it all possible.

Contents (AE3 pp. xiii-xxi)

1.5.3 Band maps (AE3 pp. 26-29)

In section 1.5.1 we saw that Mapper can make graduated color (or gray-scale) maps. This same routine can also display corresponding color bands, thereby making quick-and-dirty quasi-contour diagrams, e.g., at fixed percentage intervals. This works best for large arrays, because in that case the pixellation of the band edges is least noticeable, precisely those conditions where IsoL.xls is rather slow. Fig. 1.5.4 shows such a band map of the van der Waals equation displayed already with IsoL.xls in Fig. 1.5.3, and similarly annotated. We see that Mapper can indeed be used as a quick understudy for the more refined IsoL.xls. Here we placed the number one in the otherwise unused top-left-hand corner of the highlighted data block, which forces Mapper to set lower and upper limits, which were then specified as 0 and 2.1 to match the contours of Fig. 1.5.3.

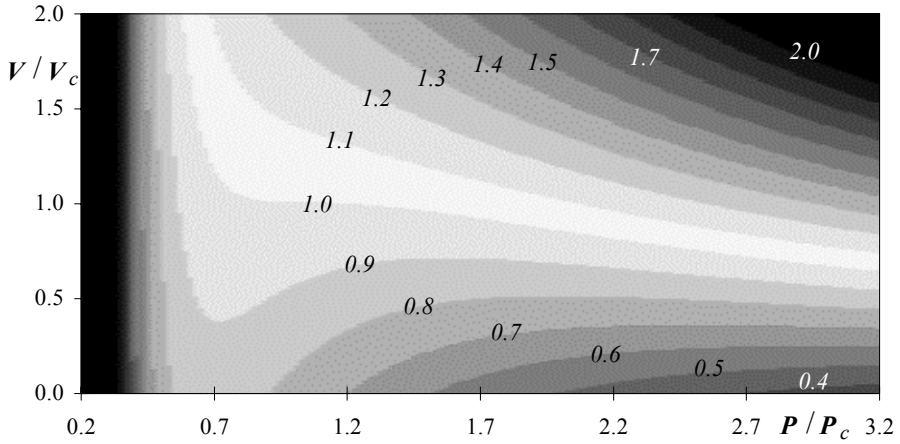

Fig. 1.5.4: A contour map drawn of the dimensionless van der Waals diagram, drawn with the 21-band gray-scale Mapper0000, with similar scale and annotation as in Fig. 1.5.3.

Figure 1.5.5 shows a map of the Rosenbrock function $z = 100\,(x^2 - y)^2 + (x - 1)^2$, of which we already encountered a 3-D plot in Fig. 1.4.4. An array of 201×201 data points representing this function is displayed here with the 10-band Mapper00. This image shows the values of z calculated in a rectangular array of evenly spaced increments in the x- and y-coordinates. The parabolic valley of Fig. 1.4.4 is here seen as a black band, and does not indicate where the lowest point in that broad valley might be located.

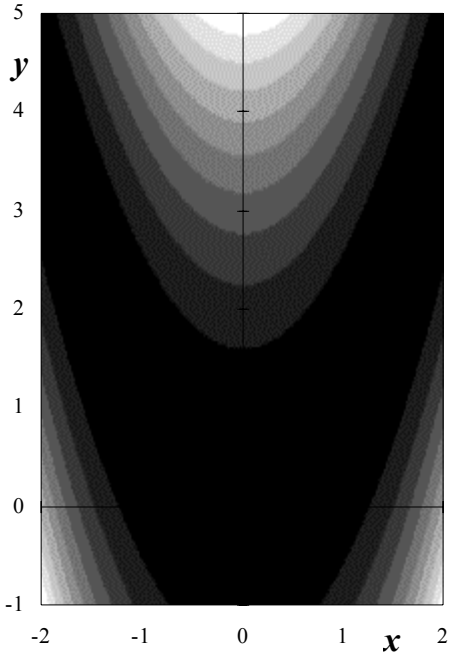

Fig. 1.5.5: A 2-D plot of the Rosenbrock function $z = 100\,(x^2 - y)^2 + (x - 1)^2$ using Mapper00.

We now make a little excursion. Suppose that we want to localize that minimum value of this function. For that we can use the instruction Min(*array*) where *array* denotes the range of data in the table of z, say C5:GU205. In this case, place the instruction `=MIN(C5:GU205)` somewhere outside the data array. It will yield a value of 0.000 because the regular grid just happens to get you right on the proper spot. But where is that minimum located? On the scale of Fig. 1.4.4 you cannot tell, and Excel has no simple command to yield that address: its LookUp functions only work on either rows or columns that, moreover, must have been sorted first. We can, of course, write a function to do the search and produce the location of that minimum value, but here we will see how we can sometimes find it visually.

In this particular example, $z = 100\,(x^2 - y)^2 + (x - 1)^2$ is the sum of two squares, and must therefore be positive for all real values of x and y. Moreover, the minimum value of z is apparently zero or very close to it. We might therefore want to plot $\log(z)$ instead of z, because a logarithmic scale tends to emphasize small values over larger ones. We therefore replace the formula used in the array, `=100*(E$4^2-$B6)^2+(E$4-1)^2`, by its logarithm, `=LOG(100*(E$4^2-$B6)^2+(E$4-1)^2)`, and call the higher-resolution 21-band Mapper0000. Now we obtain Fig. 1.5.6, with two white regions separated by a number of light-gray bands indicating that the minimum lies inside a narrow, lopsided gully, somewhere in the neighborhood of $x = 1$, $y = 1$. The picture obtained with the gradual Mapper0 closely resembles that found with Mapper000, because the contrast between adjacent gray-scale bands with only 5% differences in their darkness is too small in the absence of color. Figure 1.5.6 seems to suggest that the minimum may lie in a fairly narrow gully.

