

Building a spreadsheet model

R.S. Morris¹⁾, C.W. Rougoor²⁾ & R.B.M. Huirne²⁾

1) *Department of Veterinary Clinical Sciences, Massey University, Palmerston North, New Zealand*

2) *Department of Farm Management, Wageningen Agricultural University, Wageningen, the Netherlands*

Objectives

By the end of this chapter the reader should be able:

- to understand the basic principles of spreadsheets
- to design simple spreadsheet models
- to include risk features into spreadsheet models

18.1 Introduction

When economic analysis first gained acceptance as a decision-making aid in veterinary science, every analysis had to be conducted completely by hand - frequently involving hours of calculations and double-checking. Since then electronic spreadsheets have made the task relatively simple even the first time, and even easier to do if the same analysis must be re-run with new data.

Because repeated financial calculations are very commonly used in business, the spreadsheet caught on like wildfire over a decade ago, and became a major factor in the growth of personal computers. Spreadsheets can be just as useful in veterinary work as in other fields, and this chapter explains the application of the technique at various levels of sophistication. They offer a way for the novice to first develop a simple partial budget, but they offer the expert a powerful shortcut to conducting complex modelling and analyses.

18.2 Structure of spreadsheets

18.2.1 Cells

An electronic spreadsheet consists of a table of individual 'cells', m columns wide by n rows deep. Columns are usually designated by letters and rows by numbers. A cell is therefore uniquely designated by its column letter and row number, such as cell B12, which is the cell at the intersection of the second column and 12th row. The width of columns may be varied to suit the particular needs.

Cells can be allocated to particular uses. Common uses include text fields for labels of various kinds, data fields which expect the user to enter values when the spreadsheet is

run, and calculation fields, which contain the results of calculations based on the data which are entered. The unique feature of spreadsheets which makes them so valuable is the ability to attach a calculation formula to a cell, so that every time a new value is entered in a data entry field which affects a calculated field, the calculation is carried out again either on request or (usually) automatically, so that new values appear in all relevant cells as soon as a number anywhere in the spreadsheet is changed.

The formula for a cell can be viewed and edited at will, making it easy to check its accuracy. Once the formula is checked and permanently stored as part of a particular named spreadsheet, it can be relied on to repeat the calculation accurately as often as wished.

This feature differs from a database management program, which allows manipulation of data entered, but usually requires a formula to be applied to a group of cells by specific decision of the user, and does not allow automatic formulae to be attached to single cells with cross-references to other cells.

18.2.2 Advanced calculation procedures

The formulae in cells can include all of the standard arithmetic operations, but in addition a variety of more advanced operations which allow the spreadsheet to act as a full economic analysis system. For example, the discounting procedure for cost-benefit analysis is available as a standard operation which can be called by its name and applied to a series of annual cost or benefit figures to make cost-benefit analysis easy.

Sequences of calculations which must be carried out repeatedly for a particular application can be automated by means of a recorded set of steps called a **macro**. This can be run to conduct more complex analyses than can be achieved by simply editing data fields in the spreadsheet. Modern macro languages allow almost full programmability of the spreadsheet, permitting it to do things not envisaged by the original designer of the program.

18.2.3 Linking to other functions

Almost all spreadsheets now have databases linked to them, and powerful graphics components, which allow them to take data in a spreadsheet and turn it immediately into a graph of the user's choosing, with many additional features available to create impressive graphical displays of results of spreadsheet analyses. Because the graphing is integral to the spreadsheet, graphs can be easily updated when new data are entered. Some also have presentation creation modules, which allow computer-operated slide shows to be created directly from the spreadsheet information and directly entered text, to give public presentations of findings.

Because spreadsheets are a major component of total software sales, many smaller companies have developed creative **add-on modules** for other companies' spreadsheets, to carry out functions not available within the main spreadsheet. Some of these enhance spreadsheet function, such as the program @RISK, which extends the analytical capacity of the spreadsheet in ways to be described below (see section 18.5). Others add totally new functions, even as extreme as making the spreadsheet double as a word processing program, something for which it was never designed. So if you have a special need and the feature

is not offered by your spreadsheet, you may be able to buy it and 'attach' it to the spreadsheet.