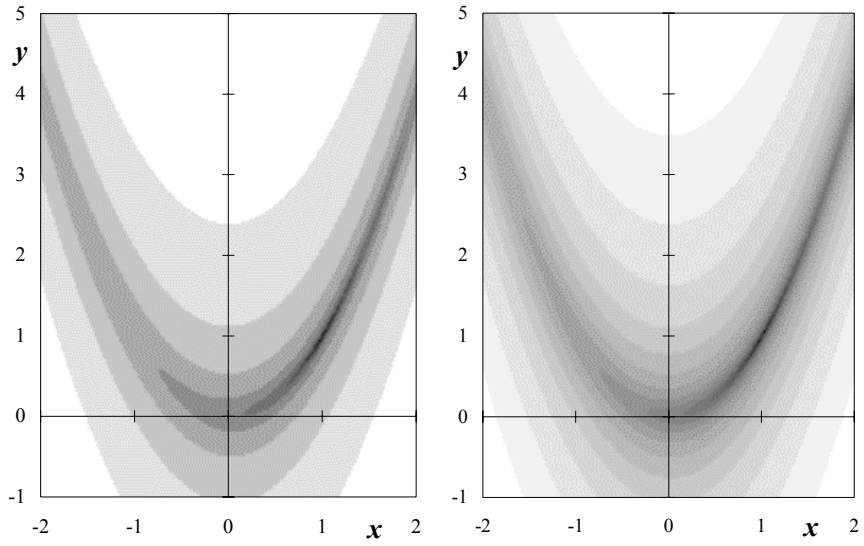

Fig. 1.5.6: Two band-plots of the logarithm of the Rosenbrock function, $\log(z) = \log\{100\times (x^2 - y)^2 + (x - 1)^2\}$, using (left) 11-band Mappper00, and (right) 21-band Mapper 000.

Another available option is to use random values of x and y within the specified range, and then display those that fall inside alternate bands. Random_Plot.xls is set up to do this, as illustrated with the same Rosenbrock function in Fig. 1.5.7. Random_Plot also estimates the minimum and maximum values within its plotting range, which may provide a good starting estimate for subsequent refinement when you are, e.g., looking for the precise location of a minimum. As it turns out, the minimum in Fig. 1.5.7 lies within the wide band where no data points are shown, but the algorithm uses all of its data, including those that are not displayed, to estimate where the minimum and maximum might be located, and actually displays those two points. For better visibility the minimum estimated by Random_Plot.xls is shown in Fig. 1.5.7 as a white circle. Due to the stochastic nature of the samples taken by Random_Plot.xls, the estimated value of that minimum is off by a few percent, but at least it shows the neighborhood in which we might find it.

When we home in on that region we find that there is indeed a *very sharp* minimum at $x = 1$, $y = 1$, which is readily confirmed by substitution of these values into the Rosenbrock formula. The global minimum sits at the bottom of a narrow trench that Random_Plot happened to hit at some distance from the minimum. Note in Figs. 1.5.8 and 1.5.9 that the global minimum is quite narrow, making it difficult to find with the usual numerical search routines. The many graphical tools we now have available allow us to find such a hard-to-find minimum. If its entrance had been located on top of a ridge rather than at the bottom of a trench, it might have been impossible to find by the usual optimization programs.

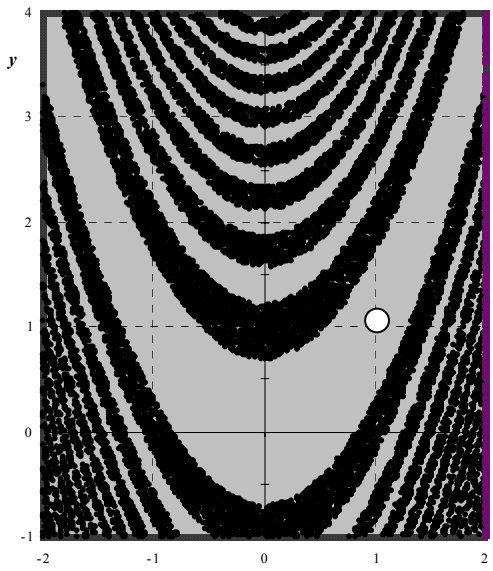

Fig. 1.5.7: A 2-D plot with Random_Plot.xls of the Rosenbrock function $z = 100\ (x^2 - y)^2 + (x - 1)^2$. The white circle indicates the estimated position of its minimum, at $x \approx 1.02703$, $y \approx 1.05704$, and the corresponding function value, $z \approx 0.001$, whereas the correct values are $x = y = 1$ with $z = 0$.

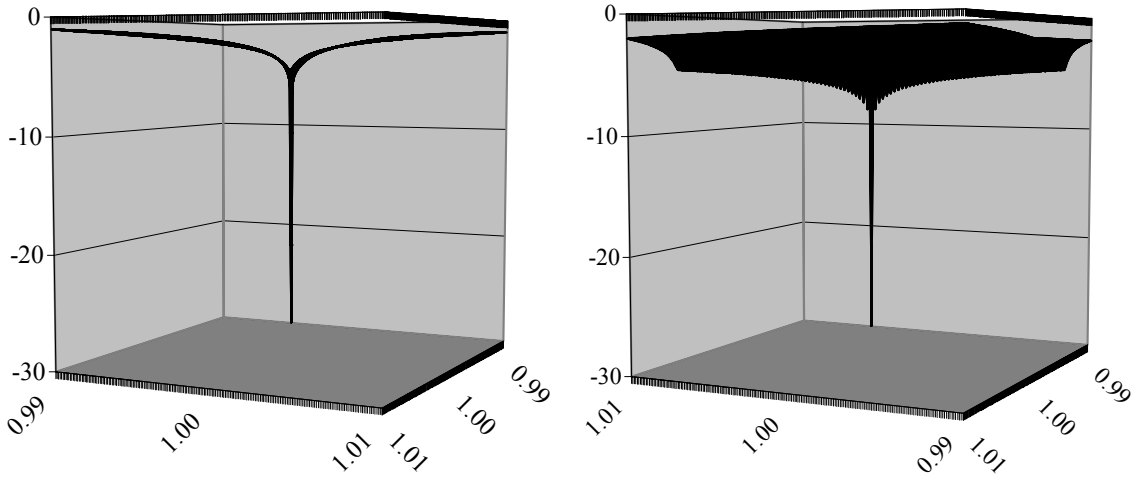

Fig. 1.5.8: Two close-up views in the region around $x = 1$, $y = 1$ of the minimum of the logarithm of the Rosenbrock function $z = 100\ (x^2 - y)^2 + (x - 1)^2$ with Excel's 3-D Surface Chart. The left-hand view is looking along the narrow trench in which the minimum sits, the right-hand view looks perpendicular to that trench. The minimum value of $\log(z)$ is at $-\infty$ which Excel represents as a large but finite negative number.