Spreadsheets which operate under Microsoft Windows have far greater capacity than MS-DOS programs to pass information automatically between different programs. This uses two techniques - Object Linking and Embedding (OLE) and Dynamic Data Exchange (DDE). They are best explained by examples. Object embedding means that an 'object' (table of analysis results, graph, etc.) is taken from one program (here the electronic spreadsheet, which is known as the OLE server because it provides the data) and embedded as an exact copy in a file of another program (commonly a word processing program, which is acting as an OLE client or user). This could also be done by pasting through the Windows clipboard, but in that case all connection with the original program is lost. When an object is embedded, and you then click on the object with your mouse in the client application, the computer will automatically load the program in which the object was created and the file which contained the original object. You can then edit it and close the 'server' application, which will return you to the client application and the file you were working on, but with the modified object now appearing on the screen.

When an object is linked rather than embedded, there is a direct linkage between the file in the client application and the file containing the 'object' in the server application. If you change the source file in the spreadsheet, next time you load the client file in your client application it will change the information in the object item to match the data in the source, without the user needing to manually make the changes or even know what the changes are.

When DDE is used, the linking of files is two-way. If a change is made in the file in either of the pair of dynamically connected programs, the 'twin' file in the other program is automatically updated before using it next time.

How does this make spreadsheets more powerful? When working on a report which includes a number of graphs and tables copied from the spreadsheet economic analyses, you can then embed each of them within the document. For modifying the layout or other features, you can go back into the spreadsheet to do it, and keep agreement between the files in the two programs, using all the power of the spreadsheet from within the word processing program. Moreover, you can embed an object in a report and complete it, then later change the spreadsheet file but the graph in the report will stay as it was when you finished the report. This is how embedding differs from linking. Linking can be very useful if you periodically have to update a document (a monthly report to a farmer or to senior managers). It always contains the same tables but the data must reflect new information, such as income and expenditure for the most recent month. Each time the document is opened, it checks to see if the spreadsheet has been altered, and if so it will update the linked objects in the word processing document to make them agree with the spreadsheet. By creating a spreadsheet to store the data and linking them to the report, you can automate and simplify the procedure of producing the monthly report. Dynamic data exchange is the most complex of these procedures to operate, but can be very useful in selected cases - for example if data are coming in continuously into a database through daily entry of new records, and you

want to maintain a spreadsheet file containing summaries of the data accurate up to the latest records entered.

18.3 Choice of program

Fashions change in spreadsheets, and the program Visicalc which started the whole trend has disappeared from the scene. For much of the 1980s the dominant program in the market was Lotus 1-2-3, because it added new features and simplified the way of working with the spreadsheet, but it has lost its dominance. Of the large number of spreadsheet programs developed since Visicalc, three Windows programs now dominate the market: Excel (Microsoft), Lotus 1-2-3 (Lotus/IBM) and QuatroPro (Novell). Most active spreadsheet users work in Windows because of the ability to handle larger spreadsheets, to display superior graphics, print to any printer and to use OLE. The spreadsheet templates supplied with this book will run in all of these. Each program has its own file format, but each can read some of the competing formats as well.

18.4 Formulating a simple economic analysis

18.4.1 Partial budgeting as an example

The most common form of economic analysis used at farm level will be a partial budget. Using as an example a budgeted analysis of parasite control in sheep which was originally prepared by hand (Anderson *et al.*, 1976), the net benefit of the control program using one treatment strategy can be laid out as shown in Figure 18.1. The analysis for this paper had taken a full week of work, but could now be done in a spreadsheet in a fraction of the time, and with greater accuracy.