2.12 How precise is the standard deviation? (AE3 pp. 73-74)

The standard deviation provides an estimate of the precision of a number, i.e., of its reproducibility under near-identical experimental conditions. We now ask the next question: how precise is the standard deviation itself? Could we perhaps keep asking that question of the resulting answer, generating a never-ending series of questions and answers, such as "What is the imprecision of the imprecision of the imprecision, etc.?", much like a continued fraction, or the images of images of images in a hall of mirrors? Fortunately, the question asked in the heading of this section turns out to have a surprisingly simple, definitive answer.

Least squares analysis is based on the assumption that the data follow a Gaussian distribution, reflecting mutually independent, relatively small, random deviations from the average. The variance v (i.e., the square of the standard devation s) can then be shown to follow a so-called χ^2 ("chi-square") distribution. This distribution, as distinct from the Gaussian distribution, depends on only *one* parameter, in this case the number of degrees of freedom $N–P$, where N is the number of data points analyzed, and P the number of adjustable model parameters used, e.g., $P = 1$ for a proportionality, $P = 2$ for a straight line, etc. This χ^2 distribution representing the variance v has a mean $N–P$, a variance v_v of that variance of $2(N–P)$, and a standard deviation s_v of that variance of $\sqrt{2(N-P)}$.

However, we need the standard deviation of the standard deviation, s_s, not the standard deviation of the variance, s_v. (For comparison with the original quantities we need their standard deviations, because they have matching dimensions, which the variances do not.) How do we go from one to the other?

Let a quantity q have a variance v and a standard deviation $s = \sqrt{v}$, so that we can express the quantity together with its imprecision as $q \pm s = q \pm \sqrt{v}$. Likewise we formulate the variance v with its standard deviation s_v as $v \pm s_v$ and the standard deviation s with its standard deviation s_s as

$$s + s_s = \sqrt{v \pm s_v} = \sqrt{v}\sqrt{1 \pm s_v/v} \approx \sqrt{v}\,(1 \pm s_v/2v) = s\,(1 \pm s_v/2v) = s\,(1 \pm s_s/s) \qquad (2.12.1)$$

where we have made the usual assumption that s_v/v is very much smaller than 1, so that we can use the general expansion $\sqrt{(1 \pm \delta)} \approx 1 \pm \delta/2$ for $\delta \ll 1$. From this we see that the relative standard deviation $s_s/s = s_v/2v$ of the standard deviation s of the quantity q is one-half of the relative standard deviation s_v/v of the variance, so that

$$s_s/s = s_v/2v = \sqrt{2(N-P)}\,/\,[2\,(N-P)] = 1\,/\,\sqrt{2(N-P)} \qquad (2.12.2)$$

A rigorous derivation of this result, for the case of the population standard deviation (rather than the sample standard deviation, i.e., with N instead of $N–P$), can be found in, e.g., J. F. Kenney and E. S. Keeping, *Mathematics of Statistics*, 2nd ed., Van Nostrand, Princeton (1951), vol. 2 pp. 170-171.

Note that all the above properties of the χ^2 distribution depend only on the number of degrees of freedom, $N–P$, and are *independent* of the actual x and y values of the data set and their individual standard deviations. We can therefore estimate the imprecision of the standard deviation merely on the basis of the magnitude of $N–P$, as illustrated in Table 2.12.1.

Clearly, the standard deviation is often over-specified, i.e., reported with far more decimals than are significant. For example, performing a simple weighing in triplicate ($N = 3$, $P = 1$, hence $N – P = 2$) yields a standard deviation with a relative precision of ±50%, so that there is no need to insist on many decimal places for such a quantity. Minor differences in standard deviations are often statistically insignificant.

$N–P$	s_s/s	s_s/s in %
2	0.5000	50
5	0.3162	32
10	0.2236	22
20	0.1581	16
50	0.1000	10
100	0.0707	7.1
200	0.0500	5.0
500	0.0316	3.2
1,000	0.0224	2.2
10,000	0.0071	0.7

Table 2.12.1: The relative standard deviation of the standard deviation, as given by (2.12.2), as a function of the number of degrees of freedom, $N–P$.

In order to get a standard deviation with no more than 10% relative imprecision, we would need at least 50 observations, while at least 5 000 measurements would be required for a standard deviation with a 1% maximum imprecision. It is therefore wise to consider most standard deviations as imprecision *estimates*, and the same applies to quantities directly derived from the standard deviation, such as "confidence" measures.

2.18 Phantom relations (AE3 pp. 87-89)

In using least squares it is tacitly assumed that the input data represent *independent* measurements. If that is not the case, quite misleading results may be obtained, as illustrated by the following problem (#9 on page 383) of K. Connors, *Chemical Kinetics, the Study of Reaction Rates in Solution* (VCH, 1990):

> "From the last four digits from the office telephone numbers of the faculty in your department, systematically construct pairs of "rate constants" as two-digit numbers times 10^{-5} s^{-1} at temperatures 300 K and 315 K (obviously the larger rate constant of each pair to be associated with the higher temperature). Make a two-point Arrhenius plot for each faculty member, evaluating $\Delta H^{\ddagger}$ and $\Delta S^{\ddagger}$. Examine the plot of $\Delta H^{\ddagger}$ against $\Delta S^{\ddagger}$ for evidence of an isokinetic relationship."