In the form of a partial budget the analysis fits easily into a spreadsheet format. Column A will be used for row descriptions, and the first row or two in each of the other columns will be used for column headings. In this simple form column B will be used for the data and calculations. Cells can be designated to receive the raw data for the analysis, with blank cells to separate each group of related items from the others. It is important to make the layout easy to read and interpret. There are plenty of rows and columns to use, as long as you keep everything you need to work on at one time on a single screen.

An important basic rule is to have each data item entered in exactly the form it was collected, and have the program do any pre-processing to get the figures into the right form for analysis. For example, put in wool weight and price per kg, and have the program calculate fleece value. Similarly, if you must adjust the figures to take account of deaths during the year, design the spreadsheet to accept figures for the number of deaths, and use this to adjust other parts of the analysis. This saves considerable time, frustration and mistakes - especially when you are trying to do an analysis in front of a farmer.

Cells are designated to receive calculated values derived from the various raw data items, and formulae are entered into these cells to produce the result automatically. Variables used in a calculation are identified by the cell in which they can be found, while constants are entered as numbers. Mathematical operations are designated in the usual way seen on

Building a spreadsheet model

the right-hand side of a mathematical equation, using the representation of each operator required by the particular spreadsheet.

Figure 18.1 Simple spreadsheet: Benefit of 'critical' parasite control strategy over 'no treatment'

1	Column A	Column B	Calculation formula
2	1. Additional returns		
3			
4	Additional fleece wool	40	
5			
6	Capital value of surviving sheep		
7	in critical treatment group	263	
8			
9	Increased value of crutchings	2	
10			
11	Total	305	B4 + B7 + B9
12			
13	2. Returns foregone		
14			
15	Capital value of surviving		
16	sheep in no treatment group	222	
17			
18	Wool salvaged from dead sheep	0	
19			
20	Total	222	B16 + B18
21			
22	3. Extra costs		
23			
24	Extra anthelmintic and labour	13	
25			
26	4. Reduced costs		
27			
28	None relevant	0	
29			
30	Net return	70	(B11+B28)-(B20+B24)
31			
32	Return on invested funds	538	B30 x 100/B24

Do not try to encompass the entire calculation in a single formula, but use extra cells somewhere in the worksheet to show critical intermediate steps in the calculation. By scanning these, errors and unexpected findings can be identified.

Clearly identify the final result and make it easy to view the result cells once the data items have all been entered.

Once the basics of spreadsheet design have been learned, such a spreadsheet can be designed and implemented almost as fast as doing the calculations once by hand, and can then be re-used and varied as much as required. It is also easy to prepare **generic** spreadsheet 'templates' for commonly used analyses, which carry out calculations on blocks of cells, so that a single worksheet can be used for a variety of purposes by simply converting the generic block of rows for 'extra costs' or 'benefits no longer obtained' into a specific analysis by inserting item identifications and data within the cell ranges set aside for that category. This will speed up the work considerably.

18.4.2 Multi-column spreadsheets

Spreadsheets really come into their own when you must link together a number of component analyses, and when there are cross-links between the various parts of the analysis. This will be necessary, for example, in an analysis of methods of improving reproductive performance in a beef enterprise, where the increased calving rate in one year will affect the number of heifer replacements available two years later, and the age distribution of cows in later years.

To represent an analysis of this type, it is necessary to have at least one column for each year, and commonly to have a summary column which shows a cost-benefit analysis of the gains of implementing the reproductive program, in comparison with no action. In this case, the spreadsheet makes it easy to discount costs and benefits as required in a cost-benefit analysis, since the discounting procedure is a built-in function.

For extension purposes, a series of these spreadsheets can be constructed for different districts within the region, using the same worksheet template but varying the data and some of the formulae to express differences between districts. The outcome for each year within each district analysis can then be automatically transferred into a linked regional spreadsheet to produce a summary of expected benefits across the region. Any costs incurred at the regional level can then be subtracted to produce a final overall cost-benefit ratio and net present value of the program. This was done, for example, in a regional analysis of a parasite control program in villages in Thailand (Meemark & Morris, 1989). Changing one variable at the farm or village level in such an analysis will immediately cause the entire analysis to be recalculated. It is also possible to link the district worksheets so that a change in certain variables (such as product and input prices) in the worksheet for district 1 will automatically transfer to all other districts, while figures such as calving rates are kept distinct.

18.4.3 Spreadsheet notebooks and three-dimensional spreadsheets

It is often useful to work with a group of spreadsheets which carry out related analyses

with different sets of assumptions. The leading spreadsheets offer systems for combining a group of spreadsheets as pages in an electronic notebook, so that they can be worked on together very easily.

18.5 Spreadsheet models with risk considerations

Computer models have been used extensively to analyse disease control problems. Often these models are written in a computer language such as Turbo Pascal, and knowledge of such a language is a prerequisite for writing the program. It is possible now to construct moderately complex models purely within electronic spreadsheets, thus allowing people with very limited programming ability to produce models which are dynamic, ie, they represent changes in a system through time, in an iterative fashion. Some spreadsheet programs have special enhancements to assist in such advanced uses. Spreadsheet modelling is an excellent starting point in developing a model, since a functional version can be developed quickly, although if the model is to be used extensively it may be best transferred to a programming language once the full design has been worked out, mainly for gains in speed and ease of use.

The complicating factor in most mathematical models arises from chance or **stochastic elements**. The @RISK computer package is available as an add-in for Lotus 1-2-3, Excel and Symphony and brings advanced modelling and risk analysis to these worksheets (Palisade, 1992). In general, the technique in an @RISK analysis encompasses the following four steps, of which the first three can be supported.

1. Developing a model - by defining a problem or situation in the format of the spreadsheet package you are using.
2. Identifying risk - setting up the risky variables in the worksheet so that their possible values can be specified with probability distributions, and identifying the risky outcome variables in the worksheet results to be analysed.
3. Analysing the model with simulation using random numbers - to determine the range and probabilities of all possible outcomes for the results of the worksheet.
4. Making a decision - based on the results provided and personal preferences.

Probability distribution plays an important role in any analysis which incorporates risk. A probability distribution is a mathematical device for presenting the quantified risk for a variable. There are many forms and types of probability distributions available in @RISK, each of which describes a range of possible values and their likelihood of occurrence. There is a wide variety of distribution types ranging from uniform and triangular distributions to more complex forms such as gamma and Weibull (Figure 18.2).

In @RISK, all distribution types use a set of arguments to specify a range of actual values and distribution of probabilities, as can be seen in Figure 18.2. The normal distribution, for example, uses a mean and standard deviation as its arguments. The mean defines the value around which the bell curve will be centred and the standard deviation defines the spread of values around the mean. Over thirty types of distributions are available in @RISK for describing distributions for uncertain values in the worksheets.