Essentially, the reader is asked to take two arbitrary two-digit y-values y_1 and y_2, assign them to pre-selected x-values x_1 and x_2 respectively, compute the resulting slope a_1 and intercept a_0, repeat this for a number of arbitrary input pairs y (for the same two x-values), and then plot the resulting a_1-values versus a_0, or vice versa. The actual procedure is somewhat less transparent, since it also involves sorting the input data, a logarithmic transformation, and giving the slopes and intercepts thermodynamic names, all steps that tend to obscure the true nature of the problem. Moreover, the above assignment uses only positive input numbers. Below we will simply take pairs of random two-digit integer values for y, associate them with two fixed x-values such as $x_1 = 300$ and $x_2 = 320$, compute the resulting slopes and intercepts, and then plot these against each other.

Exercise 2.18.1:

(1) In cells B2 and C2 place the labels y1 and y2 respectively. Do the same in cells E2:F2, and in cells H2:I2 deposit the labels a0 and a1 respectively.

(2) In cells B4 and C4 deposit the instruction `=INT(200*(RAND()-0.5))`, which will generate random two-digit integers between –100 and +100. Copy these instructions down to row 23.

(3) The numbers in B4:C23 will change every time you change something on the spreadsheet. In order to have a fixed set of random numbers, highlight B4:C23, copy it with Ctrl⌵c, highlight cell E4, and use Edit ⇨ Paste Special ⇨ Values to copy the *values* of y_1 and y_2 so obtained. After that, use the data in block E4:F23 as your random input data, while ignoring those in B4:C23 that keep changing while you work the spreadsheet.

(4) Based on the data in E4:F23, compute in column H the slope of each pair of data points (x_1, y_1), (x_2, y_2) as $(y_2 - y_1) / (x_2 - x_1)$, and in column I the corresponding intercepts as $(x_2 y_1 - x_1 y_2) / (x_2 - x_1)$.

The data in Fig. 2.18.1 seem to fall on or near a straight line, for which Trendline yields the formula $y = -311.18\ x - 0.8877$, with $R^2 = 0.9983$. Is this what you would have expected for having used random input numbers for y? You *see* a straight line, how can that possibly be *random*? What happens here?

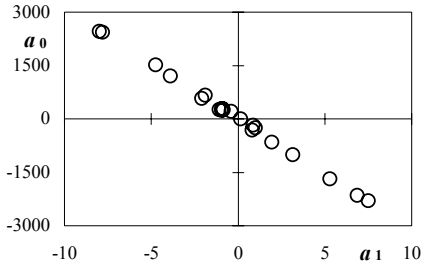

Fig. 2.18.1: An example of a phantom line you might find with $x_1 = 300$ and $x_2 = 320$.

Because each pair of input numbers y of this graph is completely determined by the calculated slope and intercept for given input values of x, the graph uses strongly *linearly correlated* pairs of input data. We already encountered the formula for that correlation, (2.10.1). The sign of (2.10.1) explains the negative correlation (causing the negative slope da_0/da_1 in Fig. 2.18.1), and the effect is the more pronounced the larger is Σx, i.e., the more eccentric are the x-values used. Plotting such slopes and intercepts against each other will then lead to a convincingly linear but physically meaningless relationship, approximating the proportionality $y = -x_{av}\, x$. This merely verifies the correlation (2.10.1) between slope and intercept, as is perhaps more evident after we rewrite $y = -x_{av}\, x$ using more appropriate symbols as $a_0 = -x_{av}\, a_1$.

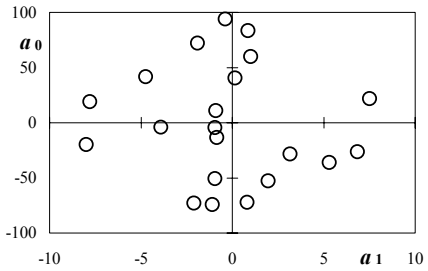

Fig. 2.18.2: The same y-values as in Fig. 2.18.1 analyzed with $x_1 = -10$ and $x_2 = +10$.

This is the origin of the "isokinetic relationship" of J. E. Leffler, *J. Org. Chem.* 20 (1955) 1202, and illustrates how neglecting the covariance can trick you. An extensive discussion of this problem, as well as a suggested solution, was given by Krug et al. in *J. Phys. Chem.* 80 (1976) 2335, 2341. For an interesting (and only seemingly alternative) explanation see G. C. McBane, *J. Chem. Educ*. 75 (1998) 919.

Exercise 2.18.1 (continued):

(6) Use the same *y*-values collected in columns H and I, but now analyze them for a pair of *x*-values *centered* around the average $x_{av} = 310$, so that $x_1 = -10$ and $x_2 = +10$. Does this support the above explanation?

Given that the input data were random, what are the parameters that determine the 'line' in Fig. 2.18.1? There is no significant intercept, just a slope, and the latter is simply $-(\Sigma x)/N$, i.e., minus the average value of *x*. In the above example we have $-(\Sigma x)/N = -(300+320) / 2 = -310$, so that we would expect $y = -310\,x$, which compares well with the result of Trendline, $y = -311.18\,x - 0.8877$, as illustrated in Fig. 2.18.3, not only in terms of its slope but also for the positions of its individual points, which each computed dot neatly nested within the corresponding larger circle of the data. Indeed, as already noticed by Leffler, in many cases the absolute values of the reported slopes of isokinetic plots were close to the average absolute temperatures of the data sets considered. In such cases the isokinetic effect is nothing more than an artifact of incorrectly applied statistics.