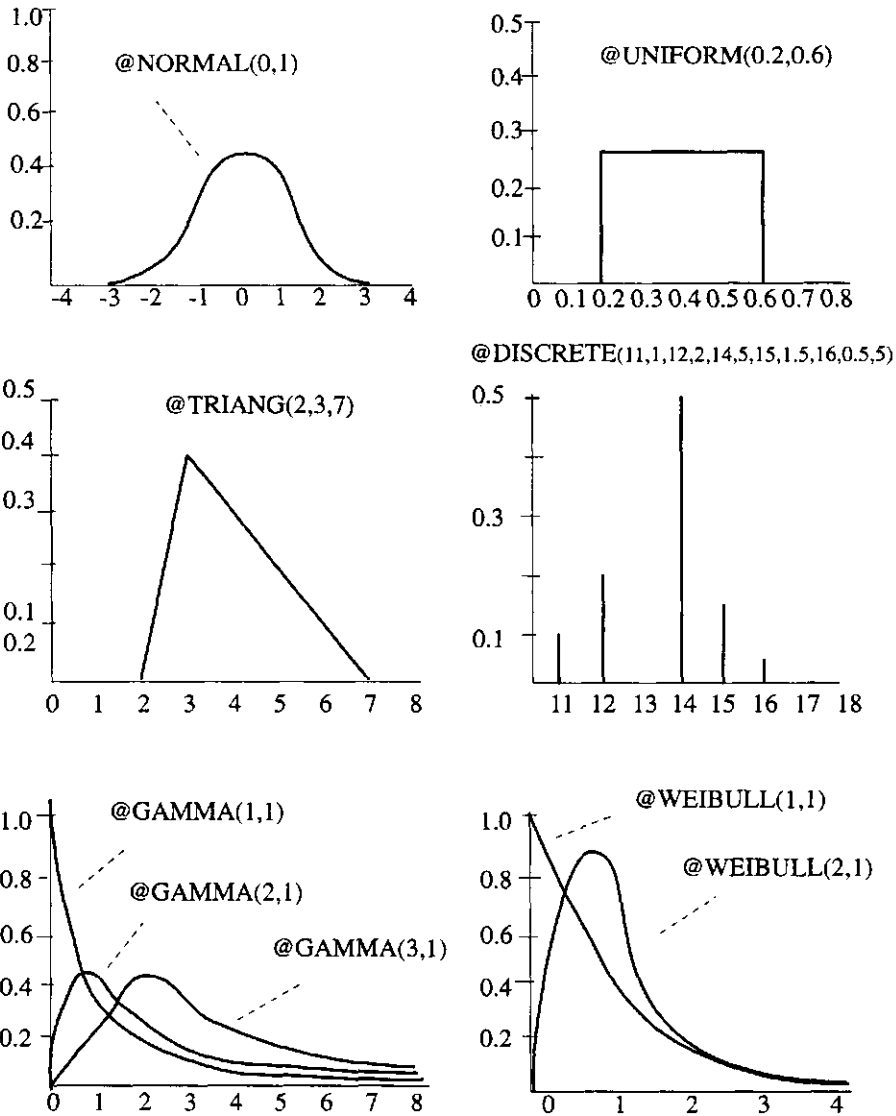


Figure 18.2 Example probability distribution graphs (normal, uniform, triangular, discrete, gamma and Weibull)

In @RISK, uncertain variables and cell values are entered as probability distribution @functions, for example: @TRIANG(A3/2.01,A4,A5), when you are using Lotus 1-2-3. This example is a triangular function with the minimum (actual value in cell A3 divided by 2.01), most likely (actual value in cell A4) and maximum (actual value in cell A5) value as arguments respectively. These @functions can be placed in the worksheet cells and

formulae just like any other 1-2-3 @function.

Sampling is used in @RISK simulation to generate possible values from probability @functions. These sets of possible values are then used to evaluate the worksheet by sampling from the distributions perhaps 100 or 1000 times. Sampling is the process by which values are randomly drawn from input probability distributions. Because of this, sampling is the basis for the hundreds or thousands of repeated 'what-if' scenarios @RISK calculates for the worksheet. Each set of samples represents a possible combination of input values which could occur. Sampling in a simulation is done repetitively - with one sample drawn every iteration from each input probability distribution. With enough iterations, the sampled values for a probability distribution will become distributed in a manner which approximates the known input probability distribution. The statistics of the sampled distribution - mean, standard deviation and higher moments - will approximate the true statistics that were input for the distribution. It is not necessary to make every variable in the spreadsheet stochastic. Commonly this is limited to a small number of critical (ie, most important) variables.

The decision maker should recognize that analysis incorporating risk cannot guarantee that the action which is chosen to follow - even if skilfully chosen to suit the personal preferences - is the best action viewed from the perspective of hindsight. Hindsight implies perfect information, which is never available at the time the decision is made. The decision maker can be guaranteed, however, that (s)he has chosen the best personal strategy given the (imperfect) information that is available.