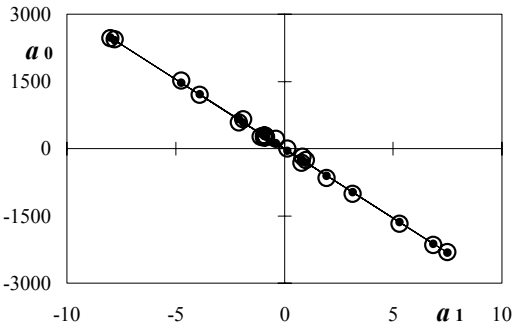

Fig. 2.18.3: The data from Fig. 2.18.1 (large open circles) and, for comparison, those computed as $a_0 = -x_{av}\,a_1$ (small filled circles connected by a thin line).

3.7 Spectral mixture analysis (AE3 pp. 103-104)

Figure 3.7 illustrates the absorption spectra of four fantasy species, made of one or more Gaussian peaks, and of an imaginary mixture made of these species. The spectral peaks were simulated as $a\,\exp[-(x-c)^2/(2b^2)]$; instead of the exponential part you can also use the instruction `=NormDist(x,mean,stdev,false)` to generate Gaussian curves $\left(1/\sigma\sqrt{2\pi}\right)\exp\left[-(x-\bar{x})^2/2\sigma^2\right]$ where $\bar{x}$ is the mean (locating the position of the peak center), σ the standard deviation (defining its width), and where 'false' specifies the Gaussian curve rather than its integral. In exercise 3.7.1 we simulate such spectra, compute the spectrum of a mixture of these components (assuming their additivity, as in Beer's law, and the absence of other light-absorbing species), add different noise to each simulated data point, then use multivariate analysis to reconstruct the composition of that mixture.

Exercise 3.7.1:

(1) In column A deposit wavelengths, and in columns B through E calculate four fantasy spectra, each with one or more Gaussian peaks. Each Gaussian peak requires three constants: an amplitude *a*, a standard deviation *b* or σ, and a center frequency *c* or mean $\bar{x}$.

(2) In columns M through Q generate random Gaussian ('normal') noise, and in columns H through K make somewhat noisy single-component spectra by adding some noise from column N to the spectrum of column B, etc., in order to create more realistic single-species spectra.

(3) Near the top of the spreadsheet enter four concentrations, and use these in column G to make a synthetic 'mixture spectrum' of the four single-component spectra, each multiplied by its assigned concentration, plus added noise from column M. (You could do without columns B through E by adding noise directly to the data in columns B through E, and then subtracting that same noise from the mixture spectrum. Noise in the single-component spectra and in the spectrum of the simulated mixture should of course be independent.)

(4) Plot the spectra of columns G through K, which might now look like those in Fig. 3.7.1. Note that the resulting curve does not show distinct features easily identifiable with any of its constituent spectra. In this particular example we have used the data of Table 3.7.1, together with noise standard deviations of 0.005 for all components as well as for the synthetic mixture. You should of course use your own data to convince yourself that this is no stacked deck.

(5) Highlight the data block in columns G through K, and call LS0 for a multivariate analysis of the mixture spectrum in terms of its four component spectra.

The results of that analysis are shown in Table 3.7.2. Despite the added noise, the absence of stark features, and considerable overlap between the various single-component spectra, the composition of the mixture is recovered quite well.

It is sometimes advocated to analyze mixtures of a few components by taking one dominant point per component, typically its absorption at λ_{max}, and solving the resulting matrix expression. That method cannot discriminate between signal and noise, and has no mechanism to assign statistical estimates for its numerical results either, and its use should therefore be discouraged.

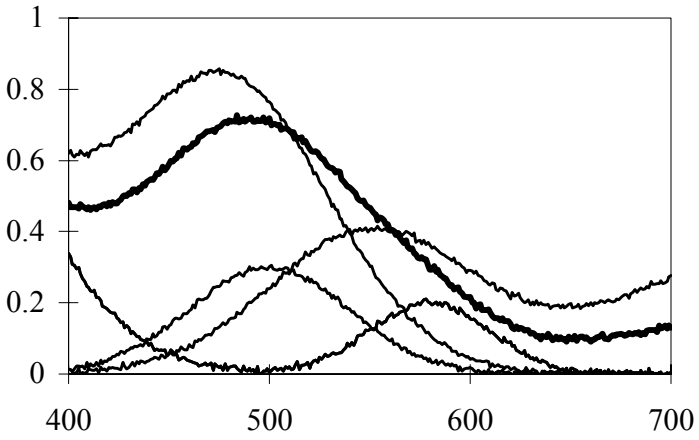

Fig. 3.7.1: The simulated single-component spectra (thin lines) and the spectrum of their mixture (heavy line). The simulation parameters used, as well as the composition of the mixture and the results of its analysis, are listed in Tables 3.7.1 and 3.7.2. Independent Gaussian noise (mean 0, st. dev. 0.005) has been added to all curves.

	ampl:	***mean:***	***st.dev.:***		***ampl:***	***mean:***	***st.dev.:***
curve 1:	300	270	80	*curve 3*:	30	500	40
	100	480	50	*curve 4*:	200	300	60
curve 2:	50	550	50		15	580	30
	70	760	80				

Table 3.7.1: The constants for the Gaussian peaks used in generating Fig. 3.7.1 with the function `NormDist()`.

	curve 1	***curve 2***	***curve 3***	***curve 4***
mixture composition:	0.650	0.500	0.300	0.200
recovered:	0.648±0.003	0.496±0.003	0.305±0.011	0.207±0.007

Table 3.7.2: The assumed and recovered composition of the synthetic mixture.

The above method is simple and quite general, as long as spectra of all mixture constituents are available. In the analysis you can include spectra of species that do not participate in the mixture: for those species, the calculation will simply yield near-zero contributions. However, a missing constituent spectrum will cause the method to fail if its contribution to the mixture spectrum is significant.

A final note: the numbers obtained for the recovered composition are mutually dependent. When a subsequent result depends on more than one concentration, the covariance matrix should be used in its computation, rather than the individual standard deviations.

3.22 The power of simple statistics (AE3 pp. 134-135)

This chapter should neither leave you with the impression that least squares analysis is very complicated, nor that it is mere frosting on the cake, something you do *after* the experiment has run, when you are ready to report the final results. To the contrary, more often than not, the application of least squares analysis is quite straightforward, especially when using the ready-made tools provided by Excel and its add-ins. Moreover, when used *during* rather than after the experimental stage of a study, it can often help you identify (and remedy) experimental problems early on, thereby saving you from having to redo the experiment.

First, a caveat: the following example involves a small set of absorption measurements reported by M. S. Kim, M. Burkart & M.-H. Kim in *J. Chem. Educ.* 83 (2006) 1884, as reproduced in Table 3.22.1. In that communication, these data were not shown for their scientific implications, but only to illustrate how to visualize the process of least squares. Therefore they should not be judged as more than a demo data set, and we will use them here in that vein, merely to highlight how least squares can and should be used. Incidentally, the point listed at the origin was not really measured, but apparently was added subsequently as a mere "dummy data point", and for that reason is not included in Table 3.22.1. We therefore have just five observations, and the question is: how should they be analyzed?

concentration c (M)	***absorbance A***
0.025	0.097
0.050	0.216
0.100	0.434
0.150	0.620
0.200	0.796

Table 3.22.1: The absorbance data listed by Kim et al.

The simplest answer is to follow the theory, i.e., to use Beer's law, $A = abc$, which for a single absorbing species in a non-absorbing medium predicts a proportionality of the form $y = a_1x$, where y is the measured absorbance A, x is the known absorbate concentration c, while the slope a_1 is the product of the molar absorptivity a and the optical path length b through the light-absorbing solution. A sometimes recommended alternative is the straight line $y = a_0 + a_1x$, justified by assuming the possible presence of an absorbing background species (absent in this case, where the solutions were prepared in the lab from reagent-grade components) or a constant instrumental offset. The latter is, of course, entirely avoidable by proper prior calibration.

Exercise 3.22.1:

(1) Enter the data for the absorbance A in one spreadsheet column, and the corresponding concentration c in the column to its immediate right. Label the columns, and make a copy of them elsewhere on the spreadsheet.

(2) Analyze one data set with LS0, the other with LS1. (You can, of course, use LinEst or Regression instead, once with and once without zero y-intercept.) For the proportionality, each of these routines will yield a slope of $4.08_4 \pm 0.06_6$, and a standard deviation s_f of the overall fit of the model function to the data of 0.01_8; for the straight line with arbitrary intercept, the corresponding results are a slope of $3.9_9 \pm 0.1_3$, an intercept of $0.01_4 \pm 0.01_6$, and $sf = 0.01_9$.

Clearly, these data do not support the second model, because the intercept $a0$ is smaller than its standard deviation $s0$. Nor is there any other valid justification for using that model with these data: the test solutions were made by weighing and/or diluting a known, pure chemical in a high-purity, non-absorbing solvent (water), so that there was nothing to cause a constant background absorption. Arguments of in-

strumental drift should not be used, because such drift affects not only the origin but all measured data points. (It is, of course, most readily *observable* at the origin.) Instrumental drift is often, to a first approximation, a linear function of time, in which case one could make multiple series of measurements as a function of time, and apply a bivariate analysis with concentration and time as the independent variables. Preferably, though, once such drift is observed, it is minimized by proper instrument warm-up, using a constant-temperature lab, and by likewise controlling all its other causative factors. Prevention is always better than subsequent remediation.

***Exercise 3.22.1 (continued)**:*

(3) Now plot the data, or use the graph in Kim et al., *J. Chem. Educ.* 83 (2006) 1884, and look along its minor diagonal, i.e., from its origin at the bottom left to its top right corner. In such a foreshortened view, the data show a clear curvature. You will see the same by computing and plotting the residuals to either model curve: the first and last points are too low, the middle point too high for a straight line. Clearly, these data display some curvature not represented by a straight line.

(4) Make two more copies of the original data set, each with an extra column for c^2, and fit the parabolas $y = a_1x + a_2x^2$ and $y = a_0 + a_1x + a_2x^2$ to the data. For the two-parameter model you will find $a_1 = 4.5_3 \pm 0.1_4$, $a_2 = -2.6_7 \pm 0.8_3$, and $sf = 0.01_0$; for the three-parameter model $a_0 = -0.024_0 \pm 0.005_9$, $a_1 = 5.0_1 \pm 0.1_3$, $a_2 = -4.5_8 \pm 0.5_7$, and $s_f = 0.004_0$.

Both parabolas yield plausible parameter values. Assuming for the sake of the exercise that these few observations really reflect a trend in the data rather than mere statistical fluctuations, we find at least three acceptable ways to represent these five data points, as $A = (4.08_4 \pm 0.06_6)\, c$ with $s_f = 0.01_8$, as $A = (4.5_3 \pm 0.1_4)\, c + (-2.6_7 \pm 0.8_3)\, c^2$ with $s_f = 0.01_0$, or as $A = -0.024_0 \pm 0.005_9 + (5.0_1 \pm 0.1_3)\, c + (-4.5_8 \pm 0.5_7)\, c^2$ with $s_f = 0.004_0$; however, the general straight line $y = a_0 + a_1x$ is not among them. Incidentally, this latter conclusion also applies at concentrations below 0.1 M, where there are only two available measurements, because the necessarily exact fit of a two-parameter expression to just two data points has no predictive statistical value.

We see that some very simple considerations using readily available spreadsheet tools allow us to find plausible fits to these data, and to exclude an often advocated but in this case clearly inappropriate model, as indicated *by the data themselves*. However, to reach these conclusions we do need to look not only at the parameter values, but *also* at their imprecision estimates. (Incidentally, using Trendline should be discouraged in the physical sciences precisely because it does not provide such estimates.) And in order to select an appropriate model, once we look for an essentially empirical model describing the data, it is helpful to consider trends in the residuals. If this were a real experiment, it should of course be checked whether such nonlinearity is reproducible and, if it is, whether it is caused by some avoidable instrumental artifact, such as stray light or slits that are too wide, or (much less likely) by an actual deviation from Beer's law. In this way, by fully integrating least squares analysis during the measurement process rather than as an after-the-fact embellishment, we can derive maximal benefits from its use. Properly used statistics can be very helpful at the experimental stage, because they can reveal problems that may need to be addressed and fixed *at that time*.