18.6 User-friendly spreadsheets

When the designer of a spreadsheet template is also the only user, layout is not a critical point in design, although it is wise to keep the structure uncluttered and easy to work with. Once a worksheet is distributed to others, it becomes just as important to produce a layout which is easy to use and error-resistant as it is to have all the calculations correct.

There are some important considerations in achieving this. Most worksheets which contain substantial calculations cannot be fitted into a single screen, so a new design structure is required. This should be based around programmed jumps from cell to cell. The worksheet is divided up into a series of single screens, such as Introduction, Data Entry, context-sensitive Help screens for various parts of the worksheet and Results. There will usually be quite a large number of cells required for intermediate calculations, which the typical user need never see.

A simple way of achieving this is to have blocks of cells, each comprising one screen-full, to display each of the major parts of the worksheet with which the user will want to interact - such as a data entry screen, one or more results screens, and an explanation screen for each area of the results. These screens should be easy to read and interpret, and automatic procedures should be used to jump from the current screen to the next one the user should see, once the user requests the jump. Blocks of screen area can be reserved for explanatory material which the user must read but should never alter. These blocks can be 'locked' so that the cursor cannot enter them in normal use.

Within blocks dedicated to data entry or calculation, subblocks should be laid out which contain closely related material which can be handled as a unit. For example, it may be necessary to add up rows 12 to 30 in column G. Instead of having to refer to each cell in the sequence by name in each formula as would be necessary if they were spread around, they can be handled as an inclusive block SUM (G12..G30) or similar.

Intermediate calculations should be stored in columns separated from the user-accessible ones sufficiently, so that the user is never aware of their existence in normal circumstances. This allows for ease of use but also allows the expert to check calculations easily. Annotation features in a spreadsheet or an add-in program allow explanatory notes to be attached to cells of a spreadsheet, mainly as an aide memoir to the logic on which a complex cell formula is based. This feature is extremely useful, since otherwise a formula that was quite clear six months ago becomes incomprehensible when reviewed, and may then be modified in a way which undermines the structure of the spreadsheet.

If any procedure must be carried out repeatedly in working with a spreadsheet, it is most efficient to program it in advance by writing a macro, a small program which most computer users can prepare. This carries out a specific set of steps within the spreadsheet, which may vary from something as simple as changing a font in a single step, to running a complex analysis by hitting one key.

The spreadsheets supplied with this book all work in accordance with these principles as much as possible, and use techniques such as macros. Not all features of each single spreadsheet program could be fully used, however, because the cases were designed such that they run in all 3 programs involved (ie, Excel, Quattro and Lotus).

18.7 Using the spreadsheet

Normally a well-designed template will take the user in a sequence of automated jumps through all the data entry stages, and after the last item has been entered it will automatically carry out its recalculation and position the cursor at the first result screen. Recalculation can however either be made automatic or be made to await a user instruction. One small problem with automated spreadsheets is that an error which is recognized after a data entry point is past cannot be corrected without repeating the run.

For a very polished spreadsheet, this can be overcome by copying each data item to a check screen just before calculation commences, where the user is given a chance to review the items entered and to loop back to correct any specific errors. After confirming the accuracy of the values, calculation will commence.

One major use of a spreadsheet is to compare the effect of some potential improvement with a 'base' analysis representing the status quo. In simple analyses this must be done by printing the result screen for the base analysis and then running the alternative and printing that to allow a comparison of the two. An advanced alternative is to allow the result screens to show two sets of figures, one for the base analysis and one for the alternative. The base analysis can either be repeated each time a comparison is done, or be processed separately and merely stored visually in the result screens to remind the user of the baseline values against which the alternative should be compared. Printed copies will contain both values.