4.4.3 The titration of an acid salt with a strong base (AE3 pp. 158-160)

We now consider a set of experimental data. As our example we use a recent report by A. L. Soli in *Chem. Educ.* 9 (2004) 42, which lists data observed for the titration of the acid salt KH_2PO_4 with NaOH. We use (4.3.3) where V_a is now the original volume of the titrated acid salt, and ${}^{H}F_a$ is the proton function

$$\begin{aligned} {}^{H}F_a &= [H^+] + [H_3PO_4] - [HPO_4^{2-}] - 2[PO_4^{3-}] - [OH^-] \\ &= [H^+] + (\alpha_3 - \alpha_1 - 2\alpha_0)\, Ca - [OH^-] \\ &= [H^+] + \frac{([H^+]^3 - [H^+]K_{a1}K_{a2} - 2K_{a1}K_{a2}K_{a3})\, C_a}{[H^+]^3 + [H^+]^2 K_{a1} + [H^+]K_{a1}K_{a2} + K_{a1}K_{a2}K_{a3}} - \frac{K_w}{[H^+]} \end{aligned} \qquad (4.4.11)$$

where the alphas are the concentration fractions, labeled with the number of attached protons. Meanwhile we have (4.3.9) for the strong base NaOH. Consequently, the progress of the titration is described by

$$V_b = \frac{-[\mathrm{H}^+] - \dfrac{([\mathrm{H}^+]^3 - [\mathrm{H}^+]K_{a1}K_{a2} - 2K_{a1}K_{a2}K_{a3})C_a}{[\mathrm{H}^+]^3 + [\mathrm{H}^+]^2K_{a1} + [\mathrm{H}^+]K_{a1}K_{a2} + K_{a1}K_{a2}K_{a3}} + \dfrac{K_w}{[\mathrm{H}^+]}}{[\mathrm{H}^+] + C_b - \dfrac{K_w}{[\mathrm{H}^+]}} V_a \qquad (4.4.12)$$

Equation (4.4.12) has two variables, $[\mathrm{H}^+]$ and V_b, which both change continuously during the titration, and seven parameters: the two concentrations C_a and C_b, the original sample volume V_a, the three acid dissociation constants K_{a1}, K_{a2}, and K_{a3} of the triprotic acid H_3PO_4, and the ion product K_w of water. Of these, the volume V_a of the original sample is known from Soli's paper as V_a = 25 mL, and the titrant concentration as C_b = 0.1049 M. The experimental observations are listed in Table 4.4.1, and cover the entire titration curve, from beginning to way past its equivalence point.

V_b	pH	V_b	pH	V_b	pH	V_b	pH	V_b	pH
0.00	4.41	10.10	6.50	21.00	7.20	27.70	8.43	30.80	10.60
0.49	5.06	11.00	6.55	22.00	7.28	27.90	8.61	31.48	10.71
1.00	5.36	12.00	6.61	23.00	7.38	28.03	8.81	32.51	10.85
2.00	5.65	13.00	6.68	23.96	7.48	28.18	9.11	33.41	10.94
2.90	5.78	14.02	6.73	25.00	7.62	28.30	9.30	34.51	11.04
4.10	5.98	14.98	6.79	26.00	7.79	28.50	9.64	35.00	11.07
4.95	6.08	16.00	6.86	26.50	7.92	28.70	9.90	36.02	11.14
6.02	6.19	17.00	6.92	26.75	8.00	28.93	10.05	37.00	11.21
7.30	6.29	18.00	6.98	26.97	8.07	29.20	10.18	38.00	11.26
8.00	6.34	19.01	7.05	27.35	8.22	29.51	10.31	39.11	11.32
9.00	6.42	20.00	7.12	27.51	8.30	30.01	10.44		

Table 4.4.1: The experimental data of Soli for the titration of 25.00 mL aqueous KH_2PO_4 with 0.1049 M NaOH. The volume V_b of added NaOH is in mL.

These 54 data pairs should amply suffice to determine the five remaining unknowns: C_a, K_{a1}, K_{a2}, K_{a3}, and K_w. We will use Solver to assign numerical values to these five unknown parameters (or, more precisely, to their negative logarithms), and SolverAid to estimate the corresponding imprecisions.

Exercise 4.4.4:

(1) Set up the spreadsheet with columns for pH, $[\mathrm{H}^+]$, V_b, $V_{b,calc}$, $V_{b,guess}$, and R, or some other abbreviation for residual.

(2) Also make a column with the labels, and a column with the corresponding numerical values, for the known parameters V_a and C_b, as well as for the unknowns C_a, pK_{a1}, pK_{a2}, pK_{a3}, and pK_w, for K_{a1}, K_{a2}, K_{a3}, and K_w, and for SSR, the sum of squares of the residuals. It is convenient for subsequent use to group these as V_a and C_b, C_a through pK_w, K_{a1} through K_w, and SSR, separated by empty cells.

(3) The duplication of K's and their negative logarithms pK is intentional: the K-values are most convenient for using (4.4.12) in computing $V_{b,calc}$, but the corresponding pK's must be used in Solver, which otherwise may ignore the smaller K-values. Alternatively use Solver ⇨ Options ⇨ Use Automatic Scaling

(4) For V_a deposit the value 25, and for C_b the value 0.1049.

(5) For C_a, pK_{a1}, pK_{a2}, pK_{a3}, and pK_w, use guess values; in Fig. 4.4.6 we have used C_a = 0.08, pK_{a1} = 3, pK_{a2} = 6, pK_{a3} = 11, and pK_{a1} = 14, but feel free to select others, especially ones that show a better initial fit.

(6) Compute K_{a1} as $10^{-\mathrm{p}Ka1}$, and do similarly for the other K-values.

(7) Place the values of V_b and pH in their columns, compute $[\mathrm{H}^+]$ as $=10^{-\mathrm{pH}}$, and in the column for $V_{b,calc}$ compute V_b based on (4.4.12) and the known and guessed parameter values. Copy the resulting values (but not their formulas) to the next column, under the heading $V_{b,guess}$, using Edit ⇨ Paste Special ⇨ Values.

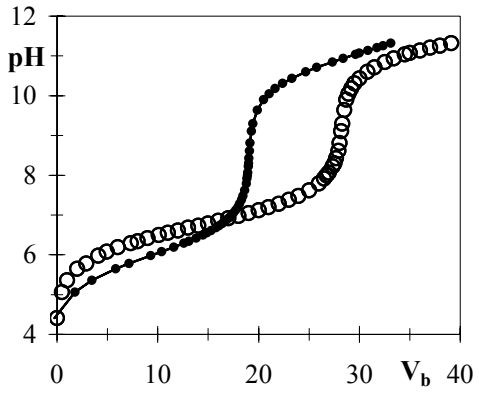

Fig. 4.4.6: The progress curve for the titration of 25.00 mL aqueous KH_2PO_4 with 0.1049 M NaOH. Open circles are the experimental data of Soli, see Table 4.4.1; the curve with small filled circles is computed with the assumed parameter values, *before* Solver is used.

(8) Plot $V_{b,calc}$ vs. pH. The corresponding plot of $V_{b,guess}$ vs. pH keeps a record of the difference Solver makes.

(9) Calculate SSR with the SUMXMY2 function, using the data under V_b and $V_{b,calc}$.

(10) Call Solver, and minimize SSR by letting it adjust the values of the five unknowns: C_a, pK_{a1}, pK_{a2}, pK_{a3}, and pK_{a1}. This will affect the values of $V_{b,calc}$ but not those under $V_{b,guess}$.

(11) Call SolverAid to find the standard deviations of C_a, pK_{a1}, pK_{a2}, pK_{a3}, and pK_{a1}, and their covariance matrix. Also plot the corresponding array of linear correlation coefficients.

(12) Compute the residuals $V_{b,calc} - V_b$, and plot them as a function of pH.

The so-called *equivalence volume* V_{eq} is defined as the volume V_b at which an equivalent amount of base has been added to the acid, i.e., the value of V_b for which $C_aV_a = C_bV_b$. Traditionally, V_{eq} is determined from the titration curve, and then used to compute C_a as C_bV_{eq}/V_a, where one introduces the standard deviations of V_a and C_b. and propagates the associated uncertainties to find that of C_a. In this case we find $V_{eq} = C_aV_a/C_b = (0.11859_2 \pm 0.00008_2) \times 25 / 0.1049 = 28.26_3 \pm 0.01_9$ mL. The approach used here simply bypasses V_{eq}, and directly yields the sought quantity C_a and its standard deviation,

For a comparison with Soli's empirical approach we restrict the analysis to 17 data points (from V_b = 26.00 to 30.01 mL) around V_{eq}. When these are analyzed in the above way we find $C_a = 0.11846_7 \pm 0.00006_5$. Again neglecting the standard deviations in V_a and C_b, this yields $V_b = 28.23_3 \pm 0.01_5$ mL, which can be compared directly with $V_{eq} = 28.22 \pm 0.026\sqrt{17} = 28.22 \pm 0.11$ mL obtained by Soli. We see that the theory-based analysis of these 17 data is some seven times more precise than Soli's strictly empirical approach. In the analysis of high-grade data, the caliber of the model used typically determines the quality of the results.

We now consider these numerical results.

(1) The values obtained for both C_a and K_{a2} are quite satisfactory. This is to be expected: C_a is computed directly, without the intermediary of an equivalence point. The value of C_a is essentially independent of (in this example: neglected) activity corrections, but this does not apply to value of K_{a2}.

(2) The value for K_{a1} obtained is not very precise, because the titration of KH_2PO_4 with NaOH only provides experimental data in the region where pH > pK_{a1}. To obtain a better value for K_{a1} one would have to either titrate H_3PO_4 instead of KH_2PO_4 with NaOH, or titrate the acid salt with a strong acid.

(3) The values obtained for K_{a3} and K_w are not very precise either. As can be seen from their linear correlation coefficient, these two numbers are strongly correlated, i.e., mutually dependent, and consequently neither of them can be determined very well from this type of experiment.

(4) Although these were experimental data, their analysis with a model that neglects activity effects is quite satisfactory in terms of determining the unknown concentration C_a, compare exercise 4.4.3. Of course, the values obtained for the equilibrium constants K_{a1} through K_{a3} and K_w do not agree too well with their literature values, since the latter have been corrected for activity effects by, in effect, extrapolating them to 'infinite' dilution.

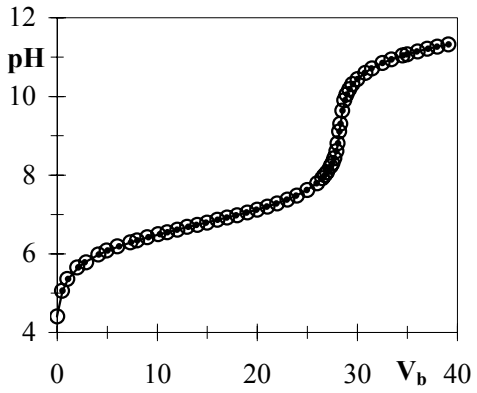

Fig. 4.4.7: The progress curve for the titration of 25.00 mL aqueous KH_2PO_4 with 0.1049 M NaOH. Open circles are the experimental data of Soli, see Table 4.4.1; the small filled circles are computed with the parameter values found by Solver. Note how the individual, computed points nest in the open circles representing the experimental data: they don't merely fit the curve, but their specific locations on it.