An important part of an economic appraisal is **sensitivity analysis**, where the most influential biological and price variables in the analysis are each varied to test how susceptible the predicted financial return is to differences between estimated and actual items in the analysis, and variation which can be expected over time in major variables such as product price. The limits within which sensitivity analysis is conducted are a matter of choice to fairly represent the field situation. With regard to prices, the long-run range of lowest to highest may be taken (perhaps adjusted to current year equivalence), or else fixed percentage variation may be taken (say 10 and 20% above and below current values). With regard to biological variation in measured variables such as growth rate or pregnancy rate, sensitivity analysis may either be taken at 1 and 2 standard deviations from the measured sample mean, or again be allowed to vary by a percentage of the mean value. It is good practice to consider the mean or expected value and two levels of variation above and below the mean, thus requiring five calculations in total.

Spreadsheets really come into their own in sensitivity analysis, because what would otherwise have been a tedious process of repeating the entire calculation becomes a simple matter of changing one or more variables and viewing the result. It even becomes possible to adjust two variables at once (say product price and growth rate response) to produce a bivariate response surface, something which is far too cumbersome by hand because 25 evaluations are required for a single sensitivity analysis. If desired, the 25 results can be stored and then graphed in a 3-dimensional representation of a response surface within the same program.

18.8 Examples of the use of spreadsheets in practice

There are numerous spreadsheet templates of veterinary economic calculations available, either through distribution from the developer or through publication of the procedure in a scientific journal. Examples at the basic level include Quek *et al.* (1986), and Gulbenkian & Viegas (1988). Dijkhuizen *et al.* (1986) provide a much more extensive analysis system using a spreadsheet to analyse sow replacement economics, embodying most of the principles outlined. Carpenter (1988a,b) demonstrates the use of a spreadsheet in epidemiological modelling. Numerous other veterinary examples of spreadsheets exist in both epidemiology and economics.

In practice some very complicated simulation models with many interacting demands and services have been used. Whole sections of an organization have been simulated. The limits are only the capacity of the computer and the time taken to work and test the program. Much of the work has been facilitated by devising special computer simulation languages such as @RISK. Quantitative analysis techniques have gained a great deal of popularity with decision makers and analysts in recent years. Unfortunately, many people have mistakenly assumed that these techniques are magic black boxes that unequivocally arrive at the correct answer or decision. No technique, including those used by @RISK, can make that claim. These techniques are tools that can be used to help make decisions and arrive at solutions. Like any tool, they can be used to good advantage by skilled practitioners, but should never be used as a replacement for personal judgment.

References

- Anderson, N., Morris, R.S. & McTaggart, I.K., 1976. Analysis of two schemes for the anthelmintic control of helminthiasis in weaned lambs. *Australian Veterinary Journal* 52: 174-180.
- Carpenter, T.E., 1988a. Stochastic epidemiologic modeling using a microcomputer spreadsheet package. *Preventive Veterinary Medicine* 5: 159-168.
- Carpenter, T.E., 1988b. Microcomputer programs for Markov and modified Markov chain disease models. *Preventive Veterinary Medicine* 5: 169-179.
- Dijkhuizen, A.A., Morris, R.S. & Morrow, M., 1986. Economic optimization of culling strategies in swine breeding herds, using the 'Porkchop' Computer Program. *Preventive Veterinary Medicine* 4: 341-353.
- Gulbenkian, M. & Viegas, F.J.T., 1988. OVISERRA: A computerized herd management system for dairy ewes. *Acta Veterinaria Scandinavica, Supplementum* 84, p. 530.
- Meemark, N. & Morris, R.S., 1989. Economic analysis of the benefits of the Basic Animal Health Service. *Proceedings International Seminar on animal health and production services for village livestock, Khon Kaen, Thailand, August 1989: 465-477.*
- Palisade, 1992. @RISK Users Guide, Risk analysis and simulation add-in for Lotus 1-2-3, Palisade Corporation, 31 Decker Rd, Newfield, NY 14867, 374 pp.
- Quek, K.L., Ng, C.Y. & Chang, C.F., 1986. PigMoney: Application of electronic spreadsheet in the economics of pig production. *Proceedings 4th International Symposium on Veterinary Epidemiology and Economics, Singapore, 1985: 256-257.